POLITECNICO DI TORINO

Corso di Laurea Magistrale in Ingegneria Elettrica

Tesi di Laurea

Clustering-based definition of the electricity market zones

Relatore
Prof. Gianfranco Chicco
Correlatore:
Dr. Andrea Mazza

Laureando
Andrea Griffone

Aprile 2019

"I have never met a man so ignorant I couldn't learn something from him."

"Non ho mai incontrato un uomo così ignorante dal quale non abbia potuto imparare qualcosa."

[Galileo Galilei]

Summary

In the last years, the progressively growing renewables' production capacity has increased the fluctuating infeed in the nodes of the electrical networks. Consequently, as the lines' transmission capacities have not been upgraded simultaneously, this has resulted in the increase of congestions in the electricity transmission grids across Europe. Thereby, countries like Germany have lately seen an enormous increase in their redispatch cost. There in particular, for instance, the redispatch cost has more than tripled from 2012 to 2015. Therefore, the need for an improved congestion management has day by day become more impelling within the continent, to cope in an adequate way with the increase of congestions in the short and medium term, namely, before the still necessary power networks upgrade. As a result, having recognized inside the establishment of optimal zonal pricing mechanisms the answer to this necessity, the European Union has emanated in 2011 through the Agency for the Cooperation of Energy Regulators ($ACER$) the Framework Guidelines on Capacity Allocation and Congestion Management for Electricity ($CACM$), which among other things have tried to clarify the rules for creating optimal zonal configurations. This made sense, since current European electricity pricing schemes are all based on uniform configurations or non-optimal zonal ones (namely, made up of price zones whose borders are defined through national boundaries or Transmission System Operator's experience on most congestible power network's lines, without any electrical or economic foundation). With these zones, the congestion management is unavoidably non-optimal, because often subject to cases in which the congestion alleviation requires to be made manually and hence costly by the Transmission System Operator itself. Moreover, there are frequently misleading economic signals, caused by a uniform price or non-optimal zonal prices unable to effectively reveal the power network's specific condition, which are not capable of driving the system's performance improvement towards its highest possible increase. Unfortunately, the aforementioned European guidelines only managed to provide general rules on the topic, without being able to successfully carry out the search for an optimal zonal configuration.

For this latter reason, the last years have seen progressively growing the scientific literature regarding the subject. And this thesis equally fits in this scenario, by trying to give a methodology aimed at locating the most suitable technique to deterministically define an optimal zonal configuration, unavoidably required to put in force an optimal zonal pricing mechanism. In particular, in fact, this thesis contains a three-level methodology. At the first level, the zonal configurations' optimality requirements are clearly stated through objective and quantitative parameters, which permit to better identify price zones' optimality rather than the above said European general rules. In the second one, having

uniquely recognized the desired output through tangible parameters, the apparently most suitable clustering algorithms to reach it are presented. The algorithms tested differ with respect to the simple geographical clustering, which could be used to produce price zones, but actually has to be rejected because only capable of giving the zonal configurations with transnational borders, different from the optimal ones above mentioned. Therefore, the second level of this thesis describes the methodology used to run a K-means clustering, a K-medoids one, a hierarchical one, a genetic algorithm and a price differential clustering. These algorithms are executed in two versions, using the Local Marginal Prices (*LMPs*) and Power Transfer Distribution Factors (PTDFs). The changes aimed at better complying with the previously declared zonal configurations' optimality requirements are described in details. After that, the methodology's third level provides a series of price zones' assessment criteria, both in terms of clustering validity indicators and economic efficiency ones. These criteria intend to evaluate the newly defined zonal configurations' optimality, and also to allow the comparison among the different price zones definition techniques previously chosen inside the second level.

Eventually, the methodology created is applied to a real case study represented by a reduced model of the European transmission grid, to test its effectiveness. The results highlight that some of the considered clustering algorithms are clearly inappropriate to work out the problem of defining an optimal zonal configuration, like the price differential clustering and the genetic algorithm, while others have comparable performance in terms of defined price zones' optimality. Anyway, among these last the Matlab[1] LMPs-based hierarchical clustering and LMPs-based K-medoids seem to represent the best techniques to fulfill this thesis' objective, as proved by their respective 1^{st} and 2^{nd} place inside the final ranking contained into Table 4.14. Nevertheless, further evaluations and researches should be done on the subject, for instance by considering also other clustering algorithms and distance metrics, in order to strengthen the aforementioned hypothesis.

KEYWORDS:
Clustering algorithm, Local marginal prices, LMPs, Power transfer distribution factors, PTDFs, Optimal zonal configuration, Zonal pricing mechanism, CACM, Congestion management, Price zone, Bidding zone, Bidding area, Market zone, Electricity market, Power market, Energy market, Cluster analysis, Power system economics, Zonal marginal prices, ZMPs, Grid partition, Zonal network model, K-means clustering, K-medoids clustering, Hierarchical clustering, Genetic Algorithm, Price differential clustering, Market modelling, Electricity market design, Nodal pricing, Uniform pricing.

[1]The distinction between "Matlab" and "customized" clustering algorithm is part of this work's nomenclature. It is afterwards described into *Section 3.2*.

Acknowledgements

Ho sognato per anni questo momento, cercando di immaginare l'emozione che avrebbe scaturito in me questo traguardo. Nei momenti più duri, quando gli esami sembravano una montagna troppo alta da scalare, la mente mi portava qui, alimentando il mio desiderio di conoscenza. Oggi finalmente quel traguardo è vicino, come non mai, e l'emozione supera già di gran lunga le aspettative.

Quello che sta per concludersi è stato un percorso bellissimo, che rimarrà per sempre impresso nei miei ricordi. Questi anni mi hanno dato tanto, dal punto di vista umano oltre che didattico, e di questo voglio ringraziare in primo luogo tutti i professori che ho avuto, dal primo all'ultimo, perché senza di loro non starei scrivendo ora queste righe. Allo stesso modo voglio ringraziare tutti i compagni di corso, o "colleghi" che dir si voglia, conosciuti in questi anni, che più di una volta si sono dimostrati fondamentali tra gli impegnativi banchi del Politecnico.

Doveroso e sentito ringraziamento va inoltre ai miei relatori, il Professor Gianfranco Chicco e il Dottor Andrea Mazza, che nell'ultimo anno mi hanno sempre seguito con pazienza, professionalità, disponibilità e competenza, permettendomi di esprimere il massimo all'interno di questo lavoro.

Un ringraziamento speciale va alla mia famiglia, per aver costantemente creduto in me. A mia madre Marisa, per avermi insegnato il valore del sacrificio, e mio padre Mario, per avermi trasmesso l'amore verso questa disciplina. Voi mi avete fatto capire che per mettere al mondo un figlio basta un attimo, ma per crescerlo occorre una vita intera. Pertanto dedico questo lavoro a voi, perché oggi più che mai mi sento orgoglioso di essere vostro figlio.

Un pensiero finale va a chi c'è stato, a chi avrebbe dovuto esserci ma non ha potuto.
Un caloroso e forte abbraccio.

Andrea

Contents

List of Figures x

List of Tables xiii

1 Introduction 1
- 1.1 The need for an optimal zonal configuration . . . 1
- 1.2 How to define an optimal zonal configuration . . . 3
- 1.3 Thesis' backbone . . . 5

2 State of the art 7
- 2.1 References summary . . . 7
- 2.2 Clustering features summary table . . . 24
- 2.3 Clustering techniques summary table and descriptions . . . 26
- 2.4 Distance metrics summary table . . . 38
- 2.5 Clustering algorithms' strengths and weaknesses . . . 42

3 Methodology 57
- 3.1 Optimality requirements for zonal configurations . . . 58
- 3.2 Suitable clustering algorithms . . . 63
  - 3.2.1 Suitable clustering algorithms' inputs . . . 66
  - 3.2.2 Suitable clustering algorithms' changes . . . 69
- 3.3 Assessment criteria for zonal configurations . . . 79
  - 3.3.1 Zonal configurations' clustering validity indicators . . . 79
  - 3.3.2 Zonal configurations' economic efficiency indicators . . . 80

4 Case Study 83
- 4.1 Description of the case study's electricity grid . . . 83
- 4.2 A priori considerations . . . 86
  - 4.2.1 Insufficiency of the penalty factor technique . . . 86
  - 4.2.2 Inadequacy of the *GA*'s starting population random initialization . . . 91
  - 4.2.3 The K-means and K-medoids initialization . . . 93
- 4.3 Methodology application . . . 95
  - 4.3.1 An input *BAs* number for each clustering algorithm . . . 95
  - 4.3.2 Resulting zonal configurations and their assessment . . . 106
  - 4.3.3 Methodology's best suitable clustering algorithms . . . 124

5 Conclusions 139
5.1 The *GA*'s false optimality . . . 139
5.2 The outsider *PDC* . . . 140
5.3 PTDFs-based algorithms vs LMPs-based ones . . . 141
5.4 The optimality exclusivity . . . 142
5.5 The methodology's worst clustering algorithms . . . 142
5.6 The methodology's best clustering algorithms . . . 143
5.6.1 Nodal *PTDFs* of most congestible lines as valid clustering feature . 144
5.6.2 The penalty factor technique's double face . . . 144
5.6.3 The podium . . . 145
5.7 Final thoughts and future developments . . . 145

A Secondary references analysis 149
A.1 References summary . . . 149
A.2 Clustering features summary table . . . 155
A.3 Clustering techniques summary table and descriptions . . . 156
A.4 Distance metrics summary table . . . 162
A.5 Clustering algorithms' strengths and weaknesses . . . 164

Bibliography 169

List of Figures

3.1 Block diagram of the thesis' methodology. 57

4.1 Map of the case study's reduced model of European transmission network. 84
4.2 European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 5 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function. 88
4.3 European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 13 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function. 89
4.4 European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 20 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function. 90
4.5 European transmission network's zonal configurations produced by the methodology's *GA*. With 5,13 and 20 *BAs* requested by the user and post-processing usage of the "CheckBAsConnection" handwritten function. . . . 94
4.6 Clustering validity indicators' trends from 2 to 10 *BAs* for zonal configurations coming from methodology's Matlab algorithms. 97
4.7 Clustering validity indicators' trends from 11 to 20 *BAs* for zonal configurations coming from methodology's Matlab algorithms. 97
4.8 Economic efficiency indicators' trends from 2 to 10 *BAs* for zonal configurations coming from methodology's Matlab algorithms. 98
4.9 Economic efficiency indicators' trends from 11 to 20 *BAs* for zonal configurations coming from methodology's Matlab algorithms. 98
4.10 Clustering validity indicators' trends from 2 to 10 *BAs* and from 4% to 16% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms. 99
4.11 Clustering validity indicators' trends from 11 to 20 *BAs* and from 18% to 32% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms. 99

4.12 Economic efficiency indicators' trends from 2 to 10 *BAs* and from 4% to 16% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms. 100
4.13 Economic efficiency indicators' trends from 11 to 20 *BAs* and from 18% to 32% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms. 100
4.14 Geographical representation of the Matlab algorithm's zonal configurations which result from 5 requested *BAs*. 106
4.15 Geographical representation of the customized algorithm's zonal configurations which result from 5 requested *BAs* and 26% of average *LMPs* tolerance. 107
4.16 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance. 107
4.17 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance. 108
4.18 Geographical representation of the Matlab algorithm's zonal configurations which result from 6 requested *BAs*. 109
4.19 Geographical representation of the customized algorithm's zonal configurations which result from 6 requested *BAs* and 26% of average *LMPs* tolerance. 109
4.20 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 6 requested *BAs* and 26% of average *LMPs* tolerance. 110
4.21 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance. 110
4.22 Geographical representation of the Matlab algorithm's zonal configurations which result from 7 requested *BAs*. 111
4.23 Geographical representation of the customized algorithm's zonal configurations which result from 7 requested *BAs* and 26% of average *LMPs* tolerance. 112
4.24 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 7 requested *BAs* and 26% of average *LMPs* tolerance. 112
4.25 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 7 requested *BAs* and 26% of average *LMPs* tolerance. 113
4.26 Geographical representation of the Matlab algorithm's zonal configurations which result from 11 requested *BAs*. 114
4.27 Geographical representation of the customized algorithm's zonal configurations which result from 11 requested *BAs* and 12% of average *LMPs* tolerance. 114
4.28 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 11 requested *BAs* and 12% of average *LMPs* tolerance. 115

4.29 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 11 requested *BAs* and 12% of average *LMPs* tolerance. . . . 115
4.30 Geographical representation of the Matlab algorithm's zonal configurations which result from 13 requested *BAs*. . . . 116
4.31 Geographical representation of the customized algorithm's zonal configurations which result from 13 requested *BAs* and 10% of average *LMPs* tolerance. . . . 117
4.32 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 13 requested *BAs* and 10% of average *LMPs* tolerance. . . . 117
4.33 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 13 requested *BAs* and 10% of average *LMPs* tolerance. . . . 118
4.34 Geographical representation of the Matlab algorithm's zonal configurations which result from 14 requested *BAs*. . . . 119
4.35 Geographical representation of the customized algorithm's zonal configurations which result from 14 requested *BAs* and 8% of average *LMPs* tolerance. 119
4.36 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 14 requested *BAs* and 8% of average *LMPs* tolerance. . . . 120
4.37 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 14 requested *BAs* and 8% of average *LMPs* tolerance. . . . 120
4.38 Geographical representation of the Matlab algorithm's zonal configurations which result from 15 requested *BAs*. . . . 121
4.39 Geographical representation of the customized algorithm's zonal configurations which result from 15 requested *BAs* and 6% of average *LMPs* tolerance. 122
4.40 *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 15 requested *BAs* and 6% of average *LMPs* tolerance. . . . 122
4.41 *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 15 requested *BAs* and 6% of average *LMPs* tolerance. . . . 123
4.42 Combined view where the 98 zonal configurations' points are colored depending on the origin test. . . . 129
4.43 Enlargement of the Pareto front where the 98 zonal configurations' points are colored depending on the origin test. . . . 130
4.44 Combined view where the 98 zonal configurations' points are colored according to the clustering algorithm. . . . 131
4.45 Enlargement of the Pareto front where the 98 zonal configurations' points are colored according to the clustering algorithm. . . . 132
4.46 Pareto front recall, for the analytic hierarchy process 133

List of Tables

2.1 Clustering features summary table of references which deal with optimal *BAs* definition using clustering algorithms. . . . 25
2.2 Clustering algorithms summary table of references which deal with optimal *BAs* definition using clustering algorithms. . . . 26
2.3 Distance metrics summary table of references which deal with optimal *BAs* definition using clustering algorithms. . . . 39

3.1 Inputs summary table of the methodology's suitable clustering algorithms. 70
3.2 Legend of the modifications summary table of the methodology's suitable clustering algorithms. . . . 77
3.3 Modifications summary table of the methodology's suitable clustering algorithms. . . . 78

4.1 Overall minimum points of the zonal configurations' assessment indicators trends, as a function of the *BAs* number. . . . 101
4.2 Trade-off user-defined inputs of the methodology's clustering algorithms, in terms of *BAs* numbers or Avg *LMPs* RoT. . . . 104
4.3 Trade-off user-defined inputs of the methodology's clustering algorithms, in terms of *BAs* numbers or Avg *LMPs* RoT. . . . 105
4.4 Methodology's best clustering algorithms of test 1. . . . 108
4.5 Methodology's best clustering algorithms of test 2. . . . 111
4.6 Methodology's best clustering algorithms of test 3. . . . 113
4.7 Methodology's best clustering algorithms of test 4. . . . 116
4.8 Methodology's best clustering algorithms of test 5. . . . 118
4.9 Methodology's best clustering algorithms of test 6. . . . 121
4.10 Methodology's best clustering algorithms of test 7. . . . 123
4.11 Ranking of the methodology's clustering algorithms over the 7 tests. According to *CVIs*. . . . 125
4.12 Ranking of the methodology's clustering algorithms over the 7 tests. According to *EEIs*. . . . 126
4.13 Ranking of the methodology's clustering algorithms over the 7 tests. According to *CVIs* and *EEIs* together. . . . 127
4.14 Ranking of the methodology's clustering algorithms over the 7 tests. According to the sum of both *CVIs* and *EEIs* for each partitioning technique. 128

4.15 Summary of compromise zonal configurations. 133

A.1 Clustering features summary table of references which deal with clustering algorithms for purposes different from *BAs* definition. 155
A.2 Clustering algorithms summary table of references which deal with clustering algorithms for purposes different from *BAs* definition. 156
A.3 Distance metrics summary table of references which deal with clustering algorithms for purposes different from *BAs* definition. 162

List of Nomenclature and Acronyms

Acronym	Meaning
ACER	Agency for the Cooperation of Energy Regulators
ATC	Available Transmission Capacity
Avg	Average
BA	Bidding Area
CACM	Capacity Allocation and Congestion Management
CCI	Congestion Contribution Identification
CDF	Congestion Distribution Factor
CDI	Clustering Dispersion Indicator
Cmzd	Customized
CSI	Cluster Size Index
CSP	Cluster-based Similarity Partitioning
CVI	Clustering Validity Indicator
CWE	Central Western European
DBI	Davies-Bouldin Index
DCOPF	Direct Current Optimal Power Flow
DSO	Distribution System Operator
EC	Entropy Coefficient
ED	Electrical Distance
EEI	Economic Efficiency Indicator
ENTSO-E	European Network of Transmission System Operators for Electricity
EU	European Union
GA	Genetic Algorithm
GSK	Generation Shift Key
HC	Hierarchical Clustering
HHI	Herfindahl-Hirschman Index
IEEE	Institute of Electrical and Electronics Engineers
IOP	Imitating Out-Point
ISO	Independent System Operator
KDD	Knowledge Discovery in Databases
KKT	Karush Kuhn Tucker
KPI	Key Performance Indicator

Acronym	Meaning
LMP	Local Marginal Price
MIA	Mean Index Adequacy
Mtlb	Matlab
NN	Neural Network
NP	Nodal Pricing
NWG	Node With Generators
NWOG	Node WithOut Generators
OP	Optimization Problem
PC	Principal Component
PCA	Principal Component Analysis
PDC	Price Differential Clustering
PTDF	Power Transfer Distribution Factor
PUN	Prezzo Unico Nazionale
RE	Redispatch Effort
RES	Renewable Energy Sources
RNN	Recurrent Neural Network
RoT	Range Of Tolerance
RSI	Residual Supply Index
SMI	Similarity Matrix Indicator
SNP	Sequential Network Partition
TS	Tabu Search
TSO	Transmission System Operator
UP	Uniform Pricing
ZMP	Zonal Marginal Price
ZP	Zonal Pricing

Chapter 1

Introduction

1.1 The need for an optimal zonal configuration

Giving a price to electricity is one of the key aspects in electrical power systems, because energy prices markedly influence both economic investments and operation decisions. Nowadays, there are mainly three categories of pricing mechanisms used in competitive electricity markets: uniform, nodal and zonal pricing.

The former, in which there is a single price for the whole power network for each hour of a day, is still used in many countries mainly due to historical reasons [35]. Nevertheless, despite being simple, it is an approach that conceals a series of disadvantages. On the one hand, because the equilibrium set on its market does not take into account the security requirements of the grid [26]. Thus, it frequently becomes unfeasible due to the occur of congestions, which require manual, and hence costly, readjustments by the Transmission System Operator (TSO) to be alleviated. On the other hand, because using a single price for an entire power network may prevent the market players from always having correct economic signals. In fact, these signals are in force when the electricity price reflects in each network point the cost of producing and carrying that quantity of energy, from the injection bus to the withdrawal one, plus the cost of associated losses and possible congestions. And when these are present, they influence investments on the system by driving its expansion towards the highest possible performance increase. But in a Uniform Pricing (UP) scheme, this only happens when no congestion occurs inside the power network. In that situation, all the Local Marginal Prices ($LMPs$) are equal to each other, and thus with the uniform price that hence represents the correct economic signal anywhere on the network. Otherwise, whether a transmission line's power flow clashes with its capacity constraint, system's nodal prices generally differ and so the unique uniform price starts giving misleading economic signals to the market players in the various points of the network, which do not lead the system towards its performance highest possible improvement. For these reasons, as also stated by Egerer et al. in [54], the UP mechanism is the less efficient one and thus must be avoided if possible.

On the opposite extreme from the performance point of view, there is the Nodal Pricing (NP) mechanism firstly proposed by Schweppe et al. in 1988 [39]. This mechanism uses $LMPs$ to price electricity inside the power network. The $LMPs$ can have a twofold

interpretation. In fact, on the analytical side they are the Lagrange multipliers associated to the equality constraints of the market clearing optimization problem focused on the nodal power balances. But more interestingly, from the physical viewpoint they represent the local value of energy, namely, the marginal cost of supplying the next kWh load to the specific system node. Within this marginal cost are included the cost of producing and carrying this energy, from the generation point to the consumption one, plus the losses-related cost and the one of avoidance of congestions arising from delivering it. Therefore, *LMPs* always embody the aforementioned correct economic signals, which address system's growth towards its performance improvement. In addition to this, also the congestion management inside a *NP* scheme reveals to be perfect, because the whole grid's security requirements are included inside the power network model of the nodal-based market clearing. Hence, if a congestion occurs within the system, the nodal prices diverge, so as to split locations that span congested lines into higher and lower price zones. Consequently, the dispatching coming out from this market is always feasible, and does never require any costly readjustments. All the possible congestions are recognized and automatically alleviated by the nodal-based market clearing, as mentioned above. For these reasons, the *NP* mechanism is worldwide recognized as the most performing way to price electricity [9], as it embeds both a free-cost congestions management and the clearest and most objective possible economic signals. Therefore, it should not be surprising that it has been adopted in many countries or areas, such as Argentina, Chile, Russia, New Zealand, PJM (Pennsylvania, New Jersey, Maryland) and New York in USA [39]. But despite this, it also includes many drawbacks. In fact, firstly transmission networks are usually very large and complex systems made up of thousands of nodes. Consequently, adopting a *NP* scheme would often mean facing a too high computational burden [40]. Secondly, the implementation of a nodal-based market requires the establishment of an Independent System Operator (*ISO*) able to combine the role of market operator with at least part of the grid operation. But this latter is not still available in some areas, like European countries [9]. Thirdly, too small Bidding Areas (*BAs*), like single-node ones, might be inside a *NP* mechanism and are typically unacceptable due to the market power that may arise inside them. Market power that would end up threatening the power network's perfect competition, the starting point for nodal system's benchmark performance. For these reasons, a network partition seems to be the most reasonable choice in power market operation.

In fact, the remaining pricing mechanism is the Zonal Pricing (*ZP*) one introduced in 1999 [19]. This can be thought of as a compromise between the simplicity of a uniform structure and the accuracy of a nodal one, since it introduces differentiation of prices but between power network zones made up of several nodes [22]. And in particular, it can become a very good compromise configuration as stated by Burstedde in [4]. He demonstrates in fact that, if the adopted zonal configuration is optimally defined, the performance loss faced by grouping system's nodes inside *BAs*, when passing from the initial benchmark NP scheme to the *ZP* one, is nearly negligible. This is because, on the one hand the less there is *LMP* variance inside the various price zones, the more the respective zonal prices retain the major part of *LMPs*' benchmark economic signals able to address the system's expansion towards its performance highest possible improvement.

On the other hand, the more there are only inter-zonal congestions instead of intra-zonal ones, the more they can be identified by the zonal-based market clearing of the *ZP* scheme. In fact, the *ZP* scheme automatically alleviates them by making zonal prices diverge, without the need of any manual and hence costly readjustment by the *TSO*. Therefore, not all the zonal configurations are the same. The more they are optimal, the more they theoretically approach nodal configurations' benchmark performance without acquiring their drawbacks. On the contrary, the more they are inappropriate, the more they create a series of market inefficiencies and arbitrage opportunities similar to *UP* mechanism's ones.

For these reasons, it ineluctably arises the question of how to define an optimal zonal configuration. To which the European Union (*EU*) has tried to response in 2011, with the publication of Framework Guidelines on Capacity Allocation and Congestion Management for Electricity (*CACM*) made by the Agency for the Cooperation of Energy Regulators (*ACER*) [18]. Since in the last years, as proven by the scientific literature, the *ZP* mechanism has progressively gained popularity across Europe. Meanwhile, the continuously growing production capacity of renewables has increased fluctuating infeed. As the lines' transmission capacities have not been upgraded simultaneously, this has resulted in the increase of system's congestions and thereby of related costly readjustments too. Nowadays, all the European power networks are based on *UP* schemes or non-optimal *ZP* ones, where consequently also intra-zonal congestions can occur, which are not able to perform a free-cost congestions management. As a result, in Germany for instance the redispatch costs caused by congestions alleviation have more than tripled from 2012 to 2015 [9]. Then, in the last years the *EU* has actually become more and more interested in finding a way to define optimal zonal configurations. In this way, optimal *ZP* mechanisms can be used to quickly reduce congestions management's costs, in order to face better the recent congestions increase, while waiting for the still necessary transmission lines upgrade. However, in this scenario the aforementioned European guidelines have not given strict rules to create optimal *BAs*, but have only defined the general features which they should have. Therefore, the target of finding a method to deterministically define an optimal zonal configuration remains still open, and becomes the subject of this thesis.

1.2 How to define an optimal zonal configuration

There are mainly two ways of defining zonal configurations. On the one hand, there is the geographical clustering. It consists in creating *BAs* by cutting the power network along its statistically most congestible lines, identified through historical data or future scenario simulations. The *BAs* creation attempts having only inter-zonal congestions inside the resulting zonal configuration, in order to not impede this latter to become the aspired optimal one. On the other hand, there is the clustering-based approach, which creates price zones through a two-stage procedure. They firstly assign characteristic values to each node of the system, and then create *BAs* by applying clustering algorithms to the just defined nodal parameters database.

The so-called geographical clustering is not an actual clustering technique, because it creates the price zones by directly acting on the power network structure according

to a precise criterion, instead of applying data mining techniques on databases made up of nodal features. Therefore, here the word "clustering" refers only to the final result of the methodology, which is indeed the union of system nodes inside geographical areas representing *BAs*. Secondly, this approach actually produces the zonal configurations of the power networks which nowadays adopt a *ZP* mechanism. In fact, there are several instances of these last around the world. The Scandinavian electricity market named "Nord Pool" already established a market framework with multiple *BAs* in the nineties [8] and currently is composed of different price zones belonging to various countries of northern Europe [55]. The Italian market introduced *BAs* in 2006 due to its power grid's heterogeneous nature, and thus today is made up of six geographical price zones, which also contain other virtual zones. Although, this second instance is a bit particular. In fact, in case of congestion there is only a zonal price divergence on the supply side, namely for generators, whereas the pricing scheme for the demand one, namely for loads, is always uniform thanks to the unique national price called "Prezzo Unico Nazionale" (*PUN*). The *PUN* has been introduced not to influence the energy consumption within the nation through a geographically dependent price, and is obtained as the weighted average of all the zonal prices, where the weights are given by the energy quantities consumed in the various zones. But, beyond this particularity, it is important to note that all the currently existing zonal configurations derive either from *TSOs*' experience on the statistically most congestible lines of their power networks, when dealing with within-national *BAs*, or from national borders, when dealing with inter-national *BAs* [39]. Therefore, for the former case it is obvious to get the link between the aforementioned geographical clustering and the nowadays zonal configurations. And for the second one it can quickly be proven, by observing that over the last years there has been a steady rise in the amount of cross border trades. But, simultaneously there has been a very little growth in the cross border transmission capacities [17], which has ineluctably caused the rise of frequency of trans-boundary transmission lines congestions. This is the reason why the geographical clustering is actually able to produce also this second type of currently existing *BAs*. Despite its simplicity, this first method of zonal configurations definition has to be rejected to carry out the initial search for a deterministic way to set up optimal zonal configurations, for two reasons. On the one hand, this first approach reveals to be the representative of the nowadays *BAs*. Thus, it has to be rejected because, as proven by Burstedde and Breuer respectively in [4] and [5], optimal zonal configurations markedly differ from the currently existing ones. On the other hand, this geographical clustering is actually used as partitioning method inside [53], where it also gives good results. But there, the split power network is radially connected and hence it is completely different from the transmission networks, which instead are never truly radial and are the object of the here analysed optimal *BAs* definition.

For these reasons, the only remaining way to fulfill the thesis' goal resides in using clustering algorithms. These last belong to the data mining topic, and in particular are processes of Knowledge Discovery in Databases (*KDD*). They provide the user with more concise visions of big databases, which otherwise could not be so easily handled. In other words, these algorithms are firstly fed with big sets of data. Then, based on observations' features and a distance metric specified by the user, they merge these data into a series

of clusters, whose number is usually user-defined. Therefore, the final result is a lighter representation of the initial database. Only its most distinguishing observations according to the user-defined specifications, which hence depend on the user's interest towards the database, are revealed as clusters' centroids, which instead include all the others as similar data. Among all the clustering algorithms, two of the most used types are the connectivity-based clustering algorithms and the centroid-based ones [4]. Both of them rely on a user-defined distance metric, which can be a classical Euclidean distance or something else. The connectivity-based clustering algorithms evaluate distances between all the couples of database's observations, and then identify clusters in a hierarchical process which can be agglomerative or divisive. Namely, they can proceed bottom-up towards distance decrease among clusters, or top-down towards distance increase among them. The centroid-based algorithms immediately define clusters' centers and then measure the distances between them and the database's observations, so as to put these last inside the clusters whose centroid is the nearest one. Nevertheless, when using clustering algorithms to attempt defining power network's optimal zonal configurations, the aforementioned classification of clustering techniques reveals to be not always effective. Instead, it becomes clearer distinguishing them on the basis of the nodal parameter chosen by the user to perform the clustering process. Therefore, always dealing with the most diffused approaches, from this second point of view the clustering algorithms are divided into *LMPs-based* and *PTDFs-based*. The rationale of the first group is straightforward for a twofold possible interpretation. On the one hand in fact, the *NP* scheme is the most performing one. And thus, when merging nodes to define *BAs* in order to move to an optimal *ZP* mechanism, *LMPs* can surely be at the base of the clustering process which defines the adopted optimal zonal configuration, to lose as little as possible of their associated benchmark economic signals. On the other hand, *LMPs* start diverging when congestions occur within the power network. Especially, they do it by separating system's areas which span congested lines into higher and lower price zones, which respectively call for additional generation or load to alleviate the associated congestion. Therefore, merging nodes on the basis of *LMP* similarity is likely to create a zonal configuration free of intra-zonal congestions, because characterized by the congestible lines as inter-zonal links, which may become the aspired optimal one. Regarding the second aforementioned group, namely the *PTDFs-based* clustering algorithms, the reasoning is slightly longer and hence subsequently provided inside *Chapter 3* with the whole methodology.

1.3 Thesis' backbone

Given this problem, namely the definition of an optimal zonal configuration, and the way to solve it, namely the application of a clustering algorithm to system's nodes, this thesis firstly takes into account a group of broadly known clustering techniques. Then it modifies them, trying to better fulfill the zonal configurations' optimality requirements. Eventually, it evaluates their performance, through the application on a real power network model and the use of specific assessment criteria.

Therefore, the remainder of this thesis is organized as follows. *Chapter 2* presents the state of the art of this topic. Consequently, a series of tables and bulleted lists are there

proposed to summarize the main aspects of the reference articles which both recognize the need for a *BAs* redefinition and decide to apply clustering techniques to fulfill it, as here done. *Chapter 3* outlines the developed methodology. First of all, the zonal configurations' optimality requirements are clearly stated, by merging *CACM*'s general guidelines with the findings of previous state of the art's reference articles. Then, the apparently most suitable clustering algorithms are listed, together with their needed inputs, to run the methods, and their relative changes, aimed at better satisfying the aforementioned optimality requirements. Eventually, a series of zonal configurations' assessment criteria, both in terms of clustering validity indicators and economic efficiency ones, is provided. In order to check the optimality of the newly defined price zones, and also to allow the comparison among the different techniques which have produced them. The so obtained methodology is then applied to a real power network model made up of 257 buses, reduced version of the European transmission grid, inside *Chapter 4*. In this way, the algorithms' performance is compared with varying numbers of *BAs*, since this latter is unfortunately a parameter which always has to be directly or indirectly defined by the user. The comparison aims at electing the reasonably most suitable clustering algorithm to carry out this thesis' goal, represented by the deterministic definition of an optimal zonal configuration. Eventually, conclusions and future developments suggestions are contained in *Chapter 5*.

Chapter 2

State of the art

This chapter contains the detailed overview of this thesis' references which both take over the current power networks' need of undertaking optimal zonal configurations and use clustering algorithms to attempt defining a technique to deterministically find them. This overview is organized with both tables and bulleted lists, in order to ease the consultation to the reader.

Within *Appendix A* it is proposed a state of the art analysis similar to the following one, but in that case the same tables and bulleted lists deal with this thesis' references which adopt clustering techniques for purposes different from price zones definition.

2.1 References summary

The following bulleted list contains a short description for each of the considered papers which deal both with clustering algorithms and optimal *BAs* definition. It is organized with two subpoints for each reference, which respectively contain: the reason why optimal zonal configurations are investigated inside it and its general description.

- **[3] Breuer et al. (2013)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets Key Performance Indicator (*KPI*). Here the accent for zones creation is put on their static efficiency, namely the efficiency of the adopted zonal configuration along years. This is important since the temporal stability of the zones is one of the European criterion for an optimal zonal configuration. For this reason, a multi-scenario analysis is here carried out.
  - **Paper's summary:** This paper is aimed at presenting an approach to determine optimal bidding areas in the European electricity system under the perspective of static bidding area efficiency. In fact, it is worth remembering that one of the rule published into the framework guidelines on *CACM* is just the temporal stability with respect to uncertainties in the system (i.e. grid extension or development of *RES*). Therefore, to ensure a consideration of the

desired inter-temporal stability of *BAs*, a multiple-period and scenario-based method is here introduced and applied to a case study on a full European grid model with all 400 kV substations. The target function of the *BAs* clustering problem results from the basic principle of aggregating substations with similar nodal prices to one bidding area. This comes from the fact that *LMPs* are the clearest and most objective economic signals, thereby able to provide an optimal congestion management. Thus, the target function minimizes for each hour the absolute difference between nodal prices and the average prices of the zones in which the respective nodes are contained, like a sort of Ward's minimum variance criterion, that minimizes the total within-clusters variance. The algorithm used for the solution of the aforementioned problem, and hence for the zones creation, is a genetic one. With its genetic operators, like crossover or mutation, and its heuristic functions, used to guarantee compliance with the constraints. These last are both physical and operative bonds of the network and some of the requirements for a good zonal configuration, like the minimum zones dimension to prevent the rise of market power and the physical connection between nodes inside the same zone to avoid the definition of unfeasible zonal configurations. The clustering inputs are the *LMPs* of each node of the system, as hourly pattern along one year. Many years are considered during the multi-scenario analysis, with different patterns for each node. Zonal configurations are represented by gene strings inside the algorithm, where each node is associated to one of the created zones. The genetic algorithm (*GA*) arrives at convergence, and hence stops, when the objective function is minimized under a certain user-defined threshold. The number of zones has to be given beforehand by the user.

- **[4] Burstedde (2012)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** The research presented in this paper develops a set of *BAs* for the Central Western European (*CWE*) regions Switzerland and Austria, for both 2015 and 2020. On the basis of hourly *LMPs* patterns along year for system nodes, used as clustering input, and a hierarchical cluster analysis, driven by a Ward's minimum variance criterion. The zonal configuration coming out from the 2020's *LMPs* dataset is used to validate the 2015's partitioning from a temporal stability point of view, since this latter is one of the key aspect for an optimal zonal configuration according to European disposals. Once done this, the older zonal configuration, namely the 2015's one, is applied and used to compare the efficiency of a zonal configuration with the one of a nodal configuration. This latter represents the benchmark for system performance, since nodal prices are the clearest and most objective economic signals that can be used to price energy in the power network. While, from the zonal configuration's point of view, three cases are considered. Respectively, a partitioning

made up of six areas and another of nine, used to evaluate the system performance sensitivity with respect to the number of zones. The reference is the current zonal configuration, with zones' boundaries coincident with national borders. This comparison is done from an economic point of view, with the evaluation of costs originating from the wholesale market dispatch and from ex-post redispatch. The comparison of these annual costs shows small deviations between nodal and zonal models, and a little dependence of total costs to the number of zones. These results seem to suggest the non-profitability of an optimized zonal configuration instead of the current one, nationally based. But this is only due to the many costs neglected in this paper. Otherwise an optimized zonal configuration would be markedly better than a non-optimized one, and a nodal scheme would decidedly perform even better than these previous two.

- **[5] Breuer & Moser (2014)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*. This is the sequel of the previous reference [3]. In fact, while this latter only presents a clustering method based on a *GA* without giving any kind of zonal configurations evaluation. This paper starts from zones obtained through the aforementioned *GA*, and provides the assessment criteria which were previously missing.
  - **Paper's summary:** This paper provides the assessment criteria for the zonal configurations coming out from previous reference [3], which were missing there. Therefore, starting from the European indications for an optimal set of zones included inside the Network Code on *CACM*, four evaluation criteria are applied within this paper. These last are categorized two each in monetizable criteria and hardly monetizable ones. On the one hand, the first two are generation costs and redispatch costs, which respectively represent the cost of the wholesale market dispatch and the cost of congestions alleviation. On the other hand, the last two are the potential of market power and the violation of network security. Both of these last have not directly a meaning of cost, consequently they cannot be simply minimized like previous ones but they have acceptable levels, namely, thresholds which define if the associated evaluation criterion is satisfied or not. The potential of market power is avoided until the Residual Supply Index (*RSI*) is under a certain critical level, while the violation of network security is prevented until the congestion energy remains under a certain critical level. This latter represents the violations of security-constraints after remedial actions are taken for congestion alleviation. It is computed via the sum of non-transferable electricity due to security limits. The four aforementioned assessment criteria are used to compare some zonal configurations of the European simplified power network. Which are obtained through the clustering algorithm presented in paper [3]. The most peculiar features coming from this evaluation are:

  - Low numbers of *BAs* violate the assessment criterion about network security.
  - High numbers of *BAs* lead to a potential of market power because of too small areas, violating the respective assessment criterion.
  - Once verified the two hardly monetizable assessment criteria, the two remaining monetizable ones reveal a slight decrease of total system costs with respect to the current zonal configuration. Unfortunately, these savings are firstly negligible compared to the typical total system costs. Then, they even decrease when keeping a zonal configuration for more than three or four years, which is likely to be done since temporal stability is an important feature for system zones according to *CACM*'s guidelines.

- **[8] Felling & Weber (2016)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper is complementary to reference [9], indeed they share the authors. This article's peculiarity resides in the clustering algorithm description. Here in fact, there is a clear flow chart which describes the whole algorithm into third section point C. This latter is a hierarchical clustering fed with hourly *LMPs* patterns along a year, that determines an optimal price zone configuration which minimizes the total within-clusters variance. Therefore, here again the Ward's minimum variance criterion is used as distance metric. Particularity of this algorithm resides in the weighting of nodes according to their infeed and demand situation. The more they are energetically relevant, the more their weight makes them considerable during the clustering process. In this way price zones are defined with sufficient supply and demand relevance and diversity or, in other words, there is an incentive to aggregate smaller zones rather than larger ones at similar price differences. This leads to *BAs* of similar dimensions that prevent the birth of isolated and too small zones, unwanted by *CACM*'s guidelines due to their possible market power. Moreover, three important things are stated inside this paper. Firstly, as evident from its figure three, the within-clusters variance and the between-cluster one are respectively inversely and directly proportional to the number of zones. This was predictable by the fact that their sum gives the dataset variance per definition. This latter is constant once given the group of data, for any number of zones. But here this is proven by the facts. Secondly, as evident from this paper's figure four, not correcting outlier prices from *LMPs* data set leads to the definition of *BAs* which have: mainly the same shape of the ones defined with corrected prices, but anyway some differences. That is why it is still better to correct the initial database, by eliminating obviously unrealistic prices related to the preliminary status of the grid and load models. Hence as first thing to do, prices under 0 €/MWh and above a certain value chosen as maximum, are respectively set to 0 €/MWh and the maximum. Thirdly,

tests on the algorithm's computational burden show a great performance of this procedure in handling very large systems as the power networks.

- **[9] Felling & Weber (2018)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*. Here much attention is laid on the temporal stability of zonal configuration performance. Therefore, a robust price zones configuration is computed taking into account six 2020's scenarios.
  - **Paper's summary:** This paper presents a hierarchical cluster algorithm which defines new *BAs* by grouping similar hourly *LMPs* patterns along a year. Moreover, two things are done in this paper. On the one hand, nodes are weighted depending on their energy relevance. Therefore, the more they withdraw or inject from or into the power network the more their weight makes them relevant during the clustering process. In this way, it is avoided the birth of too small *BAs*. As requested by *CACM*'s guidelines for an optimal zonal configuration, since these zones could be characterized by the rise of market power. On the other hand, six scenarios are considered for a single year (2020). Which respectively reflect the main drivers that influence the future development of European Electricity markets according to the trilemma of energy policy target. Namely security of supply, sustainability and economic efficiency. Each of these scenarios returns a set of hourly *LMPs* pattern along the aforementioned year, through a Direct Current Optimal Power Flow (*DCOPF*) in which transmission lines' capacity is curtailed of 15% in order to roughly satisfy the N-1 security. Therefore, each of these databases is used as input in the hierarchical clustering and so produces a sequence of zonal configurations for the particular scenario. In fact, hierarchical clustering algorithms do not need the number of groups beforehand. But meanwhile they do not return a single partitioning rather than a sequence of configurations. In addition to this, a robust configuration is determined by simultaneously inserting the six *LMPs* sets to the algorithm. In this way, each node is characterized by six trends coming from the respective hourly *LMPs* patterns along the 2020 in different scenario cases. The result is a zonal configuration that outperforms other single scenario configurations, and particularly also the current *BAs* in *CWE*. This latter configurations assessment is done looking at the within-clusters variance of each of them. The less this within-clusters variance is, the better the *BAs* are. Since their zonal prices approach as much as possible the nodal prices of the system, and hence a minimum part of these reference economic signals are lost. Particularly interesting is the mixed evaluation of zonal configurations. Namely the assessment of within-clusters variance, and so the *BAs* performance, using a different set of *LMPs* from the one used as input into the hierarchical clustering that produced the in question zonal configuration. This is an important analysis since the *BAs* choice has to be done nowadays but it has to be correct in the future too, because of stability criterion requested by *CACM*'s guidelines,

and there is no certainty in the future. Interestingly the aforementioned robust configuration outperforms all the others also in this latter mixed assessment.

- **[15] Grimm et al. (2017)**[1]
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper concerns with splitting a market area into a given number of price zones such that the resulting market design yields good social welfare results. This leads to a mixed-integer nonlinear trilevel model for computing welfare-optimal price zones in electricity markets. For problems of this kind no general-purpose solution algorithms exist. Consequently, this article proposes two different global solution approaches. One is based on the reduction of levels using problem-specific insights as well as standard Karush Kuhn Tucker (*KKT*) transformation. The other one is a problem-specific instantiation of generalized Benders decomposition. The computational results show that this latter significantly outperforms the former algorithm. It is worth remembering that here *BAs* are created, but not by clustering nodal features like *LMPs* or Power Transfer Distribution Factors (*PTDFs*). The optimal zonal configuration comes out from the iterative simulation of the energy market with different zonal schemes, as the one which maximizes the most the social welfare. That is also why this *BAs* definition method has a too much high computational burden, and then it is discarded first. In fact, testing all the possible zonal configurations of the power network one after the other is enormously time-consuming and requires a huge computational effort. Therefore, it can only be done in very small power networks, thus unrealistics.

- **[17] Imran & Bialek (2008)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper analyses the effectiveness of the zonal congestion management scheme on three zonal configurations, by using a model of the first synchronous electricity grid region of Europe. The clustering algorithms which respectively create the three aforementioned zones sets are: a geographical clustering, a fuzzy-c-means and a price differential clustering. The first creates the areas in a way that the statistically more congested lines are at the zones interfaces. This is done because one of the main criteria for forming good

[1] Actually this paper does not include a clustering algorithm aimed at defining an optimal zonal configuration. Nevertheless, it is here included due to its purpose, which is always the search of optimal *BAs*.

BAs is actually the elimination of intra-zonal congestions in favour of inter-zonal ones. Since only these last are automatically alleviated by a zonal based market, through the zonal prices differentiation, while intra-zonal congestions need a manual redispatch by the *TSO* with associated extra costs, that would be avoided in a better zonal configuration. The second and the third clustering algorithms both use a set of hourly *LMPs* pattern along a year as input. The fuzzy-c-means clusters them according to Ward's minimum variance criterion, since the objective function which drives the Optimization Problem (*OP*) is roughly aimed at minimizing the within-clusters variance. Whereas the price differential clustering merges them according to their difference. Therefore, the more two *LMPs* trends are similar, the more probably these two nodes will be in the same zone. The hourly *LMPs* come out from a *DCOPF* of the aforementioned European power network model, for the year 2004. The effectiveness of the zonal congestion management scheme on the three zonal configurations, and hence their assessment, is evaluated looking at the maximum range and standard deviation of within-clusters *LMPs*. This because: the more the nodal prices differ from the zonal ones that approach them, the more their economic signals aimed at congestions alleviations get lost, so the congestion management worsens. For this reason, a good zonal configuration from the zonal congestion management scheme's point of view must have small standard deviation and range of prices in the zones. This does not happen with the geographical clustering zones, thus discarded, but it happens with the other two clustering algorithms. Unfortunately, since no check at all is present on physical connection between the merged nodes, these last two zonal configurations have firstly to be modified by dividing the detached zones into distinct areas. And this leads to the definition of too small *BAs*, unacceptable because the market power which could rise there. To summarize, the final result of this paper is the almost impossible optimal *BAs* creation in the European market. And if these zones are actually formed, they may create market inefficiencies caused by arbitrage possibilities.

- **[18] Jakubek et al. (2015)**
  - **Paper's rationale:** To show the possible inadequacy of the LMPs-based *BAs*. By means of a constructive example it is proven that the division obtained from clustering of *LMPs* in some cases may not place the congested lines on the zones' borders. This is a con since it requires an additional costly readjustment manually done by the *TSO*, after that the market coupling mechanism has already found the supply/demand equilibrium of the zonal system.
  - **Paper's summary:** This paper asserts that *LMPs* clustering is one of the most diffused approach aimed to *BAs* redefinition. And it seems to be reasonable, since grouping *LMPs* should assign nodes that span a congested line into two different clusters allowing then the market coupling to govern this congestion as inter-zonal one. In fact, it is worth remembering that only these last can be seen and hence automatically alleviated by a zonal-based energy

market. The intra-zonal congestions cannot be seen by this latter, and then they have to be manually alleviated by the *TSO* through a costly readjustment of the market's dispatching. However, by means of a constructive example, it is here proven that *BAs* configuration obtained from *LMP* methodology might not always be efficient. Since this approach may lead to zones identified not on basis of congestion of transmission lines, but on the differences in nodal prices arising from other reasons. Therefore, these widely diffused clustering methods should be used with care and compared to alternative approaches (like the PTDFs-based ones). The LMPs-based clustering algorithm here used to create the example is a hierarchical clustering algorithm based on Ward's minimum variance criterion, fed with hourly *LMPs* patterns along a year and modified to keep the zones internally connected from the physical point of view.

- **[19] Kang et al. (2013)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper shows a *ZP* mechanism based on sequential network partition and congestion contribution identification. The first of these two passages consists of dividing a *BA* into two parts each time there is a congested line inside it, defining an intra-zonal congestion. While the second one does the aforementioned split of the targeted zone by clustering the its nodes' *PTDFs* coefficients of the same sign. Thereby the aforementioned congested line is placed exactly on the border between the two newly defined zones. Ready to define an inter-zonal congestion, instead of the previous intra-zonal one that cannot be seen and hence automatically alleviated by a zonal-based market. And consequently would require a costly and manual readjustment of the dispatching by the *TSO*. Which would mean an additional cost for the system and thus an efficiency decrease of the energy market. *PTDFs* are considered congestion contribution factors since they can be used to reflect the congestion contributions of system nodes to the congested lines. In fact, *PTDFs* coefficients are expression of transmission lines' power fluxes sensitivity with respect to the system nodes' injection of power. Consequently, a positive *PTDF* of a node towards a congested line reveals an increase of the line's power flux deriving from the increase of the generation injected by the node. Therefore, it indicates an injection of power which aggravates the congestion of the specific line. And the other way around negative *PTDFs*. Despite its computational burden, this clustering algorithm is sure an innovative way of *BAs* defining. Its main drawback is only represented by the consideration of a one-sided pool, which is unrealistic in many nowadays energy markets. And dealing with a two-sided pool would prevent the user to make the Node With/WithOut Generators (*NWGs/NWOGs*) distinction done in the paper. Which would even increase the already high computational burden.

- **[21] Kiran et al. (2017)**
  - **Paper's rationale:** To create new *BAs*, so to improve the energy market efficiency from different points of view like the congestion management or the social welfare, and meanwhile to automatically find the optimal number of zones.
  - **Paper's summary:** This paper provides an approach to create *BAs* for their application in congestion management zonal schemes, while simultaneously providing an answer to the question asking for the optimal number of zones which could partition a power network. This is done in two steps. In the first one the network is repeatedly split into multiple zones, on the basis of hourly *LMPs* patterns put into a classic genetic clustering algorithm, creating a set of zonal configurations. It is worthwhile to remember that nodal prices are commonly used as clustering feature for the *BAs* definition, since they inherently capture the impact of congestion on every node by providing an economic signal which aims towards the congestion alleviation itself. Once these zonal configurations are obtained, they are fed in a cooperative-game-based decision-making process for the identification of the optimal zones number that uses the primal-dual linear programming model of the linear bottleneck games. The efficacy of the proposed clustering algorithm is shown on a six-bus system, a *IEEE* 39-bus system, and a 193-bus practical Indian system. This methodology may help regulator or policy maker in deciding the number of *BAs* to be formed.

- **[22] Klos et al. (2014)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper proposes a PTDFs-based clustering approach called "BubbleClust algorithm". It defines the new *BAs* by grouping power network's nodes in a multidimensional space, created by *PTDF* matrix, by using coefficients related to power flows over the congested lines. The reason of this energy market zonal division method stands inside satisfying both the economic and system stability criteria. Therefore, either the control of inter-zonal congestions in a transparent manner and the minimization of intra-zonal congestions' additional costs or the accuracy of prediction of flows on the critical and frequently congested lines are all optimized. The effectiveness of the methodology is tested on an example of New England *IEEE* 39 bus system. This reveals quite good overall results, although it is ought to make two important remarks. On the one hand, the estimation of social welfare would be more accurate if redispatching costs were included. And this is rather important, since the social surplus is one of the criteria used to assess the effectiveness of the newly defined *BAs*. On the other hand, the overall social surplus gain here obtained is anyway small if juxtaposed with other parameters' orders of magnitude. Though, the 39 bus system here considered may constitute a not complex

enough space to illustrate all the potential benefits of this clustering algorithm. Therefore, further analysis should be necessary to get a clearer evaluation of this drawback. Eventually, as this paper's merit it is worth remembering that *PTDFs* are smartly weighted using lines' congestion rate factors. In order to be sure of finding the most congestible lines as inter-zonal connections when clustering *PTDFs*. So as to produce only inter-zonal congestions rather than the intra-zonal ones, which are unpleasant in zonal-based market since they represent an additional cost. Due to their alleviation, which has to be manually done by the *TSO* and hence is costly. Congestion rate factors are obtained as average of the transmission line's average congestion cost over the sum of all the other transmission lines' average congestion costs. Each of these average congestion costs is the arithmetic average of several *KKT* multipliers, namely Lagrange multipliers, all associated to the line's maximum power flux constraint and respectively deriving from several runs of *DCOPF* in different load and generation scenarios.

- **[25] Marinho et al. (2017)**
  - **Paper's rationale:** To assess zonal configurations' optimality through a newly defined index named Redispatch Effort. It provides the order of magnitude of costly actions which have to be done when the dispatching resulting from the zonal market clearing based on a certain user defined *BAs* configuration is likely to create congestions. Therefore, the less it is, the better. Because no costly actions are needed, as for the reference congestion management scheme of nodal-based energy markets.
  - **Paper's summary:** This paper proposes an indicator called Redispatch Effort (*RE*). It provides a quantitative measure of the level of congestion resulting from a zonal market clearing and requires no arbitrary choice of sensitive parameters, hence it is totally objective. In other words, this metric represents the order of magnitude of the costly actions, like changing the schedule of the operational units such as power plants, which have to be done when the dispatching resulting from the zonal market clearing based on a certain user defined *BAs* configuration is likely to create congestions. Therefore, this parameter can effectively be used as *BAs* assessment criterion. And the more a zonal configuration has a low *RE*, the more is a performing optimal zonal configuration that well approaches the benchmark congestion management of a nodal-based market. Which has no RE for definition, because all its power network's congestions are automatically alleviated by its nodal market clearing, through the *LMPs* differentiation. The aforementioned indicator is computed this way. Firstly, it is run a constrained OP which does the nodal market clearing of the system. Secondly, as a result of the previous nodal dispatching, it is assessed the net position, namely the difference between export and import, for each candidate *BA* of the evaluated zonal configuration. Thirdly, a new constrained *OP* as nodal market clearing is run for each of them. Paying attention to always substitute the inequality constraints on the power fluxes

of the *BA*'s transmission lines with the equality constraint on the *BA*'s overall power balance, set equal to the *BA*'s net position previously computed. This has to be done in order to simulate the zonal-based markets' behaviour of considering the inner parts of *BAs* as copper plates, namely power network portions without any physical limits. Eventually, once done the third step for all the *BAs* of the zonal configuration under evaluation, the *RE* index is evaluated as absolute and relative difference of the zonal dispatching's decisional variables respect to the nodal dispatching's ones treated as benchmark. This newly defined index for *BAs* assessment is then applied inside the paper on two operational problems. On the one hand it is used to judge the zonal configurations coming out from three LMPs-based clustering algorithms. Which are respectively a hierarchical clustering, a K-means clustering and a K-medoids clustering. On the other one it is used to assess the profitability of splitting or merging the current Western Europe's *BAs*. Operation done respectively by using a K-means clustering or a hierarchical one. From this latter point of view, no particular advantage is observed. Since the RE index almost remains constant by varying the number of the existing *BAs*. Instead, concerning the first application of this indicator, it is interesting to note that for a single period (namely a clustering made on a *LMPs* snapshot taken during a winter peak) the K-means outperforms the hierarchical and K-medoids approaches with a markedly lower *RE*. Whereas, for multiple periods (namely a clustering made on hourly *LMPs* patterns along a year) the hierarchical approach proves to be more effective. It is worth remembering that most of these considerations are captured with few *BAs*, namely twenty or less zones. But this is perfect, since optimal zonal configurations must always have the least possible number of zones to actually preserve their optimality. Because increasing the zones number would make the system tend to a nodal-based market. And consequently its performance would obviously increase, since nodal configurations have the best performance per definition. But meanwhile this would not be acceptable for many reasons, starting from the possibility of market power that can arise in nodal configurations, hence a zonal configuration of compromise is necessary. And to be so, that configuration has to minimize the number of zones not to acquire the reasons which prevent the usage of a nodal scheme.

- **[27] Jang et al. (2005)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper proposes an improved fuzzy-c-means approach for the efficient zone clustering of Large-scale power systems, namely for defining an optimal *BAs* configuration. The adjective "improved" refers to the internal modification that here has been made on a classical fuzzy-c-means scheme, in order to automatically prevent the algorithm from defining unfeasible zonal configurations made up of zones with geometrically distant nodes put inside

the same cluster. This condition was usually obtained in other papers through an external check. Which, during the merge of a node in a cluster, controlled the presence of at least one physical connection between them. And then, in case of absence of it, it stopped the union. Here instead, this external check is no longer necessary because implemented in the distance metric used by the algorithm. Thereby, similarity between nodes and clusters centers is here evaluated by the combination of a Euclidean distance between *LMPs* snapshots, used as clustering feature, and a geometric distance between nodes spatial positions, aimed at only merging physically linked nodes and clusters. Of course, from the centroids' point of view the spatial positions are obtained as average of the nodes' spatial positions inside the respective clusters. A real application of this clustering algorithm is shown in real Korea power network, using different weighting coefficients for the price distance and the geometric one inside the newly defined similarity measure. So to observe eventual changes in the clustering.

- **[31] Zhang et al. (2008)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper creates a combinatorial optimization model to represent the power-grid-partitioning problem. And then solves it through a heuristic algorithm embedded by tabu search. This latter is a meta-heuristic approach used inside many optimization problems. Which aims to prevent the algorithm from assessing again already considered solutions, namely from cycling, by memorizing the attributes of recently visited solutions in a tabu list and forbidding them for a certain number of iterations. In this way the tabu search speeds up the process. The remaining part of the method is a heuristic clustering algorithm, named "imitating out-point method" because of its similarity with the traditional out-point method. The nodal features used for the clustering are single values of *LMPs*, both taken as averages of a pattern or snapshots of certain circumstances. The main constraints enclosed inside the clustering deal with the physical connection and the price closeness among within-clusters nodes, the zones number limitation and the single-node *BAs* prohibition. The first two of these constraints are respectively addressed to prevent the algorithm from defining unfeasible zonal configurations made up of physically detached *BAs*, and to preserve the benchmark economic signals of *LMPs* when passing from *NP* to *ZP*. The third of the aforementioned constraints keeps the resultant zonal configuration away from the number of zones of an unfeasible nodal configuration. The last impedes the possible rise of market power that may occur in single-node areas. Both the model and the algorithm to solve it are tested on some real examples from northeaster power grid of China. The outcomes prove this method to be actually applicable for the *BAs* definition problem.

- **[32] Van den Bergh et al. (2016)**
  - **Paper's rationale:** To quantify the impact of the number of *BAs* on the market outcomes of a zonal configuration. To do this: both a hierarchical clustering algorithm, to define *BAs*, and an economic dispatch model for zonal-based markets, to compute market outcomes, are developed.
  - **Paper's summary:** This paper defines new *BAs* through a hierarchical clustering algorithm, based on Ward's minimum variance criterion and fed with nodal *PTDFs* of statistically most congestible lines. These last are logarithmically weighted using congestion rate factors, already used for that and described in previous reference [22]. In this way, the most constraining transmission lines are put between zones, ready to define inter-zonal congestions instead of intra-zonal ones. Which are remembered to be unpleasant in zonal-based market due to their alleviation, required to be manually done by the *TSO* and hence costly. The usage of a logarithmic weighting, in disagreement with the previous reference [22], relies on the fact that proportional weighting can attribute a disproportionately high importance to the most congested lines. The clustering algorithm also includes the classical additional constraint on *BAs*' contiguity during the merging of clusters, so as to prevent the method from defining unfeasible zonal configurations made up of physically detached *BAs*. To fulfil the paper's initial purpose, namely to quantify the impact of the number of *BAs* on the market outcomes of a zonal configuration, the aforementioned process is used to define several *BAs* sets of a portion of the European power network. These zonal configurations with varying zones number are then assessed through a comparison with both a reference nodal configuration, representing the benchmark in terms of market performance, and the current zonal configuration of European power network, representing the starting point. This comparison is done from an economic perspective. Therefore, an economic dispatch model for zonal-based markets is developed inside this paper too in order to define the market outcomes of each zonal configuration. According to the results, these last have a marginal improvement that decreases with increasing number of *BAs*. Hence there may be a suggested number of zones after which the additional improvement brought by an additional *BA* would be overcome by the additional costs deriving from the zonal configuration application. In fact, these last are here proven to be directly proportional to the number of zones. And this was predictable, since one of nodal configuration's drawbacks which hinder its enforcement is the too high maintenance cost which would be associated to a such complex system. For all these reasons, this paper's final suggestion is that it is not always profitable to increase the number of zones. A careful analysis is needed.

- **[36] Wawrzyniak et al. (2013)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*. The clustering algorithm

here proposed puts the focus on the temporal stability of the newly defined *BAs*. Concerning with the weather conditions variability, which affects system's *LMPs* and hence the optimal zonal configuration definition too. Because this latter can be based on the aforementioned prices, like happens in this clustering method.

- **Paper's summary:** This paper firstly points out an important lack in the literature of the time concerning the power network division into zones: the use of usually stable levels of generation. Which remains in contradiction with the increasing amount of renewable generation for which, as yet, wind farms, characterized by highly variable power output, constitute the main source. This is a problem because, the relative instability in the amount of power injected into the system by wind farms significantly influences the energy prices even if the rate of wind generation to total generation is relatively small. Hence, LMPs-based clustering algorithms for the *BAs* definition would better to take into account several wind scenarios with relative *LMPs* sets inside their processes. So as to improve the optimal zonal configuration's temporal stability with respect to the weather conditions variability. For these reasons, this paper starts from a hierarchical clustering algorithm based on Ward's minimum variance criterion and fed with hourly *LMPs* patterns along a year. Then it runs the process using 722 different historical wind scenarios, leading to different wind farms generations, *LMPs* sets and thus zonal configurations. And eventually it aggregates all the clustering results using another clustering technique, named "consensus clustering", through which the final optimal zonal configuration is found. Therefore, this latter additional clustering algorithm is the real innovation of this paper. Thanks to which many power network's future scenarios can effectively be considered during the *BAs* definition, in order to improve the temporal stability of the resulting zonal configuration as requested by *CACM*'s guidelines for optimal *BAs* sets.

- **[39] Yang (2004)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*. And to deterministically find the optimal number of zones. By using three assessment criteria on zonal configuration's price zones. Which are respectively *BAs*' lifetime, compactness and isolation.
  - **Paper's summary:** Firstly, the paper explains why LMPs-based clustering algorithm may be less efficient than the ones based on *PTDFs* of statistically most congestible lines. This is mainly due to the temporal stability of the final optimal configuration, which is ineluctably lower in LMPs-based configurations due to their clustering features' dependence on time. For this reason, it is here developed a new PTDFs-based clustering algorithm to create power network's *BAs*. Inside it, before creating price zones, all the potentially congested lines are firstly determined in a period of time based on actual operating conditions

of the system. This is done through a Monte Carlo simulation, thanks to which it is analysed the congestion probability of transmission network also considering uncertainty in the power market. After that, the sensitivities of nodal power injections to power flows on just located congested lines, namely *PTDFs* to them referred, are computed for all the system's nodes. And then, these nodal features are used to place nodes inside a high-dimensional space whose dimension is the number of congested lines. Forming this way an input dataset for the here used clustering algorithm, which is a scale-space hierarchical clustering. This latter in fact, is able to group points inside a space by simulating the human visual system. Therefore, in this case it creates power network's *BAs* by merging nodes represented by points inside the aforementioned high-dimensional space, respectively located by using as coordinates each node's *PTDFs* of most congestible lines. After having described this new clustering algorithm, this paper analyses the clusters validity problem too. In order to deterministically choose the optimal number of zones among the sequence of zonal configurations, with different number and size of *BAs*, enclosed within the resulting dendrogram. Therefore, price zones' lifetime, compactness and isolation are introduced inside this second part as parameters useful for the user to make his choice on the number of zones of the final zonal configuration. Which has to be done here like in any other hierarchical clustering algorithm. Eventually, this methodology is applied to two congestion cases to prove its effectiveness. These last are both taken from *IEEE* 118-node system, and respectively consider two or four congested lines.

- **[40] Yang et al. (2005)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*. And to deterministically find the optimal number of zones. By using three assessment criteria on zonal configuration's price zones. Which are respectively *BAs*' lifetime, compactness and isolation.
  - **Paper's summary:** This paper is highly similar to previous reference [39]. This is reasonable since these two documents share some of the authors. Anyway, there are slight changes here following. Firstly, previous reference [39] has been published on a journal named "Periodica Polytechnica". Whereas the current article comes from a conference. Therefore, it is obvious for the former one to propose a more in-depth analysis of the newly described PTDFs-based scale-space hierarchical clustering. Secondly, previous reference [39] finds the most congestible lines through a Monte Carlo simulation, thanks to which it is analysed the congestion probability of transmission network also considering uncertainty in the power market. Whereas this second paper identifies these lines by using classic historical data on power system's operation. This second difference could be due to the easier nature of this second document. Beyond these slight changes, all the rest remains the same. Even the final case study

for the actual application of the clustering algorithm, which is here run again on *IEEE* 118-node system.

- **[41] Yang & Zhou (2006)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** After the introduction, this paper explains why LMPs-based clustering algorithm may be less efficient than the ones based on *PTDFs* of statistically most congestible lines. This is mainly due to the temporal stability of the final optimal configuration, which is ineluctably lower in LMPs-based configurations due to their clustering features' dependence on time. For this reason, this paper defines new power network's *BAs* by using a fuzzy-c-means clustering algorithm fed with nodal *PTDFs* of statistically most congestible lines. These last are identifies through a Monte Carlo simulation method before the run of the aforementioned clustering process, so as to analyze the congestion probability of transmission network by considering the uncertainties in the power market too. Whereas at its end, each system node is uniquely assigned to the cluster to which it has the highest grade of membership. In fact, it is worth remembering that peculiarity of fuzzy-c-means clustering algorithm stands actually inside the possibility of assigning each point to more than one cluster by using its grades of membership. But this is not useful for *BAs* definition. Because in this latter case each point, namely each system node, must belong to only one cluster, namely one price zone, in order to create an applicable zonal configuration. This is why it is necessary the previous placement passage, where it is also embedded a control on within-clusters nodes' physical connection, so as to prevent the clustering algorithm from defining unfeasible zonal configuration made up of physically detached *BAs*. Eventually, this complete method is applied on the *IEEE* 118-system to show its effectiveness. The results seem to reveal a reasonable power network partitioning. Even if no specific assessment criteria are here proposed.

- **[42] Yong et al. (2000)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper describes the technical challenges in implementing a cluster-based congestion management system. Therefore, to do so it compares the performance of the two main representatives of this latter category to the one of the nodal pricing method. Which results in the most efficient operating point possible at any given instance, and hence represents the benchmark in terms of performance. The two cluster-based congestion management systems here analyzed are respectively the "zonal pricing" method and the "congestion-cluster pricing" one. The former one determines the zones based

on price differentials in *LMPs*, whereas the last one merges nodes into *BAs* based on their relative impacts of power injection on congested transmission lines. This last quote could make think to nodal *PTDFs* of statistically most congestible lines. But instead, in this paper they are replaced by Congestion Distribution Factors (*CDFs*). Which anyway are derived from aforementioned *PTDFs*. The nodes' position information is obviously included inside the *BAs* definition of both the two previous zonal-based markets. So as to prevent the clustering algorithm from defining an unfeasible zonal configuration from the physical point of view. The systems performance are evaluated, in order to make the aforementioned comparison, through a zonal market clearing simulation of a very simple 9-bus power network. Therefore, no clustering algorithm for the *BAs* definition is actually employed inside the two zonal-based markets. And their zonal configurations are simply defined by grouping nodes according to the similarity of their *LMPs*, or *CDFs*. The final results in terms of congestion management performance obviously elect the nodal scheme as the benchmark. Followed by the congestion-cluster pricing method, namely the zonal configuration based on *CDFs*, and then by the zonal pricing method, namely the zonal configuration based on nodal *LMPs*.

- **[44] Yao et al. (2016)**
  - **Paper's rationale:** To create new *BAs* in order to improve energy market efficiency from different points of view. Like the congestion management or the social welfare, typically used as markets *KPI*.
  - **Paper's summary:** This paper defines power network's price zones by using an improved K-means clustering algorithm fed with hourly *LMPs* patterns along a certain period, in particular one or three months. The two improved points compared with traditional K-means clustering algorithm reside in considering system's topology, by looking at the existence of physical connections among within-clusters' nodes, and not choosing randomly the initial centroids. The former betterment is included by checking, every time a node is going to be put inside a cluster, the presence of at least one physical connection between it and the nodes which are already inside the cluster. This measure aims to prevent the clustering algorithm from defining unfeasible zonal configurations, made up of physically detached *BAs*. Instead, the second amelioration tries to fix the problem for which different runs of a K-means clustering algorithm on the same input database can lead to completely different partitions. Since the choice of the initial clusters' centers hugely affects the final result of the centroid-based clustering algorithms, like the K-means is. This is a problem during *BAs* definition, because the perfect methodology should be able to deterministically find out the global optimum of the clustering problem. Namely the best applicable zonal configuration of the power network in question. But this does not always happen whether the clustering result depends on some user-defined input data, like the aforementioned centroids are. Because, in this latter case a wrong choice of them makes the clustering *OP* converge on

a local optimum, instead of the desired global one. For this reason, here the K-means' initial centroids are not randomly chosen by the user. But they are automatically indicated by the algorithm by maximizing the geographical distance between them. So as to also cover as much as possible the power network's area, in order to minimize the number of failures of the previous check on nodes' physical connection. Therefore, here the user has only to define the number of clusters like in any other K-means clustering algorithm. And then, it is the algorithm which chooses the initial centroids among the nodes of the input database. By indicating its group of nodes which maximizes, for that user-defined number, the geographical distance between them. Beyond these two improvements, the rest of the K-means clustering algorithm proceeds as usual. Therefore, the Ward's minimum variance criterion drives the points sorting inside the various clusters during the process. And the clustering algorithm stops when the user-defined convergence criterion is met. Which is here represented as usual by the non-variance of clusters configuration for two successive iterations. The aforementioned methodology is eventually applied to two case studies, so as to assess its effectiveness. Therefore, both the *IEEE* 118-bus system and a realistic regional power system not better defined are partitioned using this clustering algorithm. Respectively using hourly *LMPs* patterns along three months or along one month as clustering input. In these real cases the number of zones of the final zonal configuration is chosen by looking at the maximum among the maximum differences of *LMPs* respectively evaluated in each *BA*. The lower this parameter is, the better. Because it means that few of the *LMPs*' benchmark economic signals have been lost when passing from nodal pricing to zonal one. Therefore, this index is obviously inversely proportional to the number of zones. Since having a number of *BAs* equal to the number of nodes would give no loss of nodal economic signals. But meanwhile, this latter situation is not acceptable due to the inapplicability which characterizes nodal pricing schemes for many reasons. Consequently, a trade-off zonal configuration is needed. And hence it is chosen by looking when the marginal decrease of the aforementioned parameter begins to lessen less. In other words, here the number of zones is set where one more BA does not make the above *LMPs* dispersion diminish so much to consider profitable the zonal configuration change.

2.2 Clustering features summary table

Table 2.1 portrays an overview of the user-defined nodal parameters which are used inside each of the considered papers as clustering feature, to run the respective clustering algorithms.

Table 2.1: Clustering features summary table of references which deal with optimal *BAs* definition using clustering algorithms.

	LMP hourly patterns along a period of time	**LMP snapshot**	**CDFs**	**PTDFs of most congestible lines**	**PTDFs of most congestible lines weighted through congestion rate factors**
[3] Breuer et al. (2013)	X				
[4] Burstedde (2012)	X				
[5] Breuer & Moser (2014)	X				
[8] Felling & Weber (2016)	X				
[9] Felling & Weber (2018)	X				
[15] Grimm et al. (2017)[a]					
[17] Imran & Bialek (2008)	X				
[18] Jakubek et al. (2015)	X				
[19] Kang et al. (2013)				X	
[21] Kiran et al. (2017)	X				
[22] Klos et al. (2014)					X
[25] Marinho et al. (2017)	X	X			
[27] Jang et al. (2005)		X			
[31] Zhang et al. (2008)		X			
[32] Van den Bergh et al. (2016)					X
[36] Wawrzyniak et al. (2013)	X				
[39] Yang (2004)				X	
[40] Yang et al. (2005)				X	
[41] Yang & Zhou (2006)				X	
[42] Yong et al. (2000)		X	X		
[44] Yao et al. (2016)	X				

[a] Actually this paper does not include a clustering algorithm aimed at defining an optimal zonal configuration. Nevertheless, it is here included due to its purpose, which is always the search of optimal *BAs*.

2.3 Clustering techniques summary table and descriptions

Table 2.2 classifies the considered papers with respect to the adopted clustering algorithms. This latter is followed by a bulleted list, which contains two things for each clustering algorithm: its general description and the specific working processes which have been undertaken of it during its various applications, inside the papers included in this chapter.

Table 2.2: Clustering algorithms summary table of references which deal with optimal *BAs* definition using clustering algorithms.

	BubbleClust Algorithm	Fuzzy-c-means	Genetic Algorithm	Geographical Clustering	Hierarchical Clustering	K-means	K-medoids	SNP With CCI	Price Differential Clustering	Scale-Space HC	Consensus Clustering (CSP)	IOP Method With TS
[3] Breuer et al. (2013)			X									
[4] Burstedde (2012)					X							
[5] Breuer & Moser (2014)			X									
[8] Felling & Weber (2016)					X							
[9] Felling & Weber (2018)					X							
[15] Grimm et al. (2017)[2]												
[17] Imran & Bialek (2008)		X		X					X			
[18] Jakubek et al. (2015)					X							
[19] Kang et al. (2013)								X				
[21] Kiran et al. (2017)			X									
[22] Klos et al. (2014)	X											
[25] Marinho et al. (2017)					X	X	X					
[27] Jang et al. (2005)		X										
[31] Zhang et al. (2008)												X
[32] Van den Bergh et al. (2016)					X							
[36] Wawrzyniak et al. (2013)					X						X	
[39] Yang (2004)										X		
[40] Yang et al. (2005)										X		
[41] Yang & Zhou (2006)		X										
[42] Yong et al. (2000)									X			
[44] Yao et al. (2016)						X						

- **BubbleClust Algorithm:** It is a space-based clustering algorithm based on the so-called *PTDF* space. This latter is built this way: (a) take the *PTDF* matrix, namely a *MxN* matrix where M represents the number of lines and N the number of nodes. Each element inside it reveals the power flux contribution which results on the M-th line, indicated by the row, from the injection of 1 additional MW by the N-th node, pointed by the column. (b) These matrix columns are treated as vectors of coordinates in a M-dimensional space. Therefore, each of these vectors both corresponds to the space position of a certain node, from the overall point of view, and to the fractions of power transmitted through the system lines deriving from the injection 1 additional MW by that node, from the single elements' point of view. Having introduced *PTDFs*, it is worth remembering that they always imply a withdrawal of energy from the slack bus. Namely they indicate the sensitivity of transmission lines' power fluxes respect to the injection of power from a system node and the contemporary download from the slack bus of the same amount. And this latter reference bus has to be arbitrarily decided beforehand, causing a *PTDFs*' dependence on this choice. All this could become a drawback for PTDFs-based clustering methods. But fortunately, as proven inside the appendix of reference [22], the slack bus selection never affects the clustering result even changing the *PTDFs* on which the algorithm is based. Therefore, PTDFs-based clustering approaches become at first sight as reasonable as LMPs-based ones. (c) Going on with the *PTDF* space creation, since zones borders want to be defined along most congestible lines in order to only have inter-zonal congestions in the resulting partitioned system instead of intra-zonal ones. Which are unwanted in zonal-based energy markets due to their manual, and hence costly, alleviation. All the vectors of coordinates, namely the *PTDF* matrix's columns, are scaled through the respective congestion rate factors. These last are one per transmission line and are obtained as average of the transmission line's average congestion cost over the sum of all the other transmission lines' average congestion costs. Each of these average congestion costs is the arithmetic average of several *KKT* multipliers, namely Lagrange multipliers, all associated to the line's maximum power flux constraint and respectively deriving from several runs of *DCOPF* in different load and generation scenarios. And moreover, these average congestion costs are so called for two reasons. On the one hand, they are "averages" since they come from the average of several Lagrange multipliers associated to different runs of the *DCOPF* algorithm. On the other one, they are also "congestion costs". Since each Lagrange multiplier firstly represents the cost of the physical constraint referred to him, and this latter could become a congestion cost in case of activation of the respective constraint. (d) After having scaled *PTDF* matrix's columns with the aforementioned congestion rate factors, the nodes that embrace congested lines are spatially divided by the others. This is because congested lines are scaled through higher congestion rate factors, and

[2]Actually this paper does not include a clustering algorithm aimed at defining an optimal zonal configuration. Nevertheless, it is here included due to its purpose, which is always the search of optimal *BAs*.

hence obtain lower scaled vectors of coordinates. Whereas the nodes hugging rarely congested lines are scaled through lower congestion rate factors, and hence obtain higher scaled vectors of coordinates. (e) This split perfectly permits to shrink the dimension of the problem. Because all the nodes with high coordinates are those at the ends of rarely congested lines, which can be neglected since the method's purpose is to define zones' borders along most congestible lines. Therefore, only nodes at the ends of most congestible lines remain, namely those with smaller coordinates. The aforementioned *PTDF* space has been found. At this point, it is interesting to note that the remaining nodes are most distant when they are the couple of extremes of one among the most congestible lines. Consequently, applying now the BubbleClust clustering is perfect. Because it does not put these couples of nodes into the same clusters but, as initially wanted, it makes these couples of nodes become the edges of two adjoining *BAs*. In order to effectively obtain a zonal configuration with power network's most congestible lines along zones' borders.

  - **[22] Working process:** (a) In the above described *PTDF* space, initial single-node zones are firstly considered in correspondence of the remained nodes, which are most congestible lines' extremes. Therefore, if the previously observed most congestible lines are K, at this point there are $2K$ single-node zones or less. Since there can be cases of nodes at the edge of two or more usually congested lines. (b) Then it is entered a loop, where firstly they are evaluated the Euclidean distances between the just evaluated centroids and the not yet merged nodes, namely the ones that have been previously neglected during the *PTDF* space creation. Then it is located the smallest of them, and the respective node and cluster are merged together after having checked the presence of at least one physical connection between them. In order to prevent the algorithm from creating unfeasible zonal configurations made up of *BAs* containing spatially detached groups of nodes. And eventually the cluster' center of the modified cluster, namely its so-called centroid, is updated also taking in consideration the coordinates of the last added node. (c) This loop continues until no more unattributed nodes are present inside the *PTDF* space, and there are only $2K$ clusters which derive from the as many initial single-node zones. (d) At this point there are two possibilities. On the one hand, if the user's preference on the number of clusters has already been overcome by the just reached $2K$ groups, the algorithm stops and issues the zonal configuration requested by the user. On the other hand, if it is not, the algorithm continues merging in each step the two closest and physically adjacent clusters. Until the aforementioned user's preference on the number of clusters is finally reached, and thus the respective zonal configuration is issued.

- **Fuzzy-c-means:** The whole fuzzy clustering methods are soft clustering algorithms. They differ from hard clustering ones, like the K-means, because they can assign each point to more than one cluster through a grade of membership. The Fuzzy-c-means is the most used among fuzzy clustering methods. It is very similar to the K-means algorithm except for the result, which is here the typical "matrix of

memberships" of fuzzy algorithms instead of K-means' data clusters. In the matrix of memberships: each row represents a datum while each column represents a cluster. Thereby, each cell indicates the grade of membership of a particular point towards a specific cluster from zero to one. Moreover, it is worth remembering that these grades of memberships cannot be negative and have unit sum for each row. Since each datum can only have an overall unitary membership, divided among all the clusters.

  - **[17] Working process:** (a) Choose the number of clusters k, like K-means clustering algorithm. (b) Randomly or manually generate k cluster centers, also called centroids. (c) Assign the grades of membership of each point towards each cluster, through the objective function used to define the fuzzy membership. Which represents the distance metric inside a fuzzy clustering method, instead of the more classic Euclidean one used within K-means algorithm. (d) Recalculate the centroids as clusters' averages, like a K-means clustering, but weighted through the clusters elements' grades of membership. (e) Repeat the two previous steps until some convergence criterion is met. Which is usually the not variance of clusters composition between two following iterations, just like K-means algorithm. Finally, two variations are done inside this paper. On the one hand a proper zonal configuration is eventually created by assigning each node to only one cluster, when its grade of membership to a cluster is greater or equal than 0.95. In this way, each node cannot belong to more than one zone at the end of the clustering and so the zonal configuration becomes actually applicable. On the other hand, only physically linked nodes are merged into the same zone thanks to a check run at the end of the clustering process. This latter verifies the physical connection between nodes inside the same cluster. And in case modify the zonal configuration to guarantee it. In order to preserve the physical feasibility of the resulting zonal configuration.
  - **[27] Working process:** The same of previous reference [17], apart from the distance metric here used to iteratively merge nodes and clusters at each step. In fact, inside this paper it is used the improved fuzzy membership instead of the classical one. Which combines, through two user defined coefficients with unitary sum, the information of *LMP* similarity, normally used as clustering feature, and the one of nodes spatial position, essential to prevent the clustering algorithm from defining physically unfeasible zonal configurations. Namely *BAs* sets made up of physically detached zones.
  - **[41] Working process:** The same of previous reference [17], apart from the nodal features chosen as input for the clustering algorithm. Which are here the nodal *PTDFs* of statistically most congestible lines, identified through a Monte Carlo simulation, instead of the previous hourly *LMPs* patterns along a year.

- **Genetic Algorithm:** The *GA* is not a usual clustering algorithm, since it does not belong to none of the two prominent families of clustering algorithms: the centroid-based algorithms and the connectivity-based ones. Anyway, here it is used to solve

the optimization problem which defines the clustering and hence states the new *BAs*. This latter is the second optimization problem of the paper. Since there is first a classic *DCOPF* aimed at finding out the *LMPs* of the system nodes, then used as input for the *GA*. The objective function which drives the clustering process points to minimize for each hour the absolute difference between nodal prices and the average prices of the zones in which the respective nodes are contained. Like a sort of Ward's minimum variance criterion, that minimizes the total within-clusters variance.

  - **[3, 5, 21] Working process:** (a) It is randomly created a starting population made up of I chromosomes. Where each of them is a solution, namely a zonal configuration. This is because each of them is a code of N numbers, where N is the number of system nodes, in which each number represents the cluster to which a specific node belongs. The clusters, that is the zones, are M overall and this number has to be defined beforehand by the user. (b) The starting population enters a loop. Where genetic operators, like mutation and crossover, modify the chromosomes' genes and hence create a new population. This latter is a new group of chromosomes, namely a new set of zonal configurations, which better fit the objective function which regulates the clustering. Otherwise the new chromosomes coming out from the aforementioned loop would have been discarded, in favor of the initial population which entered the loop at the beginning of the iteration. (c) The loop stops, and so the *GA* comes to a convergence giving an optimized zonal configuration, when the objective function reaches the target within a certain user defined tolerance. Namely in this case, when it is minimized under a certain threshold indicated by the user. (d) *OP*'s constraints are considered both trying to change some chromosomes' genes at the end of each iteration, so to fix the specific outlaw solution, or using penalty terms in the target function, which move the objective function away from the optimization and hence oblige the algorithm to go ahead looking for new zonal configurations.

- **Geographical Clustering:** The geographical clustering creates zones by dividing the power network along statistically most congestible transmission lines. Thereby, in the resulting zonal configuration congested lines are at the interface or boundaries of the zones. And congestions can only occur between zones, as inter-zonal congestions, rather than inside them, as intra-zonal ones. This would be an optimal condition for zonal configurations, since they are only able to automatically alleviate inter-zonal congestions. While intra-zonal ones cannot be seen by the pool of the zonal-based market, and hence they have to be manually alleviated by the *TSO*. With associated extra costs. At this point it is worth remembering that, as stated in the introduction of paper [17], the nowadays European power network is bounded by intercontinental transmission lines since over the years there has been a steady rise in the amount of cross border trade whereas there has been very little growth in the cross border transmission capacities. Therefore, the aforementioned geographical clustering algorithm which divides the European power network along

statistically most congestible transmission lines, actually partitions this network along intercontinental transmission lines. Namely along national borders. In fact, the zonal configuration coming out from this clustering algorithm divides the European power network into seventeen zones. Which exactly reflect the current zonal configuration based on a national reason. For all these reasons the geographical clustering is roughly considerable as the current zonal configuration of power networks, where it is present of course. Eventually, please note that this clustering algorithm has no need of any kind of distance metric. Since the clustering process is not driven by the similarity of some nodal features, but by the statistically most congestible transmission lines.

  - **[17] Working process:** (a) Find the statistically most congestible lines. (b) Use them as zones' borders. (c) Obtain the power network division along these lines. Which actually coincides with a national-based zonal configuration.

- **Hierarchical Clustering:** The algorithm here used for the creation of new *BAs* is a connectivity-based and bottom-up clustering, also called hierarchical and agglomerative clustering. This type of clustering methods firstly consider each data as an independent cluster. And then they iteratively merge two clusters into a single one at each step, until all the initial points are contained into a sole group. All these algorithms proceed towards the increase at each step of the distance between clustered data. But meanwhile they differ for the linkage criterion, used at each clustering step to merge the couples of points. In this paper it is used the Ward's minimum variance criterion, according to which clusters are created by minimizing the total within-clusters variance. In other words, groups are here created by joining data as similar as possible.

  - **[4, 8, 18] Working process:** (a) Each data is considered as an independent cluster, as stated by bottom-up and hierarchical clustering algorithms. (b) At the beginning of each step it is computed the sum of squared Euclidean distances. They are one per node, and respectively calculated between the price vector of each node (which contains the hourly values of its *LMP*) and the average price vector of the cluster to which the considered node belongs. Considering the sum of Euclidean distances coming from the nodes contained into a sole cluster, it gives a measure of its homogeneity. Therefore, it is zero at the beginning of the algorithm when each cluster is made up of just one element and hence its homogeneity is total. And it grows during the clustering process, where clusters start including stranger points and so their homogeneity decreases. (c) The linkage criterion used within this hierarchical clustering is the Ward's minimum variance criterion. Since it tends to minimize the total within-clusters variance, an objective function is created using the aforementioned sum of squared Euclidean distances and it is minimized. (d) In this way, at each step it is merged the couple of clusters which increases the objective function, and so clusters' homogeneity, as little as possible. Thereby, clusters are actually created according to Ward's minimum variance criterion. (d) The

aforementioned *OP* is also constrained to avoid the physically incoherent situation of merging detached nodes into a single *BA*, namely a single cluster. This is done through a Boolean variable, namely a binary parameter, equal to one when the two nodes are adjacent and equal to zero otherwise. In this latter case, the joining of the couple of nodes inside the same zone becomes forbidden. (e) The merging of couples of nodes, and hence the clustering process, goes on until all the initial points are contained into a sole group. In that moment, the summary dendrogram is produced and one of the optimized zonal configurations is chosen by the user. Among the sequence provided by the hierarchical clustering. This choice becomes the final result of the clustering algorithm.

- **[9] Working process:** (a) Six scenarios of the future *CWE* power network are considered, useful since there is no certainty in the future. (b) Each of these scenarios is solved through a *DCOPF* on an annual basis, with transmission lines' capacity curtailment to roughly verify the N-1 security, providing hourly *LMPs* patterns along the year for that scenario. (c) The six resulting *LMPs* sets are inserted into the clustering algorithm, both individually and together. In the hierarchical clustering algorithm: (c.1) at the beginning each node corresponds to one zone. (c.2) Then at each iteration the two zones which increase the less the *OP* objective function, namely the within-clusters variance with the nodes' weights, are merged together. (c.3) In this way, a sequence of zonal configurations is created. Which ends with the creation of a single cluster and the issue of the dendrogram. (d) As a result, seven zonal configurations sequences are produced: one for each scenario and the seventh for all the scenarios together, hence called the "robust" configuration. In which the user has then to choose the number of zones by cutting the dendrogram.
- **[25] Working process:** The same as previous reference [4], fed with both hourly *LMPs* patterns along a year and *LMPs* snapshots.
- **[32] Working process:** The same as previous reference [4], fed with nodal *PTDFs* of statistically most congestible lines. Evaluated through a *DCOPF* of the European power network portion used inside the case study of the paper. This latter covers a full year on an hourly basis, and it is based on national load data for the year 2013 taken from *ENTSO-E*.
- **[36] Working process:** Basically the same as previous reference [4]. The innovative part of this paper is represented by the final consensus clustering run to put together the 722 zonal configurations, respectively associated to the considered wind scenarios. The resulting optimal zonal configuration is characterized by a better temporal stability towards weather conditions variability than all the aforementioned zonal configurations only linked to one wind scenario. This consensus clustering algorithm belongs to the problem known as aggregation of clustering. In this latter there are a number of different clustering results that have been obtained from different runs of the same clustering method. And the task of the algorithm is to create a single (consensus) clustering which generalizes the results of a whole set of runs. The consensus clustering works as following described. (a) If two objects are in the same

cluster then they are considered to be fully similar, and if not they are marked dissimilar. Similarity between two objects takes the value of one if they are in the same cluster and zero otherwise. (b) Consequently a binary similarity matrix can be created for each clustering result, namely for each wind scenario in the paper's case study. (c) The entry-wise average, namely the average independently performed on each matrix element, of such matrices representing the sets of groupings yields an overall similarity matrix. (d) Starting from highest value of this latter overall similarity matrix, and progressively going down in descending order, couples of nodes are iteratively merged into the same BAs until the user-defined number of clusters is achieved. Please note that, if the additional check on within-clusters' nodes physical connection is already enclosed in the previous hierarchical clustering from which derive the aforementioned 722 zonal configurations, it may not be necessary to include it also in this consensus clustering. Because the bigger elements of the overall similarity matrix are automatically referred to often merged nodes, which have had to satisfy many times the just described additional check. Nevertheless, it is suggested to include this control inside the final consensus clustering too. So as to be sure of preventing the overall method from defining an unfeasible zonal configuration made up of physically detached BAs.

- **K-means:** The K-means clustering is a centroid-based clustering algorithm. Its name comes from considering each of the K clusters as average of his inner points. These clusters representative values become the so-called cluster centres or centroids. Which can actually correspond to one of the respective cluster's points, like at the beginning when there is just one datum per cluster, or not. These K centroids have to be initialized by the user before the start of the clustering to values that can belong or not to the input database, hence the number of clusters is a strong input for this algorithm. After that, the process iteratively merges each of the observations to the cluster with the nearest centroid and then updates these cluster centers at the end of each step. The algorithm stops when a user defined converge criterion is met, which is typically the non-change of clusters composition among a certain number of consecutive steps. Therefore, the final outcome of a K-means clustering is the partitioning of N observations coming from an input database into K clusters in which each observation belongs to the cluster with the nearest centroid. Since the number of clusters is an essential input for this algorithm, and since it is usually a not available information beforehand, measures of clustering adequacy like the Clustering Dispersion Indicator (CDI) or Mean Index Adequacy (MIA) are often used to suggest this value. A good clustering tool must both exalt the difference between points belonging to different clusters, and make the points inside the same cluster as similar as possible. This is more reached the more the previous indices tend to zero. Consequently, analysis on their trends according to the number of clusters are typically made. And the number of partitions is eventually chosen when these indices start remaining almost constant. Obviously, they are both inversely proportional to the number of clusters. Since increasing it to the number of database's observations, as extreme condition, would make a perfect clustering

according to the previous disposals.

  - **[25] Working process:** (a) Choose the number of clusters K. (b) Manually or randomly initialize the respective K centroids. (c) Assign each dataset point to the cluster with the nearest centroid according to the distance metric used by the algorithm, which is usually a normal Euclidean one. (d) Update the clusters centers by also taking in consideration the newly inserted points. (e) Iteratively repeat the two previous steps until some convergence criterion is met, which is usually that the clusters composition has not been changed between two following cycles. The whole of these passages is done for several number of clusters in the paper, in order to look at the RE index trend according to the number of zones. Hence no CDI or MIA evaluations are done to find out the advisable number of clusters for the input database in question.
  - **[44] Working process:** In this paper, new BAs are created using a topology based K-means clustering algorithm. This latter aggregates the buses with similar dynamic $LMPs$, namely hourly, into zones by iterations. The improvement over the classic K-means algorithm is twofold. On the one hand it is considered the existence of connection between buses in one zone before merging nodes in clusters. On the other one initial clusters centroids are not chosen randomly, thereby the randomness of the solution is avoided. The number of clusters has to be given beforehand, like any K-means algorithm. In this case this is found by comparing the nodal prices' losses of sufficient experiments, which have to be minimized.

- **K-medoids:** This clustering algorithm is quite similar to K-means one, except for the choice of clusters centroids. Which here have to correspond to one of the within-clusters' points. These medoids, so called to distinguish them from the previous K-means' centroids, are anyway user defined at the beginning of the process and then updated during the clustering by modifying clusters composition. In this latter passage, remembering medoids' feature to be one of the within-clusters' points, a new method for updating the medoids has to be invented. Since centroids' technique of making the average among within-cluster points may give a value different from any of them, and hence is not suitable anymore. Therefore, amongst the arbitrary methods to solve this problem, this algorithm application based on power network analysis chooses to define new medoids at each step by looking at nodes' connections: the more a node has a high number of electrical connections with other nodes, the more its feature used for the clustering is likely to become the medoid of its cluster. It is worth to remembering that within the rest of the clustering process, namely during the partition of the N database observations into the K user defined clusters, the distance metric used to put each point inside the cluster with the nearest medoid is always the same of the previous K-means, namely a classic Euclidean distance.

  - **[25] Working process:** It is basically a K-means clustering, hence the working process previously described in the above lines, with the aforementioned modification on clusters centres, which are medoids instead of centroids.

- **Sequential Network Partition (*SNP*) With Congestion Contribution Identification (*CCI*):** This is an innovative *BAs* partitioning based on two steps. The first consists of dividing a *BA* into two parts each time there is a congested line inside it, defining an intra-zonal congestion. While the second one does the aforementioned split of the targeted zone by clustering the its nodes' *PTDFs* coefficients of the same sign. Thereby the aforementioned congested line is exactly placed on the border between the two newly defined zones. Ready to define an inter-zonal congestion, instead of the previous intra-zonal one that cannot be seen and hence automatically alleviated by a zonal-based market. And consequently would require a costly and manual readjustment of the dispatching by the *TSO*. Which would mean an additional cost for the system and thus an efficiency decrease of the energy market.

  - **[19] Working process:** (a) Take the power network of interest and look it as a sole zone. (b) Considering a one-sided pool, namely an energy market with perfect competition only on the generators' side and an inelastic demand, system nodes can be classified into two categories: *NWGs* and *NWOGs*. And then, since just the ascriptions between *NWGs* can change zonal prices while changing the ascriptions of *NWOGs* does not have any impact on them, only ascriptions of *NWGs* are integrated as decision variables of the model which simulates the market clearing of the system. Given the fact that the number of *NWOGs* is approximately three or four times of *NWGS* in an actual network, this action largely cuts the complexity and the size of the problem. But meanwhile it becomes one of the main drawbacks of the algorithm when it is treated a more realistic two-sided pool. (c) Run a *OP* to make a numerical market clearing. Only considering the *NWGs* and neglecting the *NWOGs*, including both the physical and the operative power network constraints and using the minimization of generation costs as objective function. It is worth remembering that this latter action equals to maximize social surplus under the hypothesis of a unilateral market, like the one here considered. (d) The aforementioned *OP* defines the dispatching of the system. And hence states the generators' outputs and the zonal prices, if there were more than one zone which for now is not present. (e) Taking back the *NWOGs*, so to have a representation of the whole power network, and applying the above dispatching to the complete system. It is possible to check the existence of intra-zonal congestions. Which are unwanted inside zonal-based market, since can only see and hence automatically alleviate inter-zonal congestions. While intra-zonal ones ineluctably require a costly and manual readjustment of the dispatching by the *TSO* in order to be alleviated. (f) If these intra-zonal congestions are not present, then the zonal configuration is given out as clustering algorithm's final result. But if they are noticed, the most serious of them is treated as following: the area in which it is contained is appointed as "targeted zone" and the congested line is indicated as "targeted congested line". (g) Once had these two information, the algorithm enters the second phase. Where the just defined targeted zone is split in two parts by clustering the its nodes' *PTDFs* coefficients, towards

the targeted congested line, of the same sign. In order to exactly place this targeted congested line on the border between the two newly defined zones, ready to define an inter-zonal congestion instead of the previous intra-zonal one that cannot be seen by zonal-based markets. (h) At this point, the number of zones of the considered power network is increased of one. And then the algorithm goes back to step "b", giving to the market clearing the possibility to divide the system in one more zone. The process continues in loop until no more intra-zonal congestions are detected. In that moment the algorithm will stop and give the final zonal configuration.

- **Price Differential Clustering:** This clustering algorithm merges nodes according to their *LMPs* difference. Therefore, the more two *LMPs* trends are similar, the more probably the two respective nodes will be in the same zone. Actually, this price difference can also be either requested to be satisfied during all the hours of the *LMPs* trends or as average value on the whole trends. It depends on the specific algorithm.

  - **[17] Working process:** (a) Take the hourly *LMPs* pattern along a year as input data. (b) Use them to merge nodes. When the difference between the respective average *LMPs* falls below a certain value, i.e. 5% as stated into this paper. (c) Obtain power network' zones. (d) Check the physical connection between nodes inside the same cluster. And in case modify the zonal configuration to guarantee it. In order to define a feasible zonal configuration.
  - **[42] Working process:** Actually the same of previous reference [17]. But both fed with *LMPs* on the one side, and with *CDFs* on the other one. Where these last substitute the usage of nodal *PTDFs*.

- **Scale-Space Hierarchical Clustering (*HC*):** It is a clustering process that models human visual system. Therefore, as in the process of human perception the images perceived in the brain can be regarded as a set of light points in the space. Where by increasing the scale the image is gradually blurred merging each light point into smaller blobs and then into larger ones, until they are all contained into only one big light blob and so does the relative image. Here the clusters are made by progressively merging the input points placed in a space, whose dimensions correspond to the number of features chosen for the clustering. So as to use these last as points' coordinates. This way of acting recalls the scheme of a hierarchical clustering algorithm, since couples of points are iteratively joint at each step according to the distance metric adopted inside the method. Which is typically a multi-dimensional Euclidean distance between points' coordinates inside the aforementioned high-dimensional space. Therefore, it should not be surprising that the final result of this clustering process is a dendrogram again.

  - **[39, 40] Working process:** (a) The input dataset made up of N points, namely the N power network's nodes in this case, is placed in a space whose dimension correspond to the number of features chosen for the clustering. Therefore, these last are used as points' coordinates. (b) Each of these points can

be regarded as a light point, mathematically expressed by a Dirac delta function, which belongs to a bubble, namely a cluster, with a certain center. (c) During the first iteration, each light point belongs to a bubble whose center coincides with the light point itself. Hence N small bubbles can be seen inside the aforementioned high-dimensional space. (d) Then, going on with the iterations, these bubbles are progressively merged according to the distance metric adopted inside the algorithm and bubbles' centers are simultaneously updated. The distance metric here used is a classical multi-dimensional Euclidean distance between points' coordinates at the beginning, when there are single-point bubbles, or between the coordinates of bubbles' centers then, when the merging process has started. (e) The clustering algorithm stops when all the light points are contained inside a single big bubble. In that moment, due to this clustering approach's similarity with hierarchical clustering algorithms, a summary dendrogram is issued again. (d) Once this latter has been produced, *BAs*' lifetime, compactness and isolation can be used to assess clusters validity. So as to choose the zonal configuration to keep as optimal among the sequence of *BAs* sets provided by the dendrogram.

- **Consensus Clustering (CSP):** This kind of clustering algorithms are of a higher level. Thy take many clustering results and combine them in order to create a single (consensus) clustering result. They refer to the situation in which a number of different clustering results have been obtained either from different runs of the same clustering method in different scenarios or from different runs of different clustering methods working on the same dataset. Therefore, their task is to create a single (consensus) clustering which generalizes all the previous partial results. This thing can be done through different Consensus Clustering algorithms, here it is used one of the most famous ones which is the Cluster-based Similarity Partitioning (CSP) algorithm.

  - **[36] Working process:** Essentially (a) it takes the clustering results from previous partial runs and (b) it uses them to compose the so-called similarity matrix. This latter is a matrix with dataset data points along both the columns and the rows, so that it is symmetrical. Its elements are coefficients enclosed between 0 and 1 which express the number of times that those two data points have been clustered into the same group during previous partial runs. Having 1 means always, otherwise having 0 means never. (c) Hence in this situation the *CSP* algorithm starts from the higher elements of the similarity matrix and clusters respective data points, to continue then creating clusters with the same criteria.

- **Imitating Out-Point (*IOP*) Method With Tabu Search (*TS*):** This *BAs* definition method is based on four features which characterize the resultant zonal configuration. They are: the physical connection and the price closeness among within-clusters nodes, the zones number limitation and the single-node *BAs* prohibition. The first two of these conditions are respectively addressed to prevent the algorithm from defining unfeasible zonal configurations made up of physically

detached *BAs*, and to preserve the benchmark economic signals of *LMPs* when passing from *NP* to *ZP*. The third of the aforementioned characteristics keeps the resultant zonal configuration away from the number of zones of an unfeasible nodal configuration. While the last feature impedes the possible rise of market power that may occur in single-node areas. All these things are modeled as constraints inside the heuristic clustering algorithm, named imitating out-point method, whereas the taboo search is enclosed in the method too to speed up the process.

- **[31] Working process:** (a) Find the lower bound of the number of zones, namely the minimum number able to produce a feasible solution according to aforementioned clustering algorithm's constraints. (b) Initialize the power network partitioning with this number of *BAs*. (c) Use the taboo search to find alternative zonal configurations, able to meet the problem's constraints too while keeping fixed the *BAs* number. (d) If no alternative solution is found, then divide in two the zone with the maximum nodal prices dispersion and go back to step "c" finding alternative zonal configurations with one additional *BA*. (e) Once an alternative solution is met, the algorithm stops and issues it as optimal zonal configuration. The taboo search enclosed in step "c" is made up of five passages, which are following. (i) Coding: where the input feasible configuration from step "b" is encoded in a vector of numbers. Where each element assigns a node to a BA. (ii) Neighborhood structure: where alternative zonal configurations are searched, by moving one node at a time. (iii) Tabu list: where the previously made moves are recorded. In order to avoid cycling while finding alternative solutions. (iv) Aspiration criterion: where the alternative zonal configuration of passage (ii) is kept as best one until it maximizes the clustering algorithm's objective function. (v) Stop rule: which defines the stop of the taboo search after a certain number of moves, recorded on the tabu list.

2.4 Distance metrics summary table

Table 2.3 distinguishes the papers considered inside this chapter according to the similarity metric used inside their clustering algorithms. Afterwards, a quick description of these distance measures is provided inside a bulleted list, which contains also the reference to where the specific metric has been used.

Table 2.3: Distance metrics summary table of references which deal with optimal *BAs* definition using clustering algorithms.

	Multidimensional Euclidean Distance	**Monodimensional Euclidean Distance**	**Fuzzy Membership**	**Improved Fuzzy Membership**	**Connectivity Based Distance**
[3] Breuer et al. (2013)	X				
[4] Burstedde (2012)	X				
[5] Breuer & Moser (2014)	X				
[8] Felling & Weber (2016)	X				
[9] Felling & Weber (2018)	X				
[15] Grimm et al. (2017)[3]					
[17] Imran & Bialek (2008)		X	X		
[18] Jakubek et al. (2015)	X				
[19] Kang et al. (2013)		X			
[21] Kiran et al. (2017)	X				
[22] Klos et al. (2014)	X				
[25] Marinho et al. (2017)	X	X			X
[27] Jang et al. (2005)				X	
[31] Zhang et al. (2008)		X			
[32] Van den Bergh et al. (2016)	X				
[36] Wawrzyniak et al. (2013)	X				
[39] Yang (2004)	X				
[40] Yang et al. (2005)	X				
[41] Yang & Zhou (2006)			X		
[42] Yong et al. (2000)		X			
[44] Yao et al. (2016)	X				

- **Multidimensional Euclidean Distance:**
  - **Definition:** $E_{ij} = \sqrt{\sum_{v=1}^{Ndim}(x_{vi} - x_{vj})^2}$
  - **Applications:**
    - **[3, 4, 5, 8, 9, 18, 21, 25, 36, 44]:** Multidimensional Euclidean distance between hourly *LMPs* patterns of system nodes along years. It is computed between nodal prices and the respective average zonal prices coming from the *BAs* to which the nodes belong. It wants to be minimized, like a sort of Ward's minimum variance criterion. In fact, the clustering algorithms which admittedly use this latter linkage criterion as distance metric are here included too. Since the final result is always the minimization of within-clusters variance.
    - **[22]:** Multidimensional Euclidean distance between vectors' coordinates, one per node, which define nodes positions inside the so-called *PTDF* space. Created starting from the *PTDF* matrix.
    - **[32]:** Multidimensional Euclidean distance between sets of statistically most congestible lines' *PTDFs*, one per each node of the power network.
    - **[39, 40]:** Multidimensional Euclidean distance between vectors' coordinates, one per node, which define nodes positions inside the high-dimensional space created by using the features chosen for the clustering. Namely the nodal *PTDFs* of the most congestible lines.

- **Monodimensional Euclidean Distance:**
  - **Definition:** $E_{ij} =\mid x_i - x_j \mid$
  - **Applications:**
    - **[17]:** Monodimensional Euclidean distance between single values of average *LMPs*.
    - **[19]:** Monodimensional Euclidean distance between single values of nodal *PTDFs* relative to the "targeted congested line" and belonging to the nodes inside the "targeted zone".
    - **[25]:** Monodimensional Euclidean distance between single values of *LMPs* taken from a system snapshot of a winter load peak.
    - **[31]:** Monodimensional Euclidean distance between single values of *LMPs*. Obtained as averages or peak values of actual hourly trends.
    - **[42]:** Monodimensional Euclidean distance between single values of *LMPs*, obtained from time snapshots in certain power network conditions, or *CDFs*, which substitute the more typical use of nodal *PTDFs* of most congestible lines.

[3] Actually this paper does not include a clustering algorithm aimed at defining an optimal zonal configuration. Nevertheless, it is here included due to its purpose, which is always the search of optimal *BAs*.

- **Fuzzy Membership:**
  - **Definition:** The fuzzy membership is the distance metric used within fuzzy clustering algorithms. It classifies the input data in a unitary scale based on the possibility of being a member of a specified set. One is assigned to those points that are definitely a member of the specified set. While the entire range of possibilities from one to zero is used to express intermediate grades of membership of the point to the set. Of course the larger the number, the more the point is likely to belong to the specified set. The sum of all the fuzzy memberships of a point, respect to all the available clusters, must be one. Since any point's total membership must be unitary towards the database of belonging. In this paper, the above described fuzzy membership is evaluated among power network nodes' hourly *LMPs* patterns along a year.
  - **Applications:**
    - **[17]:** Here the fuzzy membership is evaluated between the hourly *LMPs* patterns along a year.
    - **[41]:** Here the fuzzy membership is evaluated between the nodal *PTDFs* of statistically most congestible lines, evaluated through a Monte Carlo simulation run before the clustering algorithm.
- **Improved Fuzzy Membership:**
  - **Definition:** Improved fuzzy membership. It derives from the previous classical fuzzy membership. Therefore, it equally classifies the input data in a unitary scale based on the possibility of being a member of a specified set. And thereby it gives to each database's point K grades of membership: which cannot be negative, have unitary sum, are directly proportional to the point's membership to the cluster and are respectively referred to the K clusters of the partitioning. But moreover, there is a change in how these grades of membership are assigned. In fact, while in the previous fuzzy membership they only depended on a simple Euclidean distance between the nodes' features and the centroids' ones, in this improved fuzzy membership they are based on a combination of the just cited Euclidean distance between clustering features and the geometric distances between nodes spatial positions. This combination is done through two user defined coefficients with unitary sum, which determine the relevance of these two aforementioned distance metrics during the clustering process, and it is aimed at only merging physically linked nodes and clusters, so to prevent the algorithm from defining physically unfeasible zonal configurations made up of internally detached *BAs*. For these reasons, this "improved fuzzy membership" represents an alternative way to check nodes' physical connection with respect to the previous external check, based on the adjacency matrix and always run before merging points in clusters.
  - **Applications:**
    - **[27]:** Improved fuzzy membership evaluated between single values of *LMPS*, deriving from ad-hoc snapshots of their hourly trends.

- **Connectivity Based Distance:**
  - **Definition:** It is a way of updating the medoids during the K-medoids clustering steps. Using this distance metric, they are iteratively chosen by looking at the number of nodes' electrical connections: the more a node has a high number of electrical connections with other nodes, the more its feature used for the clustering is likely to become the medoid of its cluster.
  - **Applications:**
    * **[25]:** The connectivity based distance is here used, inside the adopted K-medoids clustering.

2.5 Clustering algorithms' strengths and weaknesses

The following bulleted list presents for each of the clustering algorithms, which have been used inside the papers considered in this chapter, its strengths and weaknesses. Each of these last is also endowed with a reference to where the specific comment can been observed.

- **BubbleClust Algorithm:**
  - **Pros:**
    * **[22]:** *PTDFs* are smartly weighted using lines' congestion rate factors. In order to be sure of finding the most congestible lines as inter-zonal connections when clustering *PTDFs*. So as to produce only inter-zonal congestions rather than the intra-zonal ones, which are unpleasant in zonal-based market since they represent an additional cost. Due to their alleviation, which has to be manually done by the *TSO* and hence is costly. Congestion rate factors are obtained as average of the transmission line's average congestion cost over the sum of all the other transmission lines' average congestion costs. Each of these average congestion costs is the arithmetic average of several *KKT* multipliers, namely Lagrange multipliers, all associated to the line's maximum power flux constraint and respectively deriving from several runs of *DCOPF* in different load and generation scenarios.
    * **[22]:** It seems one of the best PTDFs-based clustering algorithm. Because, computing the social welfare of the zonal configuration resulting from this clustering algorithm and comparing it to the ones obtained by two other zonal configurations coming from else articles' algorithms. Which respectively create the *BAs* by merely merging the highest absolute values of *PTDFs*, or by using also the *PTDF* matrix in order to obtain a clear-cut distinction whether a power injection in a node increases or not the power flow in a given direction over a line. This clustering algorithm's social welfare reveals to be the highest of all. And therefore, remembering that the social surplus is usually considered the most representative *KPI* of a market, it can be deduced that the BubbluClust algorithm seems to

be most performing clustering method within its category that deals with PTDFs-based approaches.

  * **[22]:** This clustering algorithm also gives the possibility not to allow zones without generator. This is important since these purely importer zones are badly seen by regulators, and thus not accepted, for many reasons. First of all, the impossibility to create an energy market without the presence of a generator. Which would be paradoxical for a *BA*, that must obviously be able to have its personal zonal market in case of system's congestions and consequent power network zonal division. And secondly, the market power opportunity that would be clearly given in zones without generators. It is worth remembering that, the activation of this extra constraint decreases the social welfare of the output zonal configuration. But this is obvious and predictable, since *OP*'s constraints always move the solution away from its global optimum. Therefore, the more constraints are put, the less the outcome is optimized.

– **Cons:**

  * **[22]:** The social welfare estimation would be more accurate if redispatching costs were included. And this is rather important, since the social surplus is one of the criteria used to assess the effectiveness of the newly defined *BAs*.
  * **[22]:** The overall obtained social surplus is anyway small if juxtaposed with other parameters' orders of magnitude. Though, the 39 bus system here considered may constitute a not complex enough space to illustrate all the potential benefits of this clustering algorithm. Therefore, further analysis should be necessary to get a clearer evaluation of this drawback.
  * **[22]:** The Generation Shift Key (*GSK*) matrix is extensively used to translate zonal injections into power flows. But it has two drawbacks. On the one hand it contains many a priori assumptions about the load and generation levels. And so it can be wrong with a certain error percentage. On the other one, this matrix becomes meaningless if there is a self-sufficient zone. Namely a zone with net position equal to zero. Therefore, these two reasons explain why the using of the *GSK* matrix can be considered an Achilles heel for this clustering algorithm. And thus, it would be necessary an alternative to its use.
  * **[22]:** Over and under estimations of most congestible lines' power fluxes are symmetrically treated in this algorithm. Still, a forecast power flow which overestimates the actual value is "safer" than reality. Whereas, a forecast power flow with underestimates the actual value is physically dangerous for the entire power system. Therefore, prediction errors which underestimate the actual power flows should be penalized much more than the ones overestimating them.
  * **[22]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent

this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.

  - **[22]:** The number of zones has to be user defined in advance of the algorithm run. Because the stop criterion of the process relies on it.

- **Fuzzy-c-means:**
  - **Pros:**
    - **[17]:** Large dataset can be easily handled like in K-means clustering, which is quite similar to this algorithm. Optimal thing for power networks analysis, actually associated with big databases.
    - **[17]:** Small standard deviation and maximum range of prices for *LMPs* associated to nodes inside the same clusters. This is a pro. Because having very different nodal prices from the zonal prices which approach them, reveals a high loss of economic signals by moving from the benchmark nodal configuration to the zonal one. Which is actually the trade-off between the ideal nodal pricing and the worst uniform one. This is why an optimal zonal configuration should have as low as possible standard deviation and maximum range of prices, namely within-clusters variance, for the *LMPs* associated to nodes within the same zone.
    - **[41]:** By using the sensitivities of nodal power injections to power flows on congested lines as clustering input, namely the nodal *PTDFs* referred to these lines, the resulting zonal configuration acquires an improved temporal stability. Which is definitively a pro according to *CACM*'s guidelines on optimal zonal configurations. This happens because, as proven in reference [39], nodal *PTDFs* of congested lines can be linked to nodes' *LMPs*. Therefore, they firstly can be effectively used to create *BAs* in alternative to more typical nodal prices. But moreover, *PTDFs* do not vary with system operating conditions. Since they only depend on power network's topology. Consequently, it becomes obvious that using these parameters as clustering input leads to define a relatively more stable final zonal configuration.
  - **Cons:**
    - **[17]:** Number of zones has to be given by the user beforehand like in K-means clustering, which is quite similar to this algorithm.
    - **[17]:** The clusters centroids must be user defined or randomly chosen at the beginning of the process, like in K-means clustering which is quite similar to this algorithm. Therefore, also here this initialization makes the clustering result depend on the initial assignments. Ruining the original idea of automatically finding an optimal zonal configuration using a clustering algorithm. Because here, if these initial assignments are not well chosen, the algorithm only converges to a local optimum and not to the global one desired by the user.

    * **[17]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.
    * **[17]:** The split of detached *BAs* at the end of the clustering process aimed at defining a feasible zonal configuration made up of zones with physically linked within-cluster nodes, namely the aforementioned extra control on this feature, is counterproductive and hence should be substituted. Because it leads to the definition of too small *BAs*, unacceptable because of the possible rise of market power that could happen there.

- **Genetic Algorithm:**
  - **Pros:**
    * **[3]:** *GA* easily handles very large systems. Optimal thing for power networks analysis, actually associated with big databases.
    * **[3]:** Hourly *LMPs* patterns along years are easy to find. It is enough a *DCOPF* of the analysed system.
    * **[3]:** Unstable zones borders would be typical for LMPs-based zonal configurations, since prices are variant time variables. But, this is here prevented through a multi-scenario analysis. From which a good temporal stability of the zones is ensured.
    * **[3]:** No initial clusters centroids have to be defined by the user as input.
    * **[5]:** Clear assessment criteria for the newly defined zonal configurations are provided inside this paper. Divided between monetizable criteria, which want to be minimized since they are costs, and hardly monetizable ones, which are verified when inside a certain range.
  - **Cons:**
    * **[3]:** The GA needs the number of zones to be user defined as input data. This is a con, since it is physically impossible to know the optimal number of zones in advance of the clustering algorithm's execution.
    * **[3]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.
    * **[5]:** The optimized *BAs* coming out from this GA seem not to be so profitable. Since the decrease of the total system costs, respect to the initial ones of the reference current *BAs*, is not so marked. This saving nearly

disappears when a multi-year cadence is chosen to redefine zone borders. Whereas it lightly increases when a quarterly changing delimitations is instituted, but this situation is not acceptable for two reasons. On one side from a political point of view, just think of *CACM*'s guidelines for an optimal zonal configuration which ask for a temporal stability of *BAs*. And on the other one from an economic perspective, since a faster changing zonal definition would certainly lead to higher costs here neglected. That would end up erasing the saving increase. For these reasons, a trade-off has to be found even to preserve the aforementioned small savings of the optimized *BAs* respect the current zonal configuration.

- **Geographical Clustering:**
  - **Pros:**
    - **[17]:** No number of clusters has to be user defined beforehand, like the majority of clustering algorithms, or at the end of the process, like hierarchical clustering. Because here the zones number is naturally derived from the statistically most congestible lines. And particularly, it is directly proportional to them. Since a higher number of congestible lines would lead to a higher number of zones borders and hence to a higher number of areas.
    - **[17]:** The zonal configurations which result from this clustering algorithm are certainly feasible from the physical point of view. Because they are sure made up of zones with physically linked nodes, since here the areas are defined by cutting the existing power network along the statistically most congestible lines.
  - **Cons:**
    - **[17]:** High standard deviation and maximum range of prices for *LMPs* associated to nodes inside the same clusters. This is a con. Because having very different nodal prices from the zonal prices which approach them, reveals a high loss of economic signals by moving from the benchmark nodal configuration to the zonal one. Which is actually the trade-off between the ideal nodal pricing and the worst uniform one. This is why an optimal zonal configuration should have as low as possible standard deviation and maximum range of prices, namely within-clusters variance, for the *LMPs* associated to nodes within the same zone. In other words: the more the within-cluster *LMPs*' heterogeneity increases, the worse. Because *LMPs* contain the clearest and most objective economic signals. Thus the more a node has to accept a zonal price different from its natural *LMP*, the more it loses the correct economic signal coming from its natural *LMP* in favour of a misleading one. That is why, being aware of the fact that economic signals deriving from *LMPs* go towards the alleviation of congestion, losing these signals bring the power network to a higher inefficiency for worse congestion management.

- **Hierarchical Clustering:**
  - **Pros:**
    - **[4]:** LMP hourly patterns along years are easy to find. It is enough a *DCOPF* of the analysed system.
    - **[4, 9]:** The number of zones has not to be defined by the user beforehand, as input. Though, at the end of the clustering it is always the user who must decide the number of zones. The only pro is the possibility of doing it in front of an overview of zonal configurations with different zones number, represented by the dendrogram.
    - **[4]:** No initial clusters centroids have to be defined by the user as input.
    - **[8]:** *BAs* of similar dimensions can be easily obtained by using weighting factors for nodes according to their energy relevance. The more they withdraw or inject from or into the power network, the more their weight makes them relevant during the clustering process. Avoiding in this way the birth of too small *BAs*, which could be characterized by the rise of market power.
    - **[9]:** The hierarchical clustering algorithm is a non-linear optimization problem. So that it would not be suitable for large dimension systems like power networks, since their solution would take too much time. A heuristic clustering algorithm like the GA would be more recommended from this point of view. Although modifying the hierarchical clustering, by only considering at each iteration the couples of adjacent zones as feasible for the merging. Namely the zones which are distinct but linked by a transmission line at least. The result is that at each step there is no a full recalculation of the objective function, but there is only an update of it. Thereby, the solution time is enormously reduced. Making the hierarchical clustering algorithm a profitable choice in power networks too. And moreover, this automatically ensures that any newly formed zone will only consist of nodes that are physically connected. From which, no physically unfeasible zonal configurations will be defined.
    - **[9]:** Unstable zones borders would be typical for LMPs-based zonal configurations, since prices are variant time variables. Nevertheless, this is here prevented through a multi-scenario analysis. From which a good temporal stability of the zones is ensured.
    - **[25]:** In terms of *RE* index, it shows better performance than K-means and K-medoids clustering algorithms. The *RE* index represents the costly redispatch effort needed in a certain *BAs* configuration, after its dispatching has been defined by its zonal market clearing, with respect to the one needed in a nodal pricing power system. This latter is null, since nodal-based markets automatically do the short-term congestion management. Without any need of redispatch. Thus the lower the *RE* is, the better for the zonal configuration. Having recalled this, in paper [25] the *RE* index of zonal configurations based on multi-periods and with less than twenty

or thirty zones is proven to be the minimum when the *BAs* derive from a hierarchical clustering. Instead of a K-means or K-medoids clustering, also considered within the paper. Therefore, hierarchical algorithm outperforms both the K-means and the K-medoids one in this category. And this is very important. Because this class indicates a feasible number of zones, since zonal configurations always have to contain their number of zones to actually be optimal, and creates zonal configurations with a good temporal stability, by considering more than a single snapshot of time as input for the clustering.

* **[36]:** Thanks to a consensus clustering algorithm is easily possible to define an optimal zonal configuration with improved temporal stability, because based on several power network's scenarios. In this paper this possibility is applied on 722 zonal configurations respectively associated to different wind scenarios. In order to improve the temporal stability towards weather conditions variability.

– **Cons:**

* **[4]:** The main con of this clustering algorithm is reported in table thirteen. In this latter the first column is nodal pricing performance, followed by the zonal pricing one with nine or six optimized zones, and finally the zonal pricing performance with six reference zones. Where "reference" stands for the current zonal configuration, based on national borders. It is worth remembering that both six and nine optimized zones schemes are considered, in order to investigate the impact of the *BAs* number on system performance. This analysis reveals a small dependence of market efficiency by the number of zones. From which six areas are chosen, in order to keep the number of zones inside the current *BAs* definition. Beyond this, the first table row is the wholesale cost of the system market, the second is the redispatch cost and the third is the total cost which includes the previous two. All of these being costs, the less the better. Consequently, as predictable the nodal pricing becomes the benchmark. Namely the system with the highest performance, as confirmed by its lowest total cost. But also, the reference zonal pricing total costs do not reveal to be so higher than the ones from nodal pricing and optimized zonal pricing. Therefore, remembering that some system costs are even neglected into this paper like costs deriving from the adjustment of the system. And they could even erode the small savings of optimized zonal pricing respect the current reference *BAs* delimitations. It realistically does not seem to be profitable to define newly born *BAs* using this clustering algorithm.
* **[4]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections

evaluations.

* **[18]:** LMPs-based clustering algorithms for *BAs* redefinition do not always place the congested lines on the zones' borders. This is a con for two reasons. On the one hand since it requires an additional manual and costly readjustment done by the *TSO*, after that the market coupling mechanism has already found the supply/demand equilibrium of the zonal system. On the other hand, because it works against the main idea the zonal market should serve. That is keeping the transactions in market equilibrium in close relation to the physical flows of power in the grid.
* **[25]:** Higher computational effort with respect to the K-means and K-medoids clustering also used in this paper. This was predictable since, generally speaking, all the connectivity-based clustering algorithms are heavier than centroid-based ones from the computational point of view. This is because, in every step they have to recalculate the objective function for each of the possible connections between clusters. And then, only the most profitable of them in terms of objective function optimization is done at the end of the step. This is the time consuming part of the connectivity-based clustering algorithms.
* **[36]:** Many too small *BAs* are observed inside the optimal zonal configuration coming out from last consensus clustering algorithm. They are unacceptable for mainly three reasons here explained. The first one is the market power, which could arise in these too small zones. Because the hypothetical sole generator within one of them would automatically be able to make the price of that zone, gaining an infinite market power and ruining the perfect competition of the desired reference market. Secondly, tiny clusters usually include only few loads and no generators. Consequently, they constitute purely importer zones which can only be fed through energy readjustments made by the *TSO* in neighbouring zones. But these last are manual, hence costly and so unwanted in an optimal zonal configuration. Thirdly, tiny *BAs* are hard to be accepted from the sociological point of view. In fact, it may be difficult for the society to accept a small zone where the price is usually higher than in neighbouring areas. Because of the typical lack of generators which characterizes these zones for the aforementioned consideration. That is why too small *BAs* are never accepted into optimal clustering algorithms. And it is always suggested to eliminate by merging them with a neighbouring zone. The choice of this latter depends on the adopted methodology. For instance, in this paper the too tiny *BAs* are respectively merged to neighbouring zones for which the within-clusters variance of the resulting newly defined *BA* increases the less. Anyway, it is worth remembering that it is not sure the connection between their presence and the hierarchical clustering algorithm here executed. They can also depend on the additional consensus clustering run at the end of the method. Therefore, further analyses are necessary.

- **K-means:**
  - **Pros:**
    - **[25]:** It is able to handle very large dataset, like all the centroid-based algorithms. Optimal thing for power networks analysis, actually associated with big databases.
    - **[25]:** Since it is well known that K-means algorithm's clustering results strongly depend on the centroids initialization made at the beginning of the process. Here the clustering process is run several times, respectively with different clusters centers initialization, and then only the best result is kept for each category. In order to avoid as much as possible finding a local optimum instead of the global one.
    - **[25]:** Its zonal configurations always reveal lower RE indices, hence worse, respect to the ones associated to BAs configurations coming from K-medoids clustering.
    - **[25]:** It requires less computational effort than the K-medoids and hierarchical clustering also used within this paper.
    - **[44]:** It is possible to fix the problem for which different runs of a K-means clustering algorithm on the same input database can lead to completely different partitions. Which is caused by the strong influence that the choice of the initial clusters' centers has on the final partitioning result of all the centroid-based clustering algorithms, like is the K-means. And it becomes a problem during BAs definition. Because the perfect methodology should be able to deterministically find out the global optimum of the clustering problem, namely the best applicable zonal configuration of the power network in question. But this cannot always happen whether the clustering result depends on some user-defined input data, like typically are the aforementioned centroids. Because, if these last are not chosen properly the clustering OP only converges on a local optimum, instead of the desired global one. For this reason, in all the centroid-based clustering algorithms like K-means, fuzzy-c-means or K-medoids it would be better to automatically initialize the clusters' centroids through the process itself, rather than doing it manually. So as to remove any uncertainty on the final clustering result once given the input database, and to permit the clustering OP to always reach its global optimum, if it is able to do so. This is actually done inside this paper. Where cluster's centers are not manually defined at the beginning of the process, but they are automatically indicated by the algorithm by maximizing the geographical distance between them. So as to also cover as much as possible the power network's area, in order to minimize the number of failures of the subsequent check on within-clusters nodes' physical connection. Which is often included in order to prevent the clustering algorithm from defining unfeasible zonal configurations. Therefore, inside this improved K-means the user has only

to define the number of clusters like in any other centroid-based clustering algorithm. And then, the algorithm itself chooses the initial centroids among the nodes of the input database. By indicating its group of nodes which maximizes, for that user-defined number, the geographical distance between them.

- **Cons:**
  - **[25]:** It requires the number of clusters as user input. And moreover it has to be given in advance of the process, instead of at the end like hierarchical clustering.
  - **[25]:** The resulting clusters strongly depend on the clusters centroids which are randomly or manually selected at the beginning of the clustering process. It is a con because: if these initial assignments are not well chosen, the algorithm only converges to a local optimum. And not to the global one, that would obviously be desired by the user. In other words, the outcome quality depends on a user's input. That is not acceptable in an optimization algorithm like this one for the find of an optimal *BAs* configuration. For these reasons, some measures would be needed to contain this drawback. But this means more complexity, namely a con for the clustering algorithm.
  - **[25]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.

- **K-medoids:**
  - **Pros:**
    - **[25]:** It is able to handle very large dataset, like all the centroid-based algorithms. Optimal thing for power networks analysis, actually associated with big databases.
    - **[25]:** Since its clustering results strongly depend on the medoids initialization made at the beginning of the process, like for the similar K-means clustering. Here the clustering process is run several times, respectively with different clusters centers initialization, and then only the best result is kept for each category. In order to avoid as much as possible finding a local optimum instead of the global one.
  - **Cons:**
    - **[25]:** All the K-means' drawbacks. Since this clustering algorithm is quite similar to the previous one, except for the choice of clusters centroids. Which here have to correspond to one of the within-clusters' points.

  - **[25]:** It has a more intensive computational burden than a K-means clustering.
  - **[25]:** Its zonal configurations always reveal higher *RE* indices, hence worse, respect to the ones associated to *BAs* configurations coming from both K-means and hierarchical clustering.

- **Sequential Network Partition With CCI:**
  - **Pros:**
    - **[19]:** The physical connection between nodes inside the same zone is granted, and hence only physically feasible zonal configurations are defined. This is because the new *BAs* are defined by cutting the existing power network along the congested lines. In order to eliminate intra-zonal congestions in favour of inter-zonal ones, which are the only one that can be seen and thus automatically alleviated by a zonal-based market.
    - **[19]:** The zones borders are sure composed of congestible lines. Since these last are actually used to define areas boundaries. Therefore, the zones number is theoretically the minimum to efficiently do the congestion management of the power network. Because new zones are created only when an intra-zonal congestion occurs, and are aimed to its removal. This is a pro, since finding the optimal zonal configuration is always important to limit the number of zones. Because if this latter tended to its maximum, namely the number of system nodes, there would obviously be only inter-zonal congestions. But this would become a nodal configuration, which is unacceptable for many reasons. From the computational burden to the possible rise of market power that may happen inside nodes. Therefore, an optimal zonal configuration should give the benchmark performance of nodal configuration as much as possible. But with the possible minimum number of zones. So to actually become the optimal trade-off between the nodal pricing's highest performance and complexity, and the uniform pricing's highest inefficiency and simplicity.
    - **[19]:** In an ideal *NP* mechanism, the distribution of nodal prices is exactly consistent with their *PTDFs* to congestion lines. This is even proven inside reference [39]. Therefore, since the here used approach clusters nodes with the same sign of *PTDF* into the same zone, nodes within the same zone would have relatively close nodal prices. Confirming the consistency of using these zones as *BAs*, namely group of nodes with *LMPs* similar to the zonal price which approaches them. So to lose as little as possible of the reference economic signals embedded inside *LMPs* when moving from them to the zonal prices of the *ZP* mechanism.
    - **[19]:** *PTDFs* are not time dependent variables. Therefore, the resulting zonal configuration is automatically stable from the temporal point of view. That is one of the requested criteria for an optimal partitioning, according to *CACM*'s guidelines. And moreover, it is not granted with LMPs-based zonal configurations due to their relying on time dependent

variables (*LMPs* are hourly functions). Which obliges those clustering algorithms to embed complicated multi-scenario analysis, to guarantee a satisfactory temporal stability of their *BAs*.

* **[19]:** No number of zones has to be user defined beforehand, like the majority of clustering algorithms, or at the end of the process, like hierarchical clustering. Because here the zones number is naturally derived from the lines defining an intra-zonal congestion. And particularly, it is directly proportional to them. Since every time that a new intra-zonal congestion is detected, the associated targeted congested line automatically becomes the border between two newly defined *BAs*.

– **Cons:**

* **[19]:** The classification between *NWGs* and *NWOGs*, with the consequent simplification of the problem by neglecting the last ones, is an action which largely cuts the complexity and the size of the *OP* that represents the numerical market clearing of the power network. But meanwhile, it is a huge drawback of the clustering algorithm. Because treating a nowadays more realistic two-sided energy market would enormously increase the computational burden of the method. Which is already high even considering the aforementioned simplification.
* **[19]:** The computational burden of this clustering process is quite high. It may be useful to use a more efficient algorithm to make the dispatching of the system, namely to solve the *OP* which defines the numerical market clearing.

• **Price Differential Clustering:**

– **Pros:**

* **[17]:** No number of cluster has to be user defined beforehand, like the majority of clustering algorithms, or at the end of the process, like hierarchical clustering. Because here the zones number is naturally derived from maximum allowed difference between average *LMPs*. And particularly it is inversely proportional to it. Since a lower maximum allowed difference between average *LMPs* would lead to a stricter zonal division and hence to a higher zones number.
* **[17]:** Small standard deviation and maximum range of prices for *LMPs* associated to nodes inside the same clusters. This is a pro. Because having very different nodal prices from the zonal prices which approach them, reveals a high loss of economic signals by moving from the benchmark nodal configuration to the zonal one. Which is actually the trade-off between the ideal nodal pricing and the worst uniform one. This is why an optimal zonal configuration should have as low as possible standard deviation and maximum range of prices, namely within-clusters variance, for the *LMPs* associated to nodes within the same zone.

  - **Cons:**
    * **[17]:** Many single-node and two-nodes zones are defined. Even if differential price percentage is increased from 5% to 10%. Since this latter action increases two-nodes areas while decreasing single-node ones. This is a con, because too small areas are economically unacceptable due to the generators' market power that could arise inside them. That would move the energy market away from the reference perfect competition.
    * **[17]:** The split of detached *BAs* at the end of the clustering process, aimed at defining a feasible zonal configuration made up of zones with physically linked within-cluster nodes, is counterproductive. Because it leads to the definition of too small *BAs*, unacceptable because of the possible rise of market power that could happen there.

- **Scale-Space Hierarchical Clustering:**
  - **Pros:**
    * **[39]:** Using the sensitivities of nodal power injections to power flows on congested lines as clustering features, namely the nodal *PTDFs* to them referred, this algorithm manages to create zonal configurations which do not only reflect nodal prices (since here it is proven that clustering the *PTDFs* of most congestible lines is actually comparable to clustering the system *LMPs*). But which also do not vary with operating conditions. Thus improving the temporal stability of the *BAs* configuration, by providing a relatively stable price zone partition in a period of time. Thanks to this algorithm's similarity to classical hierarchical clustering algorithms, this pro could be extended to these last too.
    * **[39]:** No number of zones has to be user defined beforehand, like the majority of clustering algorithms, or at the end of the process, like classical hierarchical clustering algorithms from which this method derives. Because here the zones number of the optimal zonal configuration is deterministically chosen by using three newly defined parameters for the *BAs* assessment, which are respectively *BAs*' lifetime, compactness and isolation. Through them it is automatically chosen the best zonal configuration among the sequence of *BAs* sets provided by the final summary dendrogram.
  - **Cons:**
    * **[39]:** Higher computational effort than centroid-based clustering algorithms like K-means or fuzzy-c-means. This drawback derives by the algorithm's similarity with classical hierarchical clustering algorithms. Which typically suffer this situation.
    * **[39]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely

a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.

- **Imitating Out-Point Method With Tabu Search:**
  - **Pros:**
    - **[31]:** The proposed clustering algorithm is automatically finished. Hence no number of zones has to be user defined in advance, like the majority of clustering algorithms, or at the end of the process, like hierarchical clustering. This is important, because it is a not known a priori information for the desired optimal zonal configuration.
    - **[31]:** Fast *BAs* definition method despite its heuristic clustering algorithm. This is thanks to the enclosed taboo search, which speeds up the process.
  - **Cons:**
    - **[31]:** *LMPs* snapshots here used as input feature for the clustering, either obtained as averages of hourly trends or instant values of certain moments, ask for an additional preliminary process on input data. Moreover, the resulting zonal configurations are likely to be worse than hourly trends-based *BAs* from the temporal stability point of view.

Chapter 3

Methodology

This chapter describes the methodology created inside this thesis, in order to attempt finding a deterministic approach to define an optimal power network zonal configuration.

The rationale behind this process is firstly proposed in Fig. 3.1 through a block diagram, which actually represents the organization of this chapter. More in-depth analyses follow in the subsequent sections.

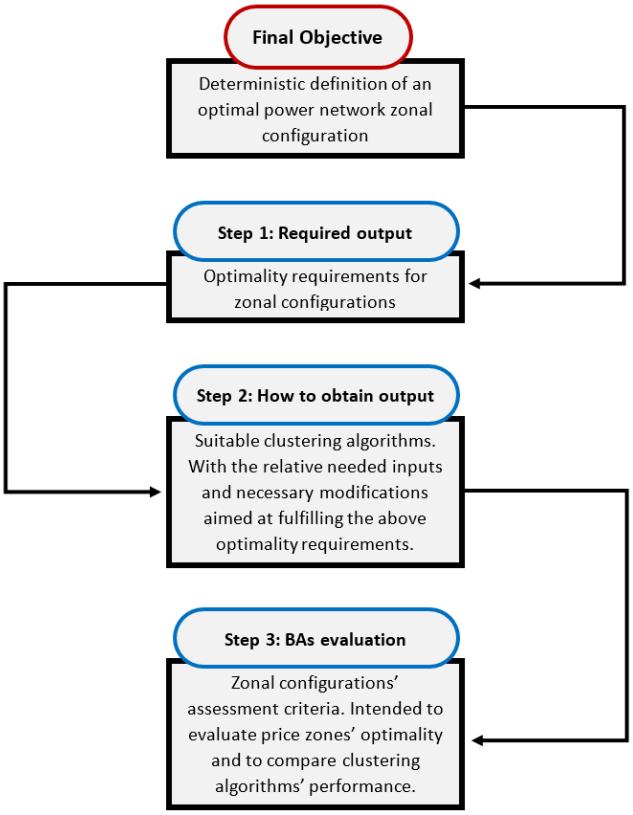

Figure 3.1: Block diagram of the thesis' methodology.

3.1 Optimality requirements for zonal configurations

In the introductory chapter it has been recognized a lately growing interest in optimal *ZP* mechanisms by the European Union (*EU*). This latter has been caused by the increase of power networks' congestions, which consequently has led to the rise of respective system costs in the sense of redispatch ones. As previously mentioned, current European power networks are all based on *UP* schemes or non-optimal *ZP* ones, where also intra-zonal congestions can occur. For this reason, these market structures are not able to perform a free-cost congestion management, since they often end up requiring a manual and thus costly readjustment by the *TSO* to alleviate their intra-zonal congestions. Therefore, this increasing interest towards the optimal zonal configuration definition has caused two consequences. Firstly, the scientific literature concerning the subject has grown, as shown in the previous chapter, which actually collects papers that already tried to deterministically define optimal *BAs* using clustering algorithms. Secondly, the *EU* has emanated in 2011 the Framework Guidelines on Capacity Allocation and Congestion Management for Electricity, through the Agency for the Cooperation of Energy Regulators. These guidelines aimed at precisely clarifying the features that a zonal configuration should have in order to be optimal. But actually, they only managed to give general indications on the subject, failing to close the game of optimal *BAs* definition. To demonstrate this, the main lines of this document are hereunder reported.

> ... The *CACM* Network Code(s) shall ensure that, when defining the zones, the *TSOs* are guided by the principle of overall market efficiency. This includes all economic, technical and legal aspects of relevance, such as, socio economic welfare, liquidity, competition, network structure and topology, planned network reinforcement and redispatching costs. The definition of zones shall further contribute towards correct price signals and support adequate treatment of internal congestion...
> ... The *CACM* Network Code(s) shall foresee stable and robust zones over time...

Having therefore noted the current absence of strict rules on zonal configurations' optimality, it firstly results necessary to make up for this lack to proceed towards this thesis' goal. In fact, only clearly stating the requirements that a zonal configuration must fulfill to be optimal, it becomes possible to quantitatively and objectively define the optimal *BAs* definition goal through unequivocal parameters, which then permit to set up a problem from the engineering point of view. For these reasons, the following lines propose a bulleted list which tries to strictly review the features requested to a zonal configuration in order to be optimal. This latter has been created by mixing *CACM*'s general provisions with case studies' findings, taken from the analyzed papers. Therefore, it does not claim to become the dogmatic truth on the subject. But anyway, in the current state of things it seems to be the most exhaustive treatment on zonal configurations' optimality requirements.

1. **"Any optimal zonal configuration should be as stable as possible from the temporal point of view".**
   This requirement derives from the European guidelines [52] on the subject, contained inside the aforementioned *ACER*'s document. In fact, the final quote of the above reported lines talks about "stable and robust zones over time". In fact, having an

often changing zonal configuration would be unacceptable for many reasons, like the additional costs linked to bureaucracy, which would arise whenever the *BAs* shape is changed. Or the market players' contrariness towards this situation, due to their inability to define profit-maximizing bidding strategies with an acceptable certainty. Anyway, it is worth remembering that it is not possible to define an optimal zonal configuration able to preserve its optimality forever, because the power network's structure unavoidably changes over time, for instance due to the market players' investments driven by the economic signals automatically embedded in energy prices. In addition in the meanwhile the power network's users change as well, just think to the unpredictable future expansion of renewables. Still, this is not what the here considered requirement asks. In fact, this latter just states the impossibility to define zonal configurations changing in a too little range of time, namely lower than three or four years, because the other way around they would be affected by the above reported drawbacks. But, in the same time, it does not close the door to the possibility of a zonal configuration regularly changing at reasonable time intervals, aimed at preserving its optimality despite of contour changes. Because otherwise there would be a risk of ending up in the Californian power market situation, where electricity prices are distorted and intra-zonal congestions are severe due to the use of three invariable price zones in the long term [39].

2. **"Any optimal zonal configuration should not hinder network security, for instance intended as N-1 security".**
   This security requirement is unavoidably included in any power network management, and hence in any of its pricing schemes too, as proven by the role of the *TSO*. Which, among other things, has exactly to coordinate the supply of and demand for electricity in order to guarantee the respect of its power network's security and reliability criteria.

3. **"Any optimal zonal configuration should boost as much as possible market efficiency".**
   This requirement derives from the European guidelines [52] on the subject, contained inside the aforementioned *ACER*'s document. In fact, the first quote of the above reported lines talks about defining zones "guided by the principle of overall market efficiency". This because, as previously stated within the introductory chapter, the *ZP* mechanism is a trade-off between the *UP* scheme's simplicity and poor performance and the *NP* scheme's complexity and excellent performance. Hence, the more a zonal configuration is optimal, the more the associated *ZP* scheme approaches *NP* benchmark performance without acquiring its drawbacks. Since these last are mainly due to the use of single-node price zones, which are not allowed inside optimal *ZP* mechanism as it will be following stated. Therefore, it goes without saying that, in order to have an optimal zonal configuration, its market efficiency has to be as high as possible, theoretically equal to the benchmark one of nodal configuration whether the newly defined *BAs* are completely optimal.

4. **"Any optimal zonal configuration should cope in adequate way with congestion management, by only having inter-zonal congestions instead of intra-zonal ones".**
   This requirement does not clearly derive from the above reported European provisions [52] on the subject. Since these last both deal with an "adequate treatment of internal congestion" and consider redispatch costs as an "aspect of relevance" in the *BAs* definition. But meanwhile, they do not distinguish between inter-zonal congestions and intra-zonal ones. Nevertheless, it is important to make this distinction. Because, as it results from the case studies' findings of the considered papers, only the inter-zonal congestions, which affect transboundary connections, can be seen and hence automatically free-cost alleviated by a zonal-based power market, through the divergence of zonal prices. Whereas the intra-zonal congestions, which affect within-zone lines, cannot be located in the same way and thus require to be manually alleviated by the *TSO*, through a costly readjustment of the dispatching already defined by the zonal-based market clearing. This happens because the power network model adopted within a *ZP* mechanism, whose physical limits become part of the inequality constraints inside the aforementioned zonal-based market clearing, consider all the inner parts of the various *BAs* as copper plates[1]. And thereby it does not include the transmission lines congested in the intra-zonal congestions, but only the ones congested in the inter-zonal ones. Therefore, in an optimal zonal configuration the price zones should always reflect areas where the transfer of energy is not limited by internal congestions, so as to permit the zonal-based power market to control the system's power flows, keeping them in the grid security limits. This is just like a *NP* mechanism would do, through a perfectly efficient congestion management, free of any additional redispatch costs.

5. **"Any optimal zonal configuration should be obtained by an easy to understand method".**
   This requirement does not derive from the above reported European provisions on the subject [52], but can be read between the lines of many studied papers. In fact, it is worth remembering when dealing with the economic world, which is unavoidably interested here talking about electricity pricing mechanisms, that the emanated measure's simplicity always becomes the discriminant of its application's success. As a proof of this, just think of the losses allocation problem. Even if in a completely precise method able to assign to each market player the exact quote of the system losses attributable to him is not currently available, there are many approaches able to make a better system losses partitioning among market players rather than the usually adopted postage stamp method. Still, all these last are based on quite difficult algorithms, which therefore end up impeding their actual put in force. For this reason, when trying to define an optimal zonal configuration it is important to preserve the simplicity of its creation method. Because otherwise the newly defined

[1] In electrical systems this terminology is used to point out power network portions where physical constraints are so high to be neglectable, because practically they never curtail the system's power flows.

optimal *BAs* could reveal to be actually impracticable in the real world, thwarting the work done.

6. **"Any optimal zonal configuration should limit as much as possible the number of *BAs*".**
   This requirement neither derive from the above reported European provisions [52] on the subject, nor is patently present in studied papers. Anyway it is important to be remembered. Since, given the fact that as previously mentioned *ZP* mechanism is a compromise between *UP* and *NP* one, it is obvious to imagine that increasing the number of *BAs* of a zonal configuration would progressively make this latter tend to a nodal configuration. And this would end up simultaneously improving the system's performance too. Nevertheless, this is not the correct way to find an optimal zonal configuration. Because having a number of price zones near to NP mechanism's one would also give to the *ZP* scheme in question all the nodal configurations' drawbacks which have been listed in the introductory chapter and nowadays prevent their put if force. Therefore, the resulting zonal configuration would actually result not to be optimal. Still having an optimal efficiency thanks to its similarity to a nodal configuration.

7. **"Any optimal zonal configuration should limit as much as possible the within-clusters variance of *LMPs*".**
   This requirement derives from the European guidelines [52] on the subject, contained inside the aforementioned *ACER*'s document. In fact, this latter states that "The definition of zones shall further contribute towards correct price signals". And, given the fact that *LMPs* represent the clearest and most objective price signals which could be defined, since they bring all the information about generation, demand, congestion and loss-costs to a nodal resolution. It naturally derives that, when merging nodes inside *BAs* in order to pass from a starting *NP* scheme to a *ZP* one, the less within-clusters *LMP* variance, the better. Because it means keeping inside the final zonal prices as much as possible of the nodal prices' benchmark economic signals. So as to actually render the zonal configuration in question an optimal one, namely able to imitate *NP*'s optimal performance without acquiring its drawbacks.

8. **"Each *BA* inside any optimal zonal configuration should have a minimal size, i.e. number of substations".**
   This requirement neither derives from the above reported European provisions [52] on the subject, nor is patently present in studied papers. And moreover it can be linked to the previous one, since it goes without saying that when increasing the number of *BAs* while keeping the power network dimensions fixed, the price zones' sizes simultaneously decrease. Nevertheless, this requirement focuses the attention on another aspect compared to the previous one. In fact, while the aforementioned rule fixes a limit on the *BAs* number aimed at preventing the resulting zonal configuration from acquiring the *NP* schemes' drawbacks, the here reported principle defines a bound on *BAs* dimensions intended to shield the zonal configuration in question from an additional issue. In fact, in a nodal configuration there are only single-node price zones and thereby the associate market power is inevitably gifted

with ad-hoc measures aimed at avoiding the rise of market power in the system's *BAs*. In a zonal configuration the single-node price zones are not the normal, and so the associated market power is often devoid of any of the aforementioned measures. This is the reason why too small price-zones, e.g. made up of just one node, cannot be accepted when defining an optimal zonal configuration. So as to prevent this latter from suffering from non-null market power in certain *BAs*, which would end up threatening system's perfect competition and hence its performance.

9. **"Each *BA* inside any optimal zonal configuration must be made up of physically linked nodes".**
   This requirement does not derive from the above reported European provisions [52] on the subject, but can be clearly read within several studied papers. In fact, having a zonal configuration made up of physically detached *BAs* is not acceptable for many reasons. First of all, from the practical point of view there is an organizational problem represented by the difficulty of managing price zones which are actually split on the territory. Though, this latter could be overcome through the use of a centralized control. Nevertheless, from the electricity pricing point of view, there is also a problem of having zonal prices divergence even when congestions actually happen in other parts of the grid. As proof of this, just think of an inter-zonal congestion between two price zones, one of which composed of physically detached nodes. This congestion would be successfully recognized by the zonal-based market clearing. And hence it would be automatically free-cost alleviated by the same, through the zonal prices divergence of the two interested *BAs*. Nevertheless, this latter action would provoke the change of electricity zonal price also in power network areas which actually have not seen the congestion in question at all. Namely the detached *BA*'s parts that are not linked to the congested transmission line. And this would create misleading economic signals in the system, preventing the adopted zonal configuration from being an optimal one.

10. **"Each node inside any optimal zonal configuration must belong to only one *BA*".**
    This requirement does not derive from the above reported European provisions [52] on the subject, but it is stated inside reference [44]. In fact, inside that paper it is clearly pointed out that each node of the system must belong to only one *BA*, in order to avoid the emergence of uncertainties during the electricity price attribution to the various power network's parts.

3.2 Suitable clustering algorithms

Having theorized the target features desired for the output through both quantitative and objective parameters, namely having listed the requirements that a zonal configuration must fulfill in order to be optimal, the question becomes now how to obtain outputs skilled this way. To answer this necessity, remember that *BAs* definition could theoretically be done in two ways, which are respectively the geographical clustering and the actual clustering algorithms. But only the latter can actually be used to produce optimal zonal configurations, as already mentioned inside the introductory chapter. The following lines rattle off the clustering algorithms which reveal to be most suitable for the aforementioned request. These last have been chosen by the author of this thesis by both looking at most diffused methods inside the scientific literature which deals with the subject, namely the papers described into the previous state of the art chapter. And picking out the clustering algorithms which, although not used so much in the previous researches, seemed to be particularly innovative but still applicable. Thus, the selected clustering algorithms are:

K-means clustering. It is a centroid-based hard clustering algorithm. The input database's observations are split out into K clusters, whose number is user-defined, according to the distance between each of them and the clusters' centroids, computed as average of each cluster's components. Therefore, in order to make the first algorithm iteration, once given the input database and the number of clusters, the user must initialize the clusters' centroids before running the process. This latter action can be done manually by the user, with a certain criterion, or randomly by the algorithm itself. But in both cases it becomes the main drawback of the final clustering result, because its quality turns out to be heavily dependent on the aforementioned initialization. Eventually, here like in many other clustering algorithms, it is up to the user to define the clustering features, namely the observations' skills that will be considered for their comparison, and the distance metric, namely the way of defining the observations' proximity or distance.

K-medoids clustering. It is another centroid-based hard clustering algorithm. Very similar to the previous one apart from the centroids. Here the centroids have always to be equal to one of the cluster's observations, and thereby are iteratively chosen for each cluster as its nearest observation to its observations' average. In the previous K-means the centroids were simply computed, at each iteration, as average of each cluster's observations.

Hierarchical clustering. It is a connectivity-based bottom-up and hard clustering algorithm. The input database's observations are initially considered as independent clusters. Then, they are progressively merged in pairs, following a user-defined linkage criterion, until they are all included inside a unique group. Therefore, once given the input database, this clustering technique should theoretically ask the user to specify only the aforementioned linkage criterion to run the process. In addition to nearest neighbors, farthest neighbors or average linkage criteria, the Ward's minimum variance criterion based on Euclidean distance as distance metric has often been used, as actually done inside this thesis' methodology. The final dendrogram,

summary of the partitioning sequence from many to only one cluster, is produced, giving then the possibility to the user to horizontally cut it to obtain an input database division in a certain number of clusters. Nevertheless, actually available hierarchical clustering algorithm codes do not allow this possibility, and require instead the number of clusters to be user-defined beforehand.

Genetic algorithm. It is not a usual clustering algorithm, since it comes into being as an optimization algorithm. However, it can be used to make an optimal zonal configuration, by writing an ad-hoc objective function able to embody as much as possible the aforementioned zonal configurations' optimality requirements. As a result, in this thesis it has been written an objective function aimed at minimizing the within-clusters *LMP* variance for each hour. The objective function has also been equipped with a penalty factor to make it difficult to merge physically detached nodes, which would end up producing unfeasible *BAs*. Once done this, the rest of the genetic algorithm proceeds as usual. Therefore, starting from a population of a certain user-defined number of chromosomes, which can be partially or totally randomly initialized, these last are iteratively undergone to genetic operations such as mutation or crossover. During these operations the unfitting chromosomes, according to the previously defined objective function, are gradually discarded due to the survival of the fittest ones, until the algorithm stops and gives out the fittest chromosome, namely the best solution of that moment. This latter convergence can be reached in two ways, because the user-defined maximum number of iterations has been reached, or because the algorithm has found a chromosome, i.e. a solution, able to optimize the objective function within a user-defined range of tolerance. Anyway, for having a deeper description of this algorithm's working process, please look at previous *Section 2.3*. Also in this clustering algorithm it is up to the user to define the number of clusters, which then could become *BAs* in the resulting zonal configuration. In fact, when defining the optimization problem that will be subsequently solved through the aforementioned genetic algorithm, the user must set up the lower and upper boundaries of its integer decisional variables which respectively assign each system node to one of the various clusters. Therefore, the lower limit will always be equal to unity while the upper one will time to time be equal to the user-desired number of *BAs*. Eventually, it is worth remembering that according to the scientific literature it is "genetic algorithm" when chromosomes' genes are bit, which hence can acquire either 1 or 0 as value. Whereas, it is "evolutionary algorithm" when chromosomes' genes are variables which can assume any integer value. Nevertheless, since the reference papers' authors always talk about "genetic algorithm", it is decided to preserve this nomenclature for coherence with the treated topic.

Price differential clustering. It is not a usual clustering algorithm. Like the previous genetic algorithm, it cannot be framed into one of the two main clustering algorithms categories, namely centroid-based and connectivity-based clustering algorithms, which otherwise include the first three aforementioned techniques. Moreover, it is a quite simple process which apparently tries to obtain an optimal zonal

configuration in the easiest possible way, namely, by merging power network nodes when both their *LMPs* difference drops below a certain user-defined range and they result to be physically connected. The just cited difference is evaluated between *LMPs*' averages over the simulation time. The main advantage of this clustering technique lies in not needing the number of clusters, namely *BAs* of the resulting zonal configuration, to be user-defined before the run of the process. In fact, this latter automatically derives by the algorithm execution itself once having set the range of tolerance between average *LMPs*.

The first three of the aforementioned algorithms, i.e. K-means, K-medoids and hierarchical clustering, are applied in a twofold version. On the one hand they are used in their standard versions, provided by the respective Matlab commands. And on the other hand, they are used in modified versions, obtained through the manual alteration of the respective source codes. This has been done in order to assess the effectiveness of a technique, often used inside the scientific literature dealing with this thesis' subject, aimed at satisfying one of the most critical requirements among the previously reported zonal configurations' optimality criteria, namely, the need of creating physically connected *BAs* so as to prevent the final zonal configuration being unfeasible, and thereby obviously non-optimal. According to many of the analyzed papers in fact, by using penalty factors during the clustering process to increase the distance of physically detached nodes, the resulting zonal configuration will be made up of only physically cohesive price zones, so complying with the above optimality requirement.

With regard to the last two clustering algorithms instead, they are applied in a sole version for two reasons. Firstly, because both of them are not based on a distance matrix, namely, a symmetrical and square matrix, with the number of rows and columns equal to the number of input database's observations. The distance matrix is instead computed in the above three methods before the process execution, and contains in the various cells the distances between all the possible couples of observations, computed between precise user-defined clustering features through a certain user-defined distance metric. Therefore, missing this distance matrix, it is impossible to think of applying penalty factors between physically detached nodes in order to make unlikely their merge in a common price zone. Secondly, because both of them are not available in terms of a Matlab command. In fact, as far as the genetic algorithm is concerned, the respective Matlab command is purely designed for being the solver of an optimization problem. Consequently, when trying to make it a clustering algorithm by using the aforementioned objective function, many errors occur with regard to the inputs syntax preventing the process fruition. Whereas, talking about the price differential clustering, there is really no Matlab command able to perform it. For these reasons, both of these two clustering algorithms are subsequently applied only in the customized version. That still contains the above approach of penalty factors as regards genetic algorithm, even if slightly modified since there a penalty factor is added to the objective function every time it is associated to a zonal configuration made up of physically detached *BAs*, in order to decrease its fitness and hopefully oust it from the surviving solutions. Whereas it does not contain at all the same with regard to the price differential clustering, because there the optimality requirement in question is straightaway complied by the process itself, which merges couples of nodes only when

their *LMPs* difference drops below a certain user-defined range and at the same time they result to be physically connected.

So, to summarize, the clustering algorithms which will be afterwards applied because chosen as the most suitable ones for this thesis' goal, represented by the deterministic definition of an optimal zonal configuration, are listed here below:

- Matlab K-means clustering
- Matlab K-medoids clustering
- Matlab Hierarchical clustering
- Customized K-means clustering
- Customized K-medoids clustering
- Customized Hierarchical clustering
- Customized Genetic algorithm
- Customized Price differential clustering

3.2.1 Suitable clustering algorithms' inputs

Firstly, the overwhelming majority of the aforementioned suitable clustering algorithms require to directly receive the number of clusters, which then becomes the number of *BAs* of the produced zonal configuration, as user input. The only algorithm that makes exception is the price differential clustering. Because there, as already said in the above lines, the price zones number naturally derives by the process execution itself once having set the range of tolerance between average *LMPs*. But actually, since this last parameter must be anyway defined by the user, also this exception ends up giving zonal configurations with user-dependant numbers of *BAs*. Therefore, directly or indirectly, it is always up to the user to choose the price zones number inside this methodology's algorithms. It is worth remembering this fact, because it also happens inside all the scientific literature which deals with clustering algorithms aimed at defining an optimal zonal configuration. And unfortunately, it constitutes one of the biggest obstacles in reaching these papers' goal. That is because, even once given the power network, it is impossible to know beforehand the exact number of *BAs* that will be associated to its optimal zonal configuration. But nevertheless, the user must still choose a number of clusters, as clustering algorithms require it as input. As a result, the scientific literature previously described inside *Chapter 2* has differently tried to overcome this impasse, also using algorithms external to the clustering ones aimed at price zones definition. But actually, the most performing ways for setting this hard-to-find input have revealed to be only two. On the one hand in fact, the *TSO*'s experience or needs regarding its power network can sometimes fix a priori the zonal configuration's *BAs* number. On the other one instead, it can be better to choose it by looking at some parameters' trends according to the number of price zones. Inside this thesis, since the subsequent case study will be based on a reduced model of the European transmission grid which hence interests the competence areas of several *TSOs*, the choice

between these two approaches is straightforward on the second. Even if, according to this thesis' author, this latter should always be preferred to the other because founded on power network's physical behavior instead of its past functioning or some bureaucratic requirements. Therefore, inside the following chapter, the zonal configurations' assessment criteria that will be subsequently described within the third level of the here outlined methodology will provide the basis for the aforementioned evaluation. This will suggest a reasonable number for a set of optimal BAs for each of the adopted clustering algorithms.

Secondly, the totality of the aforementioned suitable clustering algorithms needs to be fed with a nodal database of clustering features. Namely a set of data which fixes for each node a parameter, that will then be used inside the partitioning process to make the clusters, according to a certain user-defined distance metric that will eventually result high between different clusters' observations and low between same cluster's ones. The choice of this nodal database of clustering features is straightforward as regards the genetic algorithm and the price differential clustering. In fact, this latter method exactly bases its partitioning on average *LMPs*. Therefore, it necessarily requires as input a database made up of hourly patterns of system nodes' *LMPs*, so as to compute for each of them the average value over a certain period of time, which will then be used as discriminant of nodes union. As far as the genetic algorithm is concerned, it needs to be fed with the same nodal database of clustering features because its aforementioned objective function, that has to be written in order to transform it from an optimization algorithm to a clustering one, is exclusively imaginable aimed at minimizing the within-clusters *LMP* variance for each hour. And consequently, it goes without saying that the here considered genetic algorithm requires as input the same *LMP* hourly patterns of the price differential clustering. The genetic algorithm needs them to compute the above objective function, so as to carry out the desired power network partitioning. With regard to the other three clustering algorithms instead, namely the K-means, K-medoids and hierarchical clustering, which are afterwards applied both in a Matlab version and in a customized one, the choice of this nodal database of clustering features is not so obvious. In fact, it still makes sense to use *LMP* hourly patterns as clustering feature, since they always represent the clearest and most objective price signals that could be defined. These price signals moreover diverge when a congestion occurs, making the resulting zonal configuration have most congestible lines as transboundary connections ready to define inter-zonal congestions instead of the unwanted intra-zonal ones. It also becomes possible to use as clustering feature the nodal *PTDFs* of most congestible lines. In order to prove the validity of this latter statement, the next lines report a small demonstration, inspired by reference [39].

When writing the optimization problem aimed at doing the numerical market clearing of a power market in perfect competition, the first thing to do is to write an objective function. This latter is normally represented by the social surplus maximization or, if the market is based on a one-sided pool (i.e. a market with competition only on the suppliers' side which thereby is characterized by an aggregated demand with null elasticity), by the generators' costs minimization. After that, they have to be written the *OP*'s constraints, distinguishable in equality and inequality constraints. These last include the physical limits of system's generators, loads and transmission lines, and are one by one associated to a Lagrange multiplier of type μ inside the Lagrangian function that then permits to

solve the problem, which differs from zero when the associated constraint is activated. The equality constraints define the system's power balance between overall generation and demand, indispensable to keep stable the electricity frequency, and can be written in two ways. On the one side, it can be written a sole equality constraint containing all the *OP*'s decisional variables, represented by the various energy quantities respectively injected and withdrawn by the system's generators and loads. In this case, there will be a unique Lagrange multiplier of type λ, which will represent the *LMP* of the slack bus. On the other side, N equality constraints can be written, with N equal to the number of system nodes. *N-1* constraints will put in relation the net quantity of energy injected from that node with the loads and generators there present, while the N-th constraint will put in relation the net quantity of energy injected from the slack bus with all the ones coming from the other *N-1* nodes of the system. In this case, there will be N Lagrange multipliers of type λ, which will respectively represent the *LMPs* of all the system's nodes. This latter approach is the most interesting one, to proceed with the here provided demonstration. In fact, using this second way of writing the *OP*'s equality constraints, the whole nodal prices are obtained. By assuming to use a *DC* power-flow-model, it can be noted that they are composed as follows:

$$LMP_N = \lambda_N \tag{3.1}$$

$$\begin{cases} LMP_j = \lambda_j = \lambda_N + \overrightarrow{h}_j^T * \overrightarrow{\mu} \\ j = 1, \dots, N-1 \end{cases} \tag{3.2}$$

Where $\overrightarrow{h}_j \in \Re^{L,1}$ is a column vector, with L equal to the number of power network's transmission lines, that contains the *PTDFs* of system's lines referring to the j-th node. The *PTDFs* represent the sensitivities of that node's power injection to system's lines power flows having assumed to take the energy in question from the slack bus. Furthermore, $\overrightarrow{\mu} \in \Re^{L,1}$ is a column vector that contains the Lagrange multipliers associated to the various inequality constraints of the power network's transmission lines. As previously said, the Lagrange multipliers differ from zero when the respective constraint is activated, and hence the respective line is congested. Therefore, by making the difference between two different nodal prices it results:

$$\begin{cases} LMP_j - LMP_i = \lambda_j - \lambda_i = (\overrightarrow{h}_j'^T - \overrightarrow{h}_i'^T) * \overrightarrow{\mu}' \\ i \neq j \\ i = 1, \dots, N-1 \\ j = 1, \dots, N-1 \end{cases} \tag{3.3}$$

Where $\overrightarrow{h}_j' \in \Re^{L',1}$ and $\overrightarrow{h}_i' \in \Re^{L',1}$ are two column vectors, with L' equal to the number of power network's congested transmission lines, which contain the *PTDFs* of system's congested lines which respectively refer to the j-th and the i-th node. Furthermore, $\overrightarrow{\mu}' \in \Re^{L',1}$ is a column vector that contains the Lagrange multipliers associated to the various inequality constraints of the power network's congested transmission lines, because all the others are null for the aforementioned reason. Thus, from (3.3) it derives that the

difference between two nodal prices can be traced back to the sum of the differences between the two nodes' *PTDFs* associated to power network's congested lines, each of them weighted through the Lagrange multiplier μ of the respective congested line's inequality constraint. The only case of ambiguity could derive from the difference between the nodal price of a normal node and the one of the slack bus. But in that case it would result:

$$\begin{cases} LMP_j - LMP_N = \lambda_j - \lambda_N = \overrightarrow{h}'^T_j * \overrightarrow{\mu}' \\ j = 1, \ldots, N-1 \end{cases} \tag{3.4}$$

That once again proves the direct proportion between *LMPs* difference and congested lines' *PTDFs* one. Since here, the only dissimilarity is represented by having a node with null *PTDF* on whatever lines, namely the slack node. For these reasons, it has almost been explained the aforementioned statement. Which identifies in nodal *PTDFs* of most congestible lines a clustering feature able to compose the nodal database of clustering features for some of the suitable clustering algorithms reported inside this chapter, in alternative to the more typical *LMP* hourly trends. The only missing part resides in the "most congestible" adjective, since the aforementioned demonstration deals with actually congested lines. Nevertheless, this difference is simply due to the temporal shift that always characterizes the zonal configuration, namely, the fact that zonal configurations must be defined in the present, with the currently available information, but then they must work in the future, with unknown system conditions. Therefore, just as future *LMP* hourly patterns are statistically estimated by historical data and future scenarios simulations, which together try to make up for the lack of their exact trends. The future congested lines required for choosing nodal *PTDFs* also are statistically estimated through the same technique. From this it derives the above mentioned talking of "most congestible" lines, instead of actually congested ones.

For these reasons, summarizing, inside this thesis' methodology the previously chosen suitable clustering algorithms are fed as afterwards reported inside Table 3.1.

3.2.2 Suitable clustering algorithms' changes

The suitable clustering algorithms of the here presented methodology, now even classified according to their inputs, still require some changes to attempt fulfilling this thesis' goal, which is always the deterministic definition of an optimal zonal configuration. Because to get this latter, the zonal configurations produced by the aforementioned partitioning methods should comply as much as possible with the zonal configurations' optimality criteria previously listed inside *Section 3.1*. But unfortunately, none of these clustering algorithms is exactly designed for this purpose. Apart from the price differential clustering, which anyway requires some interventions to increase its chance to define an optimal zonal configuration. Therefore, the following lines take up the previous zonal configurations' optimality requirements and show, for each of them, the eventual changes that have been necessary to methodology's algorithms to make them comply with it, acknowledged it was possible.

Table 3.1: Inputs summary table of the methodology's suitable clustering algorithms.

	Number of clusters/*BAs*	**Average LMPs tolerance**	**LMP hourly patterns**	**Nodal PTDFs of most congestible lines**
LMPs-based Matlab K-means	X		X	
PTDFs-based Matlab K-means	X			X
LMPs-based Matlab K-medoids	X		X	
PTDFs-based Matlab K-medoids	X			X
LMPs-based Matlab Hierarchical clustering	X		X	
PTDFs-based Matlab Hierarchical clustering	X			X
LMPs-based Customized K-means	X		X	
PTDFs-based Customized K-means	X			X
LMPs-based Customized K-medoids	X		X	
PTDFs-based Customized K-medoids	X			X
LMPs-based Customized Hierarchical clustering	X		X	
PTDFs-based Customized Hierarchical clustering	X			X
LMPs-based Customized Genetic algorithm	X		X	
LMPs-based Customized Price differential clustering		X	X	

1. **"Any optimal zonal configuration should be as stable as possible from the temporal point of view".**
   There are mainly two ways of fulfilling this requirement according to the scientific literature which deals with the subject. On the one hand in fact, few papers compute zonal configurations by using clustering algorithms only fed with actual historical data. And then they check their *BAs*' temporal stability by comparing their first zonal configurations with new ones exclusively based on future scenarios data. On the other one instead, the majority of papers straightaway computes temporal stable zonal configurations. By immediately giving as input to the respective clustering algorithms both actual historical data and future scenarios ones. Inside this thesis it is adopted this second approach, much more effective as proven by the papers' case studies themselves, but only with the LMPs-based clustering algorithms. Because to use it, these last require historical data and future estimations of *LMP* hourly patterns, which usually can be easily evaluated through power network's historians and system simulations on future scenarios. The PTDFs-based clustering algorithms require historical data and future estimations of nodal *PTDFs*, which are typically hard to define as regards the future *PTDFs*. Because *PTDFs* depend on grid topology, and thereby future scenarios of it would be needed in order to estimate the aforementioned future *PTDFs*. But usually they are not available, as also in this thesis, and thus that estimation becomes impossible. In conclusion, it is still worth remembering that actually the above said historical data of *LMP* hourly patterns or nodal *PTDFs* may derive from system's simulations as well, like normally already do the future estimations of these parameters, instead from real power network's historians. In particular, this is exactly what has been done inside this thesis' methodology, where power network's historical data and future scenarios ones have always been obtained through system's simulations, respectively based on the current and the future most valid scenarios. It is important to provide now this clarification, since some of the following requirements, like the next one, will exactly deal with system's simulations used to produce both power network's historical data and future scenarios ones.

2. **"Any optimal zonal configuration should not hinder network security, for instance intended as N-1 security".**
   This requirement is easily fulfilled by all the methodology's suitable clustering algorithms. By simply reducing the available lines' transmission capacity to 70% within the Matlab script which simulates the power network operation, and thereby gives all its necessary information on historical data and future scenarios ones.

3. **"Any optimal zonal configuration should boost as much as possible market efficiency".**
   This requirement cannot be complied beforehand the partitioning execution, through the manipulation of the associated clustering algorithm. Because to assess the market efficiency of a certain zonal configuration it is firstly necessary to have the *BAs* set itself, so as to somehow evaluate its performance from the economic point of view. Therefore, since the methodology's suitable clustering algorithms cannot

be changed ex-ante to fulfill this requirement, their resulting zonal configurations are judged ex-post through the afterwards described economic efficiency indicators. Which are exactly aimed at evaluating *BAs*' optimality from the here considered market efficiency point of view. In this way, it can be said that the whole of the aforementioned clustering algorithms complies with this requirement.

4. **"Any optimal zonal configuration should cope in adequate way with congestion management, by only having inter-zonal congestions instead of intra-zonal ones".**
   This requirement is automatically respected when using *LMPs* as clustering feature. Because they diverge when a congestion occurs, in particular at the two nodes which span the congested link. Thereby, they permit the zonal configurations deriving from them to have most frequently congested transmission lines as transboundary connections, ready to define optimal inter-zonal congestions instead of unwanted intra-zonal ones. But moreover, having previously shown the direct proportion between *LMPs* difference and the one computed by nodal *PTDFs* of most congestible lines, it can be stated that this requirement is as much satisfied when using the PTDFs-based suitable clustering algorithms. For these reasons, the whole of the methodology's partitioning methods complies with this requirement.

5. **"Any optimal zonal configuration should be obtained by an easy to understand method".**
   According to the thesis' author, by also referring to the level of the clustering algorithms normally adopted inside the scientific literature which deals with the subject of defining optimal zonal configurations, it can reasonably be affirmed that all the methodology's suitable clustering algorithms are simple enough to comply with the requirement in question.

6. **"Any optimal zonal configuration should limit as much as possible the number of *BAs*".**
   According to the considerations previously reported inside *Section 3.2.1*, the number of *BAs* is a particularly hard-to-find input which unfortunately has always to be user-defined and can only be suggested by *TSO*'s needs or some parameters' trends. Therefore, since anyway its final assignment depends on the user, it could be said that this requirement's fulfillment is always up to the user's responsibility. Nevertheless, this thesis' methodology wants to do more to render this achievement more likely. And hence it provides a criterion to define the *BAs* number's extremes, which could ultimately meet the here considered requirement. Consequently, the minimum number of *BAs* should obviously be two, since having just one price zone would mean having a uniform configuration instead of a zonal one. Whereas, the maximum number of *BAs* should reasonably be lower than about the 5% of the power network's nodes, as regards national electricity grids, or about the 8% of the same, as regards continental electricity grids. This because these last experimentally reveal to be sensible upper limits for keeping the respective zonal configurations away from the nodal ones and their relative unwanted drawbacks. In fact, by looking

for instance at the Italian transmission grid, as closer to this thesis' author sensitiveness, it is made up of 873 nodes, according to Terna's data updated to 2013 on electricity power stations. Therefore, a number of price zones equal to the 5% of the power network's nodes would mean having a national zonal configuration with 44 *BAs*. This reasonably becomes the upper limit for defining a new and optimal zonal configuration aimed at improving system's performance, since the currently existing starting one is composed of only six zones, namely more than seven times less. Whereas, by looking at the reduced model of the European transmission grid which will be afterwards used within the case study, it is made up of 257 nodes. Therefore, a number of price zones equal to the 8% of the power network's nodes would mean having a continental zonal configuration with 20 *BAs*. Which again becomes a reasonable upper limit for defining an optimal zonal configuration. Because slightly bigger than the 13 *BAs* that would have emerged if it had been considered also here the upper limit as the 5% of power network's nodes, like in the previous national electricity grids. Which would have not reckoned the existence of transboundary energy exchanges and relative congestions. The so created *BAs* number's extremes permit to consider satisfied the here considered optimality requirement for all the methodology's suitable clustering algorithms. In particular, they will afterwards represent in the fourth chapter the limits of the zonal configurations' assessment criteria trends according to price zones number, which will be used to suggest the input number of *BAs* to each of the adopted methods.

7. **"Any optimal zonal configuration should limit as much as possible the within-clusters variance of *LMPs*".**
   This requirement is automatically respected when using LMPs-based clustering algorithms. On the one side, the K-means clustering, the K-medoids one and the hierarchical one are traditional clustering techniques. Therefore, they naturally make the clusters by merging highly similar observations and keeping divided different ones. Acting this way, they unavoidably end up defining data groups, which then could become *BAs*, with small inner variance of the user-defined clustering feature, here represented by the *LMPs*. On the other side, the genetic algorithm and the price differential clustering have respectively an objective function exactly aimed at minimizing the within-clusters *LMP* variance for each hour, and a clustering process which maybe merges couples of nodes only whether their average *LMPs* difference drops below a certain user-defined range of tolerance, which actually still makes them produce price zones with small *LMP* variance inside them. With regard to PTDFs-based clustering algorithms instead, the same things cannot be said. Because their traditional clustering algorithms, which are remembered to be the only one existing in the methodology's PTDFs-based partitioning techniques, obviously produce clusters with small inner *PTDFs* variance, for a reasoning similar to the previous one, but simultaneously nothing can be said on their inner *LMP* variance. As a result, in order to still make these PTDFs-based clustering algorithms comply with the here considered optimality requirement, their resulting zonal configurations are judged ex-post through the clustering validity indicators afterwards described. These criteria, fed with the *LMP* hourly trends, are exactly able to evaluate the

within-clusters *LMP* variance, so as to limit it as expressed by this requirement. At this point though, it is worth remembering that this last judgment, even if surely used for making the PTDFs-based suitable clustering algorithms fulfill the aforementioned optimality criterion, is then anyway applied to all the methodology's partitioning techniques. Because the zonal configurations' assessment criteria are introduced not only to check the *BAs*' optimality, which would hence justify the above said evaluation of PTDFs-based methods, but also to allow the comparison among the different price zones definition techniques.

8. **"Each *BA* inside any optimal zonal configuration should have a minimal size, i.e. number of substations".**
   First of all, in the previous description about zonal configurations' optimality criteria, contained in *Section 3.1*, it has already been noted that this requirement is closely linked to the previous one. In fact, when increasing the number of *BAs* while keeping the power network dimensions fixed, the price zones' sizes simultaneously decrease. Therefore, as well as the previous requirement's fulfillment has been initially entrusted to the user's responsibility, here it could be done the same. Nevertheless, within this methodology it has been inserted an additional check on this optimality requirement. In fact, all the zonal configurations produced by the various suitable clustering algorithms are always subject to a "handwritten" function, named "NoSingleNodesBAs", before being given on screen as clustering results. The latter function firstly looks for single-node *BAs* inside the input zonal configuration, and then removes these last if present by merging the respective nodes to the geographically nearest price zones. Therefore, thanks to this additional check in post processing, it can firmly be stated that the optimality requirement here considered, dealing with price zones' minimum dimensions, is surely satisfied by all the methodology's suitable clustering algorithms. Because its fulfillment is no more left to user's responsibility but is automatically achieved by an ad-hoc change of the partitioning methods, just like it has been done for the previous optimality requirement through the *BAs* number's extremes. The only drawback is that, the usage of the just described handwritten function can produce final zonal configurations characterized by a number of price zones smaller than the one initially given by the user as input. This however can be neglected, since experimentally this decrease reveals to be always limited to few price zones and because there are no particular reasons to behave strictly towards the *BAs* number, given the fact that the *BAs* number has already been recognized as an input without a clearly definable optimal value.

9. **"Each *BA* inside any optimal zonal configuration must be made up of physically linked nodes".**
   This is one of the most hard-to-satisfy optimality requirements, since the overwhelming majority of the suitable clustering algorithms completely neglects this criterion when making clusters, but simultaneously it is one of the most important ones too, because having physically detached *BAs* renders unfeasible and hence useless the produced zonal configuration, for reasons previously shown inside *Section 3.1* The

only clustering algorithm which fortunately makes an exception, by naturally fulfilling the here considered requirement, is the price differential clustering, because it merges couples of nodes only when their average *LMPs* difference drops below a certain user-defined range and at the same time they result to be physically connected. As far as all the other clustering algorithms are concerned instead, manual alterations of them are required to only produce physically connected and hence feasible price zones. In this sense, many of the scientific literature's papers regarding the optimal *BAs* definition through the use of clustering algorithms, which have been previously described inside *Chapter 2*, affirm that using a penalty factor to increase the distance of physically detached nodes would impede their merge and thereby would make the respective partitioning method comply with the here considered optimality requirement. Therefore, in order to assess the effectiveness of this technique, as previously said inside *Section 3.2* the clustering algorithms both based on a distance matrix, which hence can be modified through the penalty factors in question and available in terms of source code, necessary to change the clustering process, are altered this way inside ad-hoc customized versions, which consequently theoretically fulfill the here considered optimality requirement. These last are, as already mentioned, the customized K-means clustering, the customized K-medoids clustering and the customized hierarchical one. As regards the genetic algorithm instead, also included inside the customized algorithms because not available in terms of Matlab command like the price differential clustering, as previously better explained inside *Section 3.2*, the aforementioned distance matrix is not present and hence seems to prevent the application of the just described penalty factor technique. But actually, thanks to slight modifications it is still adopted also there, thus making the genetic algorithm too comply with the here considered optimality requirement. In fact, the only difference inside this latter case is represented by the summing of a penalty factor to the objective function, instead of to the distances of physically detached nodes, every time it is associated to a zonal configuration made up of physically detached *BAs*, so as to decrease its fitness and hopefully oust it from the surviving solutions. For these reasons, all the methodology's suitable clustering algorithms, apart from those based on Matlab commands, seem to respect the optimality requirement in question. Therefore, in order to make also these last conform to the here considered criterion, an ad-hoc handwritten function named "CheckBAsConnection" has been developed. This latter, once received as input a zonal configuration, checks its *BAs*' physical integrity by using the power network's adjacency matrix (namely a *NxN* symmetrical matrix, with *N* equal to the number of system's nodes, which has 1 between physically linked nodes and 0 otherwise). Then, if it finds some detached price zones, it separates them into completely different *BAs*, so obtaining a surely feasible final zonal configuration. Thanks to this change of zonal configurations in post processing, also the methodology's suitable clustering algorithms based on Matlab commands are now claimable conform to the here considered optimality requirement. The only drawback is that the usage of the just described additional function can produce final zonal configurations characterized by a number of price zones larger than the one initially given by the user as

input. This however can be neglected, since experimentally this increase reveals to be always limited to few price zones, and because there are no particular reasons to behave strictly towards the BAs number, given the fact that it has already been recognized as an input without a clearly definable optimal value.
Finally, it is still worth remembering one thing before proceeding to the next optimality requirement: the above described additional function aimed at checking BAs' physical integrity is not only capable of treating zonal configurations produced by methodology's Matlab clustering algorithms, but it can also operate on any other BAs set. That is because it is an additional function external to the partitioning processes, which hence acts on their results without considering how they have been actually obtained. It is important to emphasize now this fact because, if within the subsequent case study the aforementioned penalty factor technique will reveal to be unable to really make the customized clustering algorithms only produce physically connected BAs, it will be just necessary to pass also these last's zonal configurations through the above "CheckBAsConnection" function in order to surely make them comply with the here considered optimality requirement. Without compromising the reason for their presence among the methodology's suitable clustering algorithms. For this reason, since nothing has already been stated regard the case study and hence nothing can currently be said upon the above penalty factor technique's effectiveness too, the following summary table of the changes applied to the methodology's algorithms will contain, inside the column referred to the here considered optimality requirement, both the modifications as far as the customized partitioning techniques are concerned (except for the price differential clustering that albeit customized automatically satisfies this optimality requirement). On the one side, the internal change to the clustering process represented by the aforementioned penalty factor technique, still devoid of a sure effectiveness. And on the other one, the possible additional function to insert downstream in case of insufficiency of the previous measure. Which anyway will always remain, at least making lighter the zonal configurations' alteration made by the function in question.

10. **"Each node inside any optimal zonal configuration must belong to only one *BA*".**
This optimality requirement is automatically fulfilled by all the methodology's suitable clustering algorithms. Because all of them are hard clustering techniques, which hence strictly put each of the input database's observations inside one and only one cluster. This is also the reason why inside the previously chosen suitable clustering algorithms is missing a fuzzy-c-means, despite being sometimes used inside the scientific literature dealing with this thesis' subject. In fact, this latter is a soft clustering technique. Therefore, it does not uniquely place each of the input database's observations inside a sole cluster like the previous hard clustering algorithms do, rather, it gives to each of them a number of membership grades, between zero and one and with unitary sum, equal to the user-defined number of clusters. Consequently, using a fuzzy-c-means as clustering algorithm aimed at defining an optimal zonal configuration, would mean having to add something to fulfill the here considered optimality requirement, otherwise not naturally satisfied. Like a downstream

process, actually used inside the papers which adopt this partitioning method for the optimal *BAs* definition, in which the final matrix of membership grades is queried to put each of the system's nodes inside the price zone, namely the cluster, to which it has the highest membership grade. Nevertheless, since the fuzzy-c-means clustering is actually a K-means clustering apart from the aforementioned matrix of membership grades, it is much more reasonable to avoid using this soft clustering technique in favor of a K-means clustering. This automatically satisfies the here considered optimality requirement, without needing anything in post processing.

In order to conclude this section, the changes of methodology's suitable clustering algorithms are summarized in Table 3.3, whose legend is provided in Table 3.2.

Table 3.2: Legend of the modifications summary table of the methodology's suitable clustering algorithms.

Symbol	Meaning
O	Optimality requirement impossible to satisfy for the clustering algorithm
-	Optimality requirement naturally satisfied by the clustering algorithm
Xi	Optimality requirement by the ex-ante *inputs* alteration or simply thanks to their usage
Xo	Optimality requirement satisfied by the ex-post *outputs* alteration or evaluation. Obtained respectively through the use of additional handwritten functions, downstream to the clustering process, or of ad-hoc assessment criteria
Xp	Optimality requirement satisfied by the internal alteration of the clustering *process* itself
1	Zonal configuration temporal stability
2	Power network security (N-1)
3	Zonal-based market efficiency boost
4	Inter-zonal congestions rather than intra-zonal ones
5	Clustering algorithm simplicity
6	Limited number of *BAs*
7	Minimal within-clusters *LMP* variance
8	*BAs* minimal sizes
9	*BAs* physical integrity
10	*BAs* membership exclusivity

Table 3.3: Modifications summary table of the methodology's suitable clustering algorithms.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
LMPs-based Matlab K-means	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xo	-
PTDFs-based Matlab K-means	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xo	-
LMPs-based Matlab K-medoids	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xo	-
PTDFs-based Matlab K-medoids	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xo	-
LMPs-based Matlab Hierarchical clustering	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xo	-
PTDFs-based Matlab Hierarchical clustering	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xo	-
LMPs-based Customized K-means	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xp/(Xo)	-
PTDFs-based Customized K-means	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xp/(Xo)	-
LMPs-based Customized K-medoids	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xp/(Xo)	-
PTDFs-based Customized K-medoids	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xp/(Xo)	-
LMPs-based Customized Hierarchical clustering	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xp/(Xo)	-
PTDFs-based Customized Hierarchical clustering	O	Xi	Xo	Xi	-	Xi	Xo	Xo	Xp/(Xo)	-
LMPs-based Customized Genetic algorithm	Xi	Xi	Xo	Xi	-	Xi	-	Xo	Xp/(Xo)	-
LMPs-based Customized Price differential clustering	Xi	Xi	Xo	Xi	-	Xi	-	Xo	-	-

3.3 Assessment criteria for zonal configurations

Having theorized the target features desired for the output, within the first section of this chapter, and moreover having chosen the apparently most suitable techniques to find results skilled that way, inside the second section of the same, the questions become now how to evaluate the quality of the produced outputs and how to compare the performance of the different approaches used to obtain them. The answer to these queries comes from the subsequently described third level of this thesis' methodology. In fact, the following lines provide a series of zonal configurations' assessment criteria. These criteria are intended to both evaluate the *BAs* optimality, according to some of the previously reported optimality requirements which thereby become satisfied for the associated clustering algorithm (assuming it was still necessary), and to allow the comparison among the different price zones definition techniques adopted inside the here outlined methodology.
These zonal configurations' assessment criteria are discernible in two categories, namely the clustering validity indicators and the economic efficiency ones.

3.3.1 Zonal configurations' clustering validity indicators

These indices are commonly used when evaluating the goodness of a clustering result. When they are fed with the user-defined clustering feature, they give a measure of its within-clusters variance, which is remembered being small when the partitioning process is successful. Since normally a clustering algorithm merges the highly similar observations of the input database and simultaneously keeps distinct the markedly different ones. In particular, these clustering validity indicators directly proportional to the within-clusters variance of the input parameter are:

Mean Index Adequacy (MIA).

Clustering Dispersion Indicator (CDI).

Similarity Matrix Indicator (SMI).

Davies-Bouldin Index (DBI).

An in-depth description of each of them is provided inside reference [60]. For the here described *BAs* evaluation goal, they are always fed with the *LMP* hourly patterns, whichever is the user-defined clustering feature within the suitable clustering algorithm that has produced the under judgment price zones. This because one of the zonal configurations' optimality requirements, previously reported inside *Section 3.1*, exactly asks to limit as much as possible the within-clusters *LMP* variance. Therefore, by actually using nodal prices to fed the aforementioned clustering validity indicators with both PTDFs-based and LMPs-based partitioning techniques, firstly there can be an interesting comparison between their performance based on a zonal configurations' optimality criterion. Moreover, this latter can be fulfilled by the PTDFs-based clustering algorithms as well, which otherwise would not naturally satisfy it as previously better explained inside *Section 3.2.2*. Finally, it is still worth remembering that, since all the above mentioned parameters are directly proportional to the within-clusters variance of their input parameter, during

the following assessments the lower these indices will be the more optimal the evaluated zonal configuration will prove to be. Especially, since each of these parameters will case by case be normalized respect its maximum value among the zonal configurations' ones of the compared clustering algorithms, each of these indices will have a unitary value in correspondence of its worst zonal configuration, several minor values in correspondence of the intermediate ones, a minimum value in correspondence of its best zonal configuration.

3.3.2 Zonal configurations' economic efficiency indicators

As regards the second category of zonal configurations' assessment criteria which is made up of economic efficiency indicators, these last are indices commonly used to make market power evaluations. That is because one of the zonal configurations' optimality requirements, previously reported inside *Section 3.1*, actually asks to boost as much as possible the market efficiency of the zonal configuration in order to make it optimal. And hence, for this latter's validation by all the methodology's algorithms and for these last's comparison, the above economic efficiency indicators become useful. In fact, theoretically the most efficient market ever is the one characterized by an absolute perfect competition, and because this latter parameter is the opposite of the aforementioned market power. Therefore, going down in particular, the here adopted economic efficiency indicators are:

Concentration ratio (R_m). With m equal 4, as this is the most frequently used version of the index. According to which there is perfect competition under $R_4 = 0.0001$ and monopoly over $R_4 = 0.71$.

Herfindahl-Hirschman Index (HHI). According to which there is perfect competition with $HHI = 10000/P$, where P represents the number of market's producers, and monopoly with $HHI = 10000$.

Entropy Coefficient (EC). According to which there is perfect competition whether $EC = ln(P)$, where P again represents the number of producers, and monopoly when $EC = 0$.

Local Herfindahl-Hirschman Index (L-HHI). According to which there is perfect competition with $L\text{-}HHI = 0$ and monopoly with $L\text{-}HHI = 1$.

An in-depth description of each of them is provided inside reference [57]. Among these, the first and the fourth index have a unitary range, where 0 and 1 respectively represent the best and the worst situation according to the above cited zonal configurations' optimality requirement about market efficiency. Consequently, they represent again the same situation already seen for the previous clustering validity indicators, once normalized respect to their maximum values among the compared algorithms. Therefore, it would be useful to bring also the *HHI* and the *EC* in the same rating scale, so as to ease the partitioning methods' evaluation and comparison. Consequently, on the one hand the *HHI* is considered divided by its maximum possible value, namely $HHI = 10000$ representing a monopolistic market, in order to have once more a unitary range where 0 and 1 respectively represent the best and the worst situation. And on the other one the *EC* is

considered divided by its maximum possible value too, namely $EC = ln(P)$ representing a perfect competition market, ending up having again a unitary range. Nevertheless, this latter has 0 and 1 respectively on the worst and the best situation according to the above cited zonal configurations' optimality requirement about market efficiency. Therefore, it is still necessary to reverse its scale, by considering $1 - EC/ln(P)$ as real entropy coefficient, in order to have once more the same rating scale of the other indicators. At this point, that all the aforementioned economic efficiency indicators have the same unitary range with 0 and 1 respectively best and worst situation, it is worth mentioning that making also a normalization for each of them respect to their maximum values among the compared algorithms, like previously done for the clustering validity indicators, would be counterproductive in this case. In fact, the just outlined parameters already have a unitary rating scale, unlike the prior ones before their normalization, and moreover this latter holds a physical sense in its extremes, since 0 and 1 respectively represent a perfect competition or a monopoly. Therefore, making here the above said normalization would only mean losing part of the information, without acquiring any advantage in terms of algorithms' evaluation or comparison. In fact, the resulting economic efficiency indicators would maintain their unitary rating scale, as they already had before the normalization, and would always have a quartet of unit values among the compared clustering algorithms. Which, however, would not reveal the presence of monopolies, but would only indicate the worst partitioning methods according to the respective economic efficiency indicators. For this reason, in order to preserve the physical sense of their unitary rating scale, this second category of zonal configurations' assessment criteria is not normalized respect to the maximum values among the compared algorithms. Thus, having one of these indicators equal 1 or 0 would always mean having a monopoly or a perfect competition.

Eventually, it must be considered that the R_4, the *HHI* and the *EC* are only capable of evaluating the market power presence inside uniform-based markets, and not zonal-based ones. Therefore, for the here described zonal configurations' evaluation goal, they will be applied once on each price zone of the assessed *BAs* set, so having a triple measure of the perfect competition level for each of them. Then, among these last it will be kept only the highest value for each of the evaluation parameters, namely the one referring to the most monopolistic and hence worst price zone according to it. Instead, as far as the *L-HHI* is concerned, it is the only economic efficiency indicator exactly designed for evaluating the market power presence inside a zonal-based market. In fact, it becomes 1 whether inside this latter there are various *BAs* all characterized by absolute monopolies or there is a sole price zone which holds the whole of the market shares and is an absolute monopoly. Therefore, it is the only assessment parameter which manages to evaluate the market power presence on two levels, i.e. both within and between the various *BAs*. However, the *L-HHI* has still to be firstly computed once per each price zone and secondly kept for its maximum value, by definition for having a market power evaluation of the total zonal-based market. As a result, in the following economic assessments of the zonal configurations, each of these last will be represented by one value per each of the four aforementioned economic efficiency indicators. The first three of these last will refer to the zonal configuration's worst *BAs* according to them, whereas the last one will give an overall evaluation of the zonal-based market's efficiency.

Chapter 4

Case Study

The thesis' methodology previously described inside *Chapter 3*, aimed at locating the most suitable clustering algorithm for deterministically define an optimal zonal configuration, is here below applied to a real power network model, in order to test its effectiveness. Therefore, this chapter will initially give a portrait of the used electricity grid model inside *Section 4.1*. Then it will make some a priori considerations within *Section 4.2*, useful to make before the methodology execution so as to prevent the repetition of some of its parts, which unavoidably would become necessary at the end of the same if the aforementioned a priori considerations were not be done beforehand. And eventually it will deal with the actual methodology application inside *Section 4.3*, together with the associated considerations.

4.1 Description of the case study's electricity grid

This section contains a description of the power network model on which it has been afterwards applied the thesis' methodology. In particular, it is a reduced model of the European transmission electricity grid composed of 257 nodes, that has been obtained by applying a K-means clustering algorithm to the original 380 kV European network made up of more than 6000 bus. Hence, this model spans 33 out of 36 *ENTSO-E* countries, which are inter alia: Albania, Austria, Bosnia and Herzegovina, Belgium, Bulgaria, Estonia, Finland, Croatia, Czech Republic, Denmark, France, Germany, Great Britain, Greece, Hungary, Ireland, Italy, Latvia, Lithuania, Norway, Poland, Portugal, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden and Switzerland. Thanks to the above cited K-means clustering, it ends up having 460 branches, 24 DC lines, 1448 generators and roughly 360 GW of load, scattered on the above said 257 nodes. Inside Fig. 4.1 it is provided a representation of the electricity grid model in question.

The choice of the power network model to be used for testing the thesis's methodology has fallen on this one for two reasons. On the one hand because it concerns an area large enough for giving to the following considerations, on the previously chosen methodology's suitable clustering algorithms, the generic meaning desired for this thesis' goal. On the other hand, because it has been used inside reference [56] for creating a Matlab program able to simulate its hypothetical day ahead market, on the basis of bids and offers referred

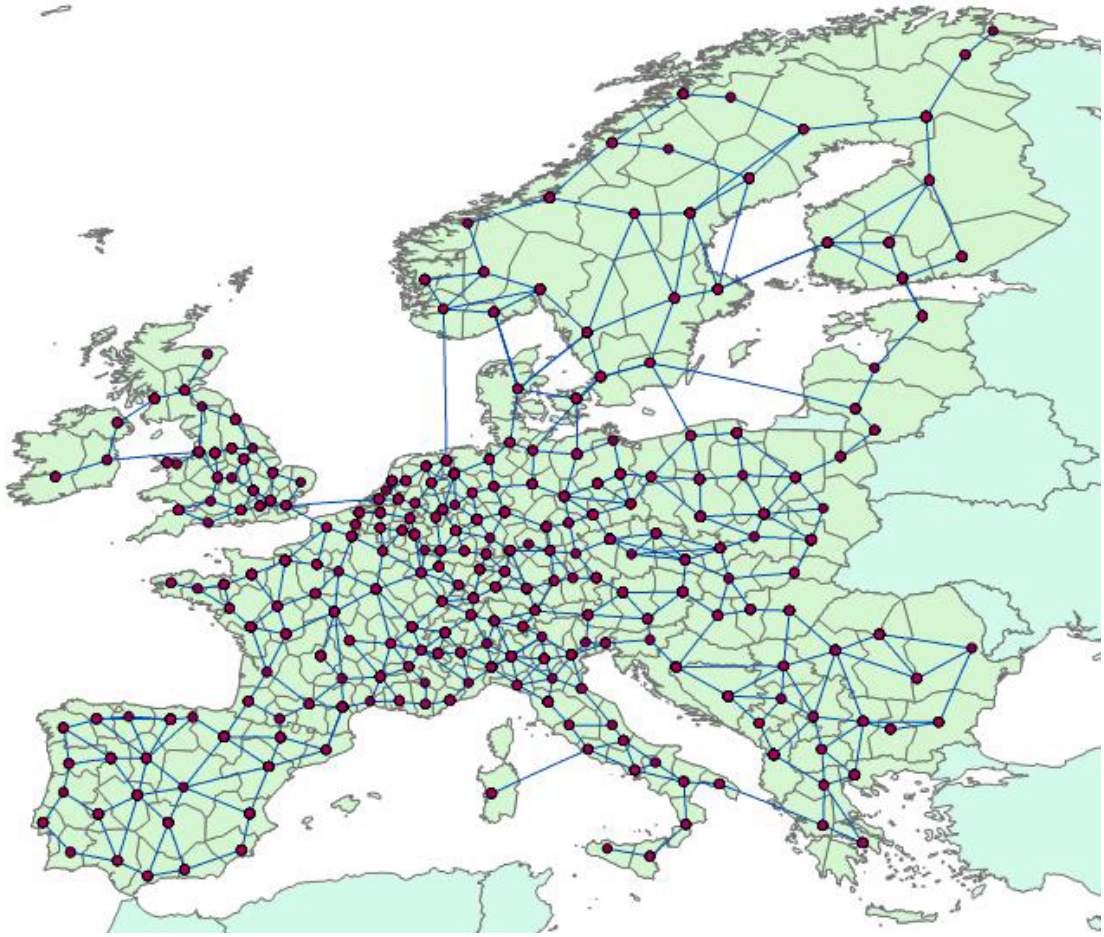

Figure 4.1: Map of the case study's reduced model of European transmission network.

to a specific user-defined scenario. This latter in fact, based on Matpower, becomes a particularly powerful tool for this thesis' needs. Since it manages to provide all the data required as input by the methodology's algorithms, from the *LMP* hourly patterns to the nodal *PTDFs* of the most congestible lines. The only drawback to this apparently perfect situation, is represented by the computation time of the aforementioned script. Indeed, a 24-hour market simulation actually requires 8 hours on a 64-bit desktop PC driven by a i5-6500 processor of 3.19 GHz and equipped with 16 Gb of RAM. Therefore, the initial idea of simulating two years, respectively arising from the current and the future most valid scenario among the available ones (to be used for zonal configurations' temporal stability reasons previously explained inside *Section 3.2.2*), has been unavoidably abandoned in favor of a more reasonable simulation time of a week. As a result, the Matlab program in question has been run twice, with the following settings:

- 1^{st} run: 2017's scenario from Monday 8 May to Sunday 14 May.
- 2^{nd} run: 2040DG's scenario from Monday 8 May to Sunday 14 May.

Both times the extracted results have been the *LMP* hourly patterns along the week and the list of at-least-once congested lines over the simulation time. The choice of May and that particular week has been done in order to consider a continental load level as close as possible to its annual average value, so as to imitate at least the original idea of using years as simulation time. In this sense, the aforementioned week does not contain any particular festivity at European level, and its typical weather on the continent is markedly away from the extremes summer and winter. Thereby, the continental load level of that period is actually close to its annual average value, as proven by the load profiles provided inside reference [56], which inter alia reveal a maximum and a minimum European load level respectively during winter and summer. Instead, with regard to the second run's scenario year, it has been chosen the 2040DG because inside the Matlab program in question it is the latest future scenario available which represents a prosumer-centric development, namely, a system's growth oriented towards events as the distributed generation increase or the electrical vehicles penetration. The actual realization of this scenario indeed, would likely sharpen the congestions problem and consequently the associated nodal prices divergences. Thus, since the most congestible lines on the one side and the *LMPs* on the other one represent the two input kinds which can be used to feed clustering algorithms aimed at defining optimal *BAs* sets, it is better to consider beforehand this scenario, during the optimal zonal configurations definition itself, so as to make the resulting price zones fulfill their optimality requirement about temporal stability even in case this particularly challenging future scenario is realized. Finally, in order to conclude this overview on the electricity grid model which will be used in the here outlined case study, two things must still be said. Firstly, as better explained inside *Section 3.2.2* to which reference is made for further details, during both the aforementioned Matlab program's runs the transmission capacity of the power network's lines has been reduced of 30%, to comply with the zonal configurations' optimality requirement regarding the application of the N-1 security criterion. Secondly, it would have been better if the future scenario (i.e. the 2040DG) had been endowed with a prospective on the European electricity grid future topology, and not only with estimations on future bids, offers and lines' transmission capacities, this latter

evaluated downstream to the *ENTSO-E*'s investment plan on the continental power network up to 2040. Acting that way, it would have been obtained also an estimation on the future system's *PTDFs*, otherwise only present for the future *LMP* hourly patterns, which could have been used together with the 2040's list of the at-least-once congested lines in order to have a future estimation of the nodal *PTDFs* of most congestible lines. This could have made the PTDFs-based methodology's suitable clustering algorithms comply with the previously reported zonal configurations' optimality requirement on temporal stability, that otherwise is only fulfilled by the LMPs-based methodology's algorithms, as better explained inside the previous *Section 3.2.2*. Beyond these auspices, this future topology estimation is not available in the power network model in question. Therefore, the *PTDFs* matrix does not vary from one scenario to the other, and consequently it is only computed once, by arbitrarily using the ad-hoc Matpower command at the end of the first simulation, and by obviously setting the slack node in correspondence of the one gifted with the highest level of generation. Conventionally, this is done to simulate at best the power network's live keeping of active and reactive power balances, even if actually, this latter behavior is unthinkable to be borne by a sole node, however big its generation may be. Hence in reality the power balance is obtained through the contribution of several nodes of the system, by doing what is technically called a "distributed slack node".

4.2 A priori considerations

This section contains some a priori considerations useful to make before the methodology execution. In fact, these considerations would anyway appear at the end of the discussion, thereby ending up forcing the repetition of some parts of the methodology. These considerations are distinguishable in three, and hence are separately treated within the sections hereafter reported. Inside these last, it is firstly described the insufficiency of the penalty factor technique for defining only physically cohesive *BAs*. Secondly, it is shown the inadequacy of the *GA*'s starting population random initialization. And thirdly, it is faced the K-means and K-medoids initialization problem.

4.2.1 Insufficiency of the penalty factor technique

In *Section 3.2.2*, about the zonal configurations' optimality requirement concerning the price zones' physical integrity, it has been stated that many of the scientific literature's papers, regarding the optimal *BAs* definition through the use of clustering algorithms, affirm that using a penalty factor to increase the distance of physically detached nodes would impede their merge. Thereby, it would make the respective partitioning method comply with the optimality requirement in question. Therefore, as better explained inside *Section 3.2.2*, in order to assess the effectiveness of this approach, the methodology's customized suitable clustering algorithms have actually been endowed with this penalty factor technique (apart from the price differential clustering, which already fulfills naturally this optimality requirement). This has been done hoping to manage to easily define physically feasible zonal configurations, without the need of modifications in post processing through the usage of the handwritten function "CheckBAsConnection" previously described in *Section 3.2.2*. Nevertheless, experimental tests show the absolute inability

of this penalty factor technique in making the customized suitable clustering algorithms comply with the zonal configurations' optimality requirement in question. This happens whichever is the user-defined *BAs* number, that directly or indirectly must be always defined by the user as better explained in *Section 3.2.1*. Proof of this is afterwards provided inside Fig. 4.2, Fig. 4.3 and Fig. 4.4, which respectively portray the zonal configurations of the case study's electricity grid produced by the methodology's customized suitable clustering algorithms with a low, medium and high number of *BAs*. These figures in fact, are already obtained by giving as input to the various partitioning methods the *LMPs* hourly patterns or the most congestible lines' nodal *PTDFs* deriving from the aforementioned 1^{st} and 2^{st} run of the [56]'s Matlab program, and reveal several cases of physically detached price zones. Like the zone 1 and the zone 2 within LMPs-based customized K-means' zonal configuration of Fig. 4.2, the zone 1 within PTDFs-based customized K-means' zonal configuration and PTDFs-based customized K-medoids' one of Fig. 4.2, the zone 4 within PTDFs-based customized K-medoids' zonal configuration and LMPs-based customized hierarchical clustering's one of Fig. 4.3, and the zone 4 within PTDFs-based customized hierarchical clustering's zonal configuration of Fig. 4.4. Therefore, the penalty factor technique reveals to be always insufficient to carry out the purpose for which it has been created. Whether the user-defined *BAs* number is low, namely 5 as in Fig. 4.2, or intermediate, namely 13 as in Fig. 4.3, or maximum according to the limits previously declared inside *Section 3.2.2* (in order to comply with the 6^{th} zonal configurations' optimality requirement), namely 20 as in Fig. 4.4.

Moreover, with respect to these just mentioned three figures, it is worth noting that there are some cases, like the LMPs-based customized K-means' zonal configuration of Fig. 4.3, in which there is a "Final BAs", tacit "number", differs from the input one defined by the user and reported inside "Requested BAs", again tacit "number", box. This could seem to be an incorrectness, since the whole of the here adopted customized algorithms should be only endowed with the aforementioned penalty factor technique according to the overlying lines (reason why the price differential clustering is missing within this section, since as already said it does not include this technique), and hence should be devoid of any zonal configurations' change in post processing. But actually, it is not so. In fact, as better explained inside *Section 3.2.2*, all the methodology's suitable clustering algorithms, and thereby also the customized ones, are downstream modified by an additional handwritten function named "NoSingleNodeBAs". This function is exactly aimed at preventing the presence of single-node price zones inside the produced zonal configurations, that otherwise would compromise the zonal-based market's efficiency due to market power reasons. As a result, this is the reason why within the LMPs-based customized K-means' zonal configuration of Fig. 4.3 there is a "Final BAs" number different and especially lower than the input one defined by the user and reported inside the "Requested BAs" box.

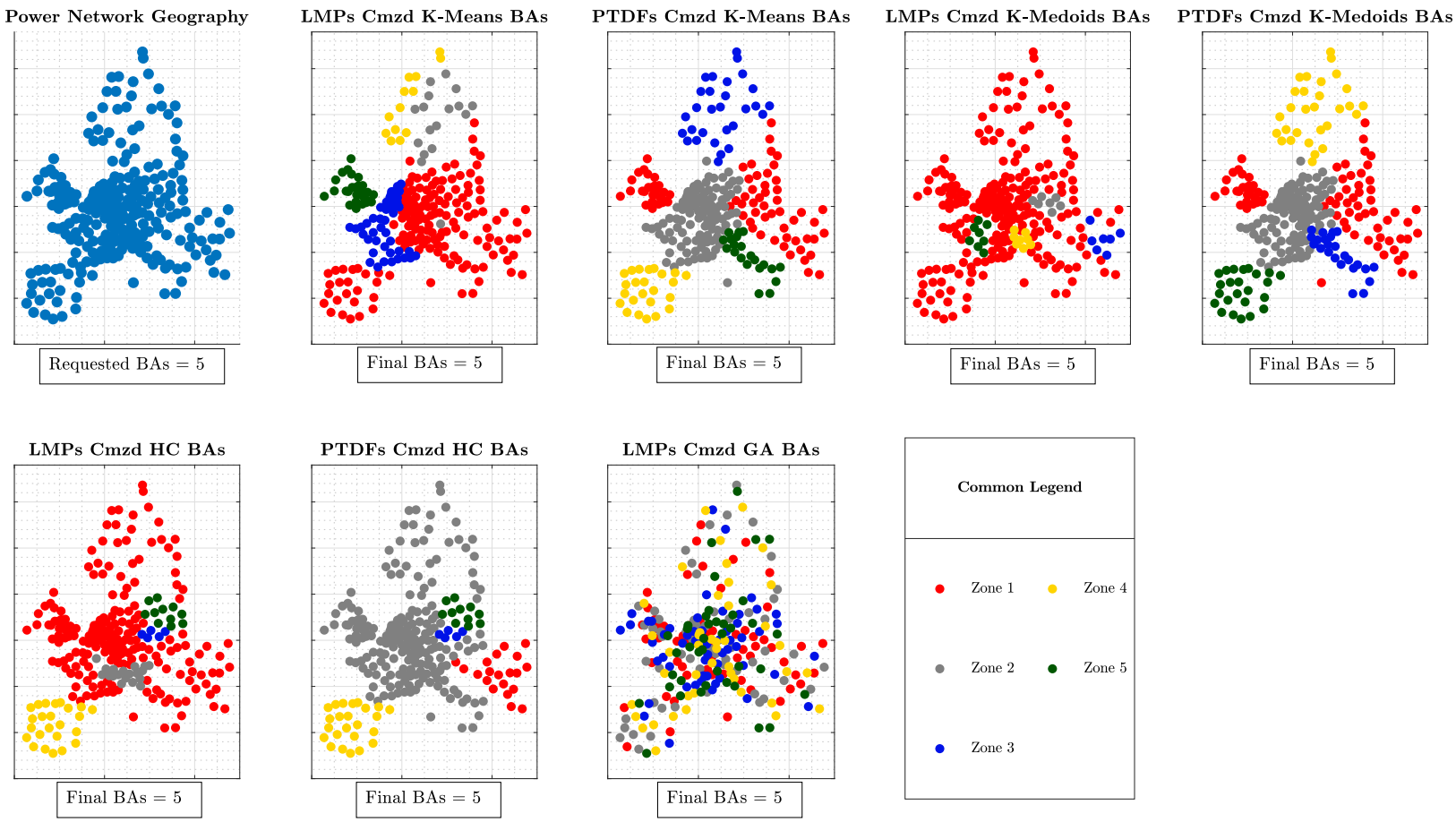

Figure 4.2: European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 5 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function.

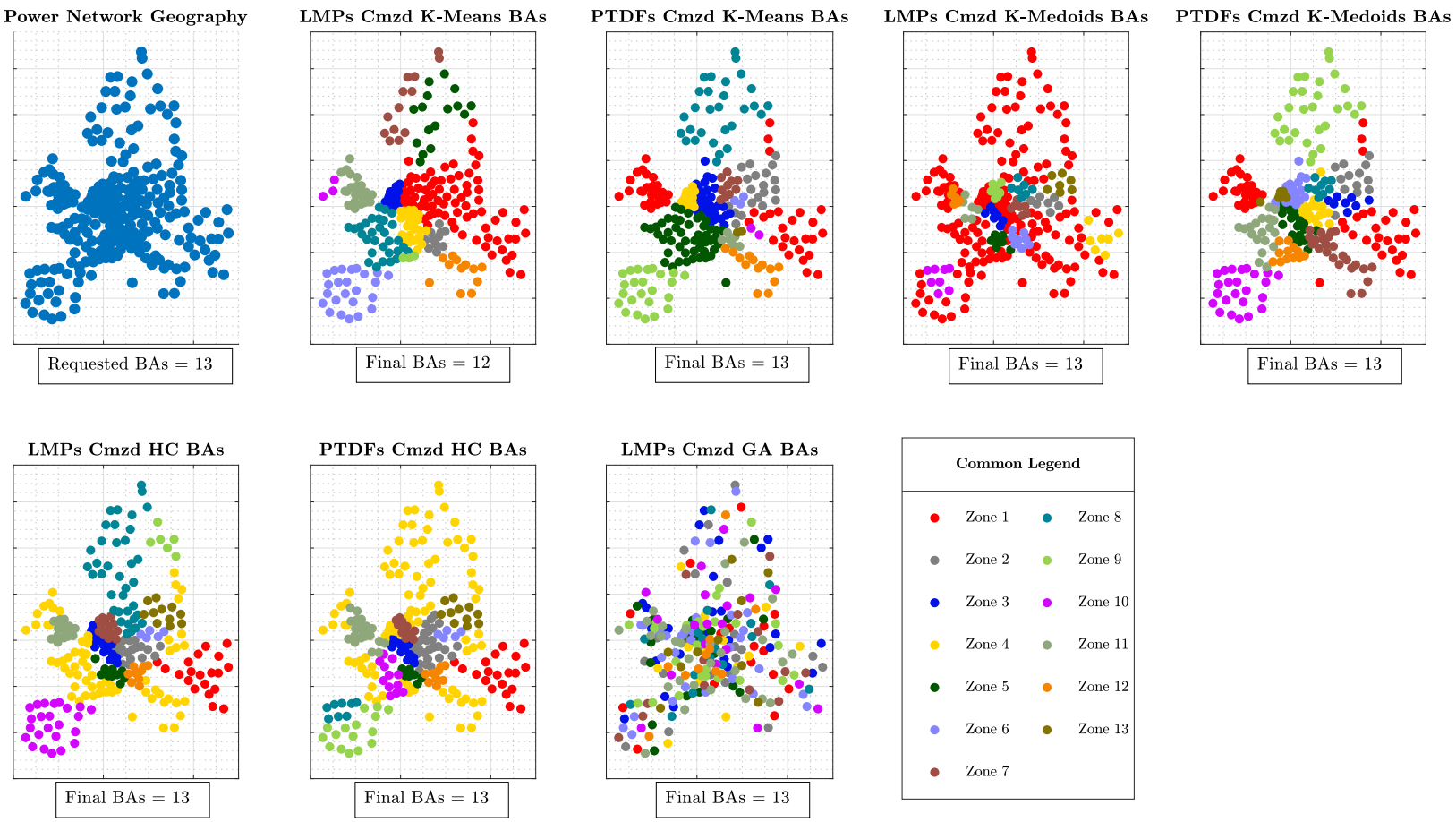

Figure 4.3: European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 13 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function.

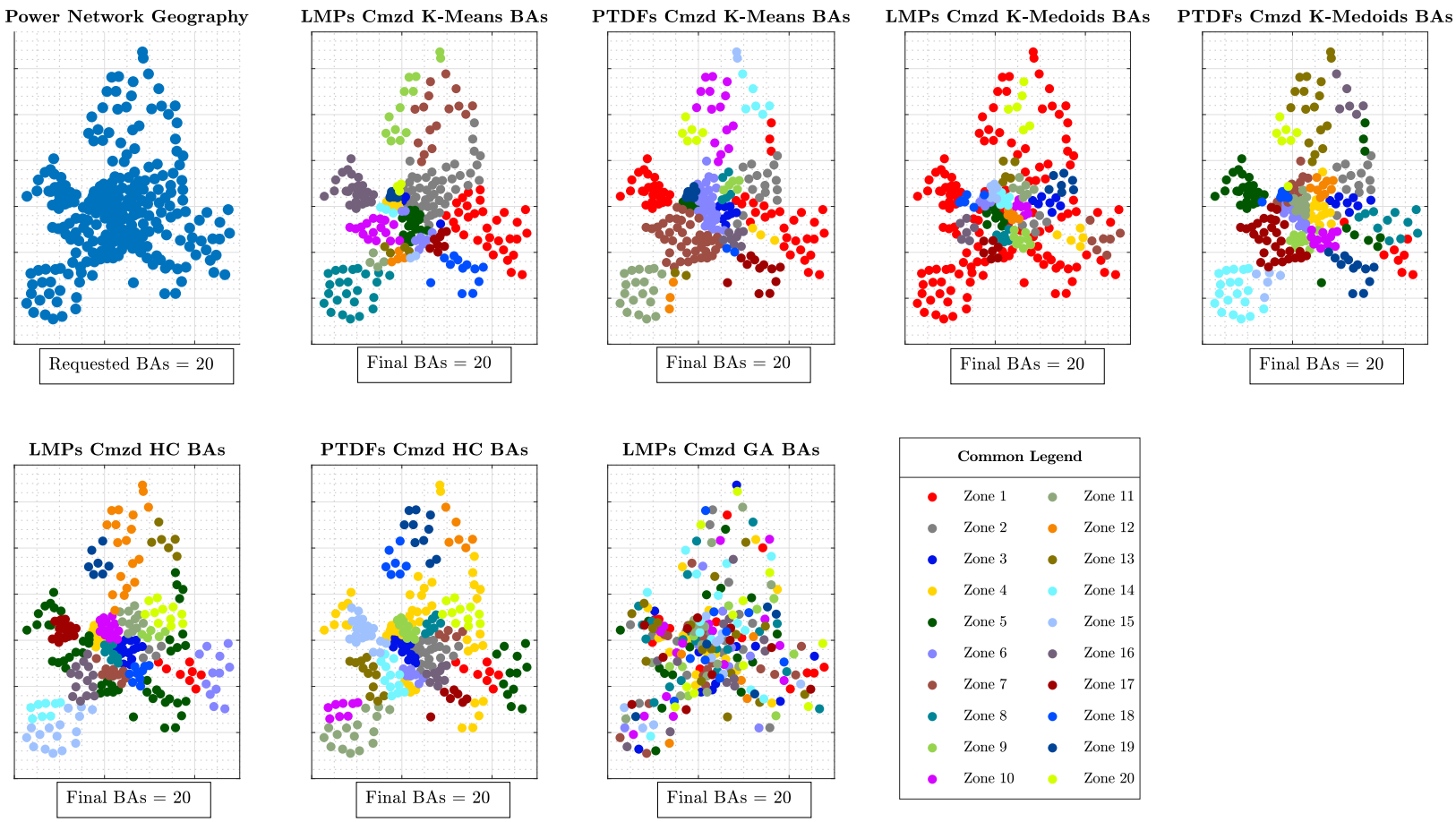

Figure 4.4: European transmission network's zonal configurations produced by the methodology's customized clustering algorithms, with 20 *BAs* requested by the user and no post-processing usage of the "CheckBAsConnection" handwritten function.

For these reasons, having proved the insufficiency of the penalty factor technique for defining only physically cohesive price zones, the aforementioned handwritten function "CheckBAsConnection" is from here on out applied downstream of all the methodology's suitable clustering algorithms apart from the price differential clustering (*PDC*). It was already applied on Matlab's partitioning techniques, and now it has also been enabled on customized algorithms apart from the *PDC*, due to the insufficiency of their penalty factor technique. The only exception remains the *PDC* because, thanks to its nature, it has always automatically complied with the zonal configurations' optimality requirement concerning the price zones' physical integrity, without the need of any penalty factor technique or handwritten function used in post-processing. By acting this way, as already hinted inside *Section 3.2.2* when alluding to the possible penalty factor technique's failure, the methodology's Matlab suitable clustering algorithms will only produce physically feasible zonal configurations thanks to the invasive intervention of the aforementioned additional handwritten function "CheckBAsConnection". Whereas, the customized ones will do the same through a less invasive intervention of the function in question, thanks to a preliminary reduction of the union possibility of physically detached nodes obtained through the above said penalty factor technique.

As a final remark, it is useful to try giving a reason to the just described penalty factor technique's failure. Especially for the future developments, in order to make everybody convinced on it, so as to prevent from making the same error of many scientific literature's papers. Therefore, it is important to note that using a penalty factor to increase the distance of physically detached nodes could not become in principle the definitive solution to avoid their merge. Because this measure does only make less probable this situation, without strictly preventing it. And hence, if the generic clustering algorithm does not find reasonable alternatives to the aforementioned union, it still ends up doing it, thereby producing physically unfeasible zonal configurations. Since one way or the other it needs to find a convergence, with the user-defined number of clusters.

4.2.2 Inadequacy of the *GA*'s starting population random initialization

The genetic algorithm comes into being as an optimization algorithm but, as already explained inside *Section 3.2*, it can be transformed into a clustering algorithm able to potentially define optimal zonal configurations by simply writing an ad-hoc objective function aimed at minimizing the within-clusters *LMP* variance for each hour (which moreover is endowed with the penalty factor technique, as all the customized suitable clustering algorithms except the *PDC* one). However, beyond the effectiveness of this newly defined clustering algorithm in finding optimal *BAs* which will be afterwards assessed inside *Section 4.3*, it first must be noted one thing regarding its setting. In fact, a classical *GA* requires to be defined:

The population size. It defines the number of best chromosomes which are preserved at each iteration. The more it is, the better, as the *GA* expands its solutions' space and thus increases its possibility to find the global optimum. But simultaneously, the growth of this population also increases the algorithm's computational burden. Hence, a trade-off has to be found.

The population initialization. It defines how to create the *GA*'s starting population.

The crossover probability. In a unitary range, it defines the probability to which population's chromosomes are subject to this genetic operator (crossover).

The mutuation probability. In a unitary range, it defines the probability to which chromosomes' genes are subject to this genetic operator (mutation).

The maximum iteration number with the same fittest solution. It is the *GA*'s primary stop criterion. In fact, if the population's fittest chromosome according to the user-defined objective function does not vary for a number of successive iterations equal to this parameter, the *GA* stops and gives it out as optimization result.

The maximum iterations number. It is the *GA*'s secondary stop criterion. In fact, if the algorithm executes a number of iterations equal to this parameter without converging for the aforementioned primary stop criterion, it finally stops and gives out the population's fittest chromosome of that moment as optimization result.

According to what is normally done, these last parameters are set as follows inside the methodology's *GA*:

The population size. Sufficiently high to create a reasonably large solutions' space of the *GA*, and sufficiently small not to threaten the process' performance. Therefore, in this case study's instance of a 257-bus power network, it could be rational to have a population of 300 chromosomes.

The population initialization. Randomly done, as usual in the literature to solve general problems.

The crossover probability. Set to 0.9. Hence quite high.

The mutuation probability. Set to 0.1. Hence quite low, as usual because more than this would make this genetic operation provoke more chaos than benefit.

The maximum iteration number with the same fittest gene. Set to 10. Hence a number in the order of tens, as usually done.

The maximum iterations number. Set to a very high number, much greater than the number of *OP*'s decisional variables as usually done, just to provide an emergency stop criterion to the algorithm. Therefore, in this case study's instance of a 257-bus power network and hence 257 integer decisional variables, it is set to $20 * 257 = 5150$. Namely twenty times the *OP*'s decisional variables.

But, proceeding this way, the methodology's *GA* produces totally senseless zonal configurations, much more similar to a color palette rather than an electricity grid's *BAs* set. And this, was already visible in the previous Fig. 4.2, Fig. 4.3 and Fig. 4.4, in correspondence of the *GA*'s boxes. But moreover, it is still visible in the following Fig. 4.5,

which only portrays the *GA*'s zonal configurations obtained with the same user-defined numbers of *BAs* (namely 5, 13 and 20) but after having enabled the downstream passage through the additional handwritten function "CheckBAsConnection". This function is remembered to be essential to run for all the methodology's suitable clustering algorithms apart from the *PDC*, in order to obtain from them physically feasible zonal configurations due to reasons previously described inside *Section 4.2.1*, whose presence is proven by the "Final BAs" boxes' numbers greater than the "Requested BAs" ones, that furthermore reveal the typical prevalence of the handwritten function "CheckBAsConnection" over the "NoSingleNodeBAs" one.

This phenomenon was realistically predictable in advance, because actually the genetic algorithm is basically an optimization algorithm with a strong random nature, which has just been adapted to a clustering purpose. Therefore, it is surely difficult to succeed in obtaining optimal zonal configurations through its usage. But this latter becomes even harder if the starting population is randomly initialized, even though the algorithm's stop criteria are set to very huge values, which obviously are not acceptable due to the consequences that this action would have on the process' computational burden. For these reasons, in order to actually create zonal configurations instead of color palettes through the methodology's *GA*, it has been decided to steer this latter's initial exploration of the solutions' space. The aim is not to lose its random contribution, anyway useful to try finding a more optimal zonal configuration, which instead would have been lost if its zonal configurations would have only been rejected as senseless, and to still get a reasonable zonal configuration despite of not huge stop criteria. As a result, from here on out the *GA*'s starting population will be whenever half initialized using the zonal configuration coming from the LMPs-based customized K-means, ex-ante the application of both the additional handwritten functions "NoSingleNodeBAs" and "CheckBAsConnection". In order to be sure of having a number of clusters, inside the inherited zonal configuration in question, actually equal to the user-defined one. So as to properly execute the *GA*, whose fittest chromosome will be finally passed anyway through the aforementioned additional handwritten functions. Which are remembered to be essential to make the *GA* comply with the 8^{th} and the 9^{th} zonal configurations' optimality requirement.

4.2.3 The K-means and K-medoids initialization

The previous *Section 4.2.2* has proved the inadequacy of using randomness to initialize the starting population of the methodology's *GA*. However, this is not the only case among the suitable clustering algorithms in which a random initialization must be refused. In fact, also the centroids of Matlab and customized K-means or the medoids of Matlab and customized K-medoids require to be initialized somehow. Doing it randomly is not the most correct approach, for a twofold reason. On the one hand, because choosing them without any physical or numerical criterion could lead the final zonal configurations being senseless, as the previous *GA*'s ones during its population's random initialization, or at least less performing than *BAs* sets based on smart criteria. On the other hand, because choosing them in a random way would deprive the following findings of any repeatability, fundamental piece of whatever scientific research.

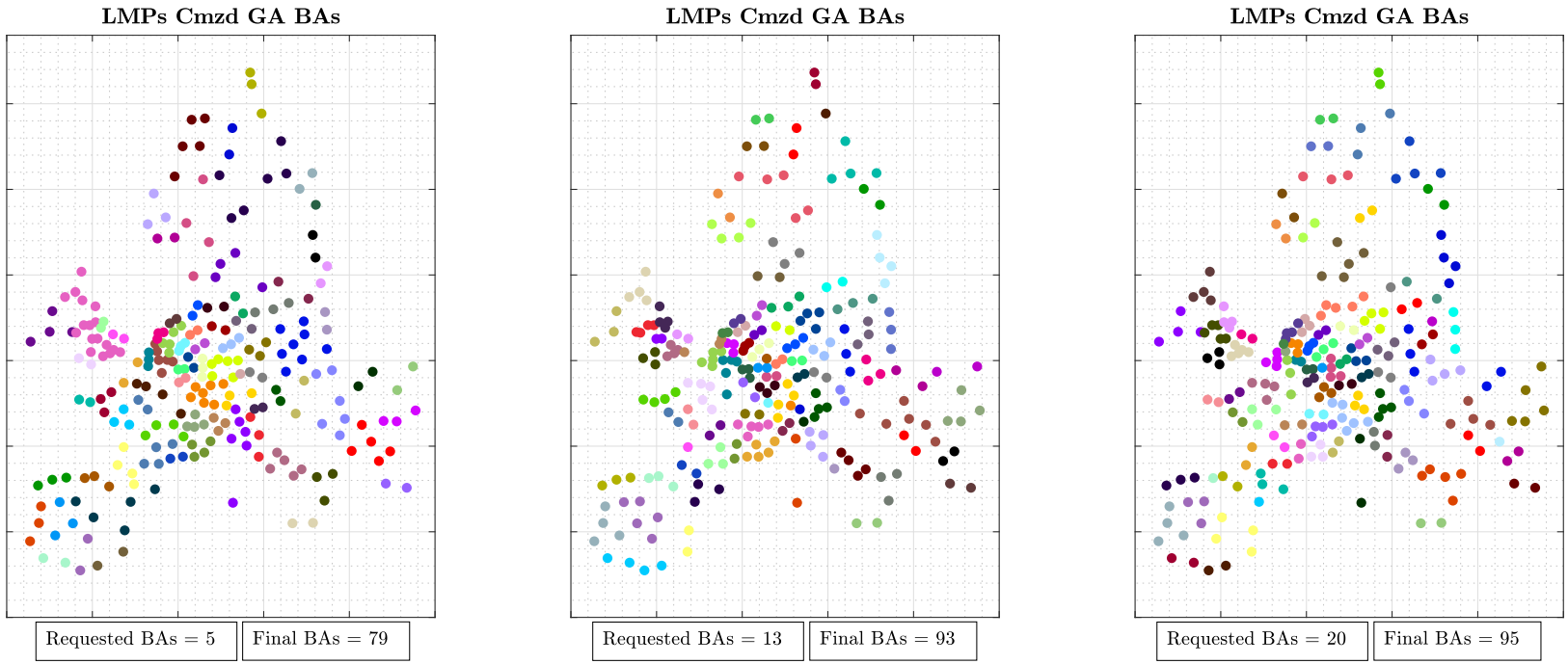

Figure 4.5: European transmission network's zonal configurations produced by the methodology's *GA*. With 5,13 and 20 *BAs* requested by the user and post-processing usage of the "CheckBAsConnection" handwritten function.

For these reasons, from here on out the centroids and medoids of Matlab K-means and K-medoids will be always initialized using the *K-means++* algorithm, naturally embedded inside Matlab and used by default by these commands if no random initialization is specifically required by the user. The centroids and medoids of customized K-means and K-medoids will be always initialized using an external algorithm able to compute a certain user-defined number of most distant observations, according to the Euclidean distance evaluated between them on the basis of a specific user-given database of features. In this case, the features will be obviously made up of *LMP* hourly trends or nodal *PTDFs* of most congestible lines, depending on whether the algorithm used is LMPs-based or PTDFs-based.

4.3 Methodology application

This section contains the actual methodology application, together with the associated considerations. Therefore, first of all within the next *Section 4.3.1* the previously described zonal configurations' assessment criteria will be deeply investigated with varying number of *BAs*. This will allow to decide a reasonable input number of *BAs* for each of the methodology's algorithms, as formerly decided in *Section 3.2.1*, since this parameter has always to be user-defined, directly or indirectly. Subsequently, inside *Section 4.3.2* the just decided *BAs* numbers will be actually used to make the various zonal configurations. And eventually, within the last *Section 4.3.3* the best methodology's algorithms will be presented, together with the associated considerations.

4.3.1 An input *BAs* number for each clustering algorithm

This section shows trends of both the clustering validity indicators and the economic efficiency ones as a function of the number of *BAs*. These last indeed, as previously stated inside *Section 3.2.1*, are the best way to suggest the user a reasonably optimal *BAs* number for each of the methodology's clustering algorithms. Or a reasonably optimal average *LMPs* tolerance, which however defines indirectly the price zones number and is the user-defined input as regards the *PDC*. Therefore, being aware of the *BAs* number's extremes of 2 and 20 which have been previously defined inside *Section 3.2.2*, the following pages provide the aforementioned evaluation indicators' trends from 2 to 10 *BAs* and from 11 to 20 *BAs*, so as to facilitate the reading of the graphs. Inside these last, it is worth remembering three things.

Firstly, each of the clustering validity indicators has been normalized respect to its maximum value among the ones of the same algorithms family evaluated along the various number of *BAs* here considered. In other words, for instance the *MIA* values of all the customized clustering algorithms' zonal configurations will be normalized respect to the maximum *MIA* value emerged among them. Whereas, the *MIA* values of all the Matlab clustering algorithms' zonal configurations will be normalized respect to their separate maximum.

Secondly, the economic efficiency indicators have not been normalized, so as not to lose the physical sense of their unitary rating scale. But however, their unitary range has the same meaning of the above normalized clustering validity indicators' one, i.e. a progressively improving condition going from 1 to 0.

Thirdly, all the following plots are endowed with semitransparent vertical lines, which born along the horizontal axis, in correspondence of the various "Requested *BAs*" values, and hold numbers on the lower right-sides. These last, reveal quite important information. In fact, each of them represents the final number of price zones that populates the zonal configuration created by the respective clustering algorithm, which entitles the origin graph, when this latter is run with that specific number of requested *BAs*. In other words, taking for example the Clustering Validity Indicators' (*CVIs*) trends from 2 to 10 requested *BAs* referring to the Matlab LMPs-based K-means' zonal configurations, contained inside Fig. 4.6, when looking at 6 requested *BAs* the semitransparent vertical line reports the number 8. It means that the partitioning technique in question has been actually run with 6 requested *BAs* but, due to the additional handwritten functions "NoSingleNodeBAs" and "CheckBAsConnection" downstream executed in order to make the algorithm comply with the 8^{th} and the 9^{th} zonal configurations' optimality requirement, the final zonal configuration has revealed 8 *BAs*. Moreover, the fact that this latter "Final *BAs*" number is greater than the initial user-defined one, contained inside the "Requested *BAs*" horizontal axis, shows the typical prevalence of the handwritten function "CheckBAsConnection" over the "NoSingleNodeBAs" one, which however is not compulsory as afterwards demonstrated. As a final remark, the same semitransparent vertical lines are also present inside the *CVIs* and Economic Efficiency Indicators' (*EEIs*) trends referred to the customized *PDC*'s zonal configurations. Nevertheless, in that case their numbers only represent the final price zones numbers of the various *BAs* sets produced, just influenced by the handwritten function "NoSingleNodeBAs", since the other one is never applied on this clustering algorithm as previously remarked inside *Section 4.2.1*. Moreover, there is no possibility of comparison respect to the horizontal axis' values, which contains instead the average *LMPs* ranges of tolerance.

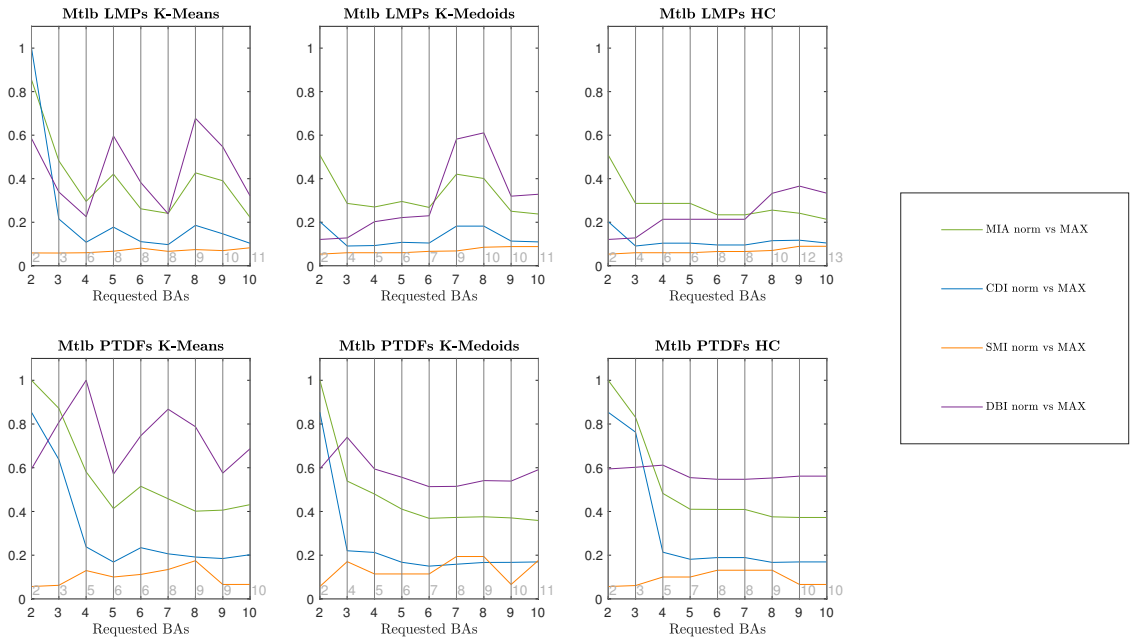

Figure 4.6: Clustering validity indicators' trends from 2 to 10 *BAs* for zonal configurations coming from methodology's Matlab algorithms.

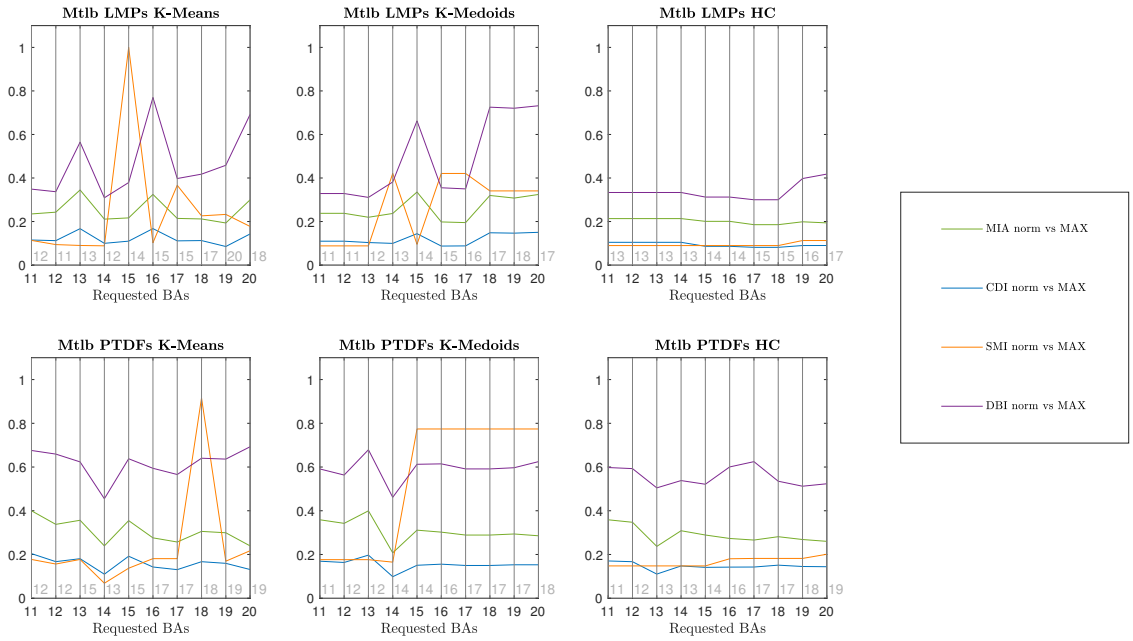

Figure 4.7: Clustering validity indicators' trends from 11 to 20 *BAs* for zonal configurations coming from methodology's Matlab algorithms.

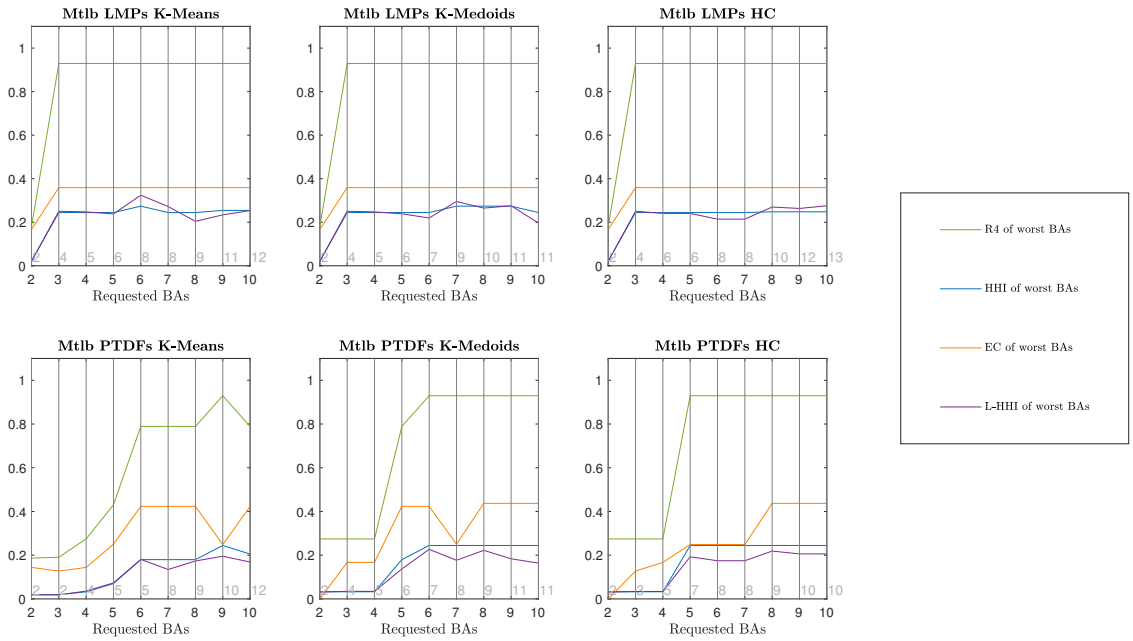

Figure 4.8: Economic efficiency indicators' trends from 2 to 10 *BAs* for zonal configurations coming from methodology's Matlab algorithms.

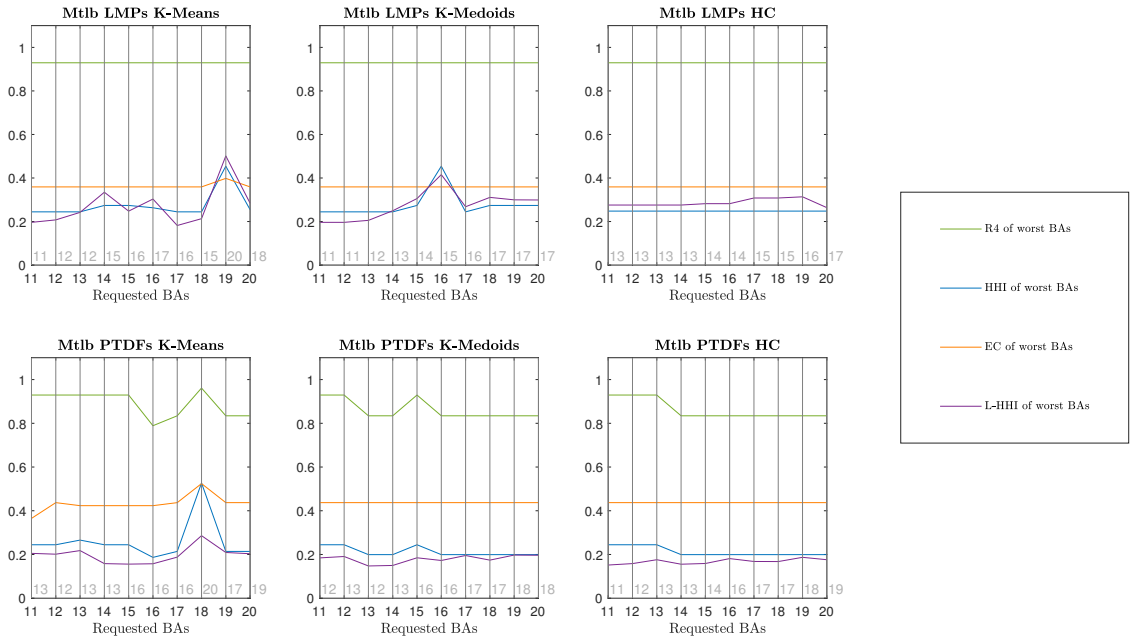

Figure 4.9: Economic efficiency indicators' trends from 11 to 20 *BAs* for zonal configurations coming from methodology's Matlab algorithms.

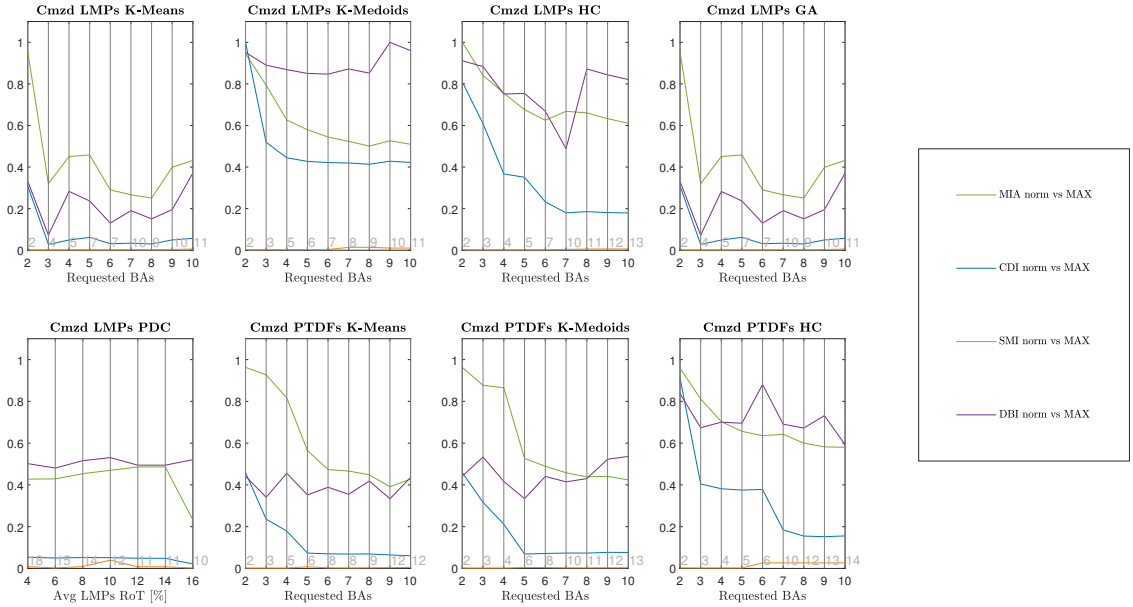

Figure 4.10: Clustering validity indicators' trends from 2 to 10 *BAs* and from 4% to 16% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms.

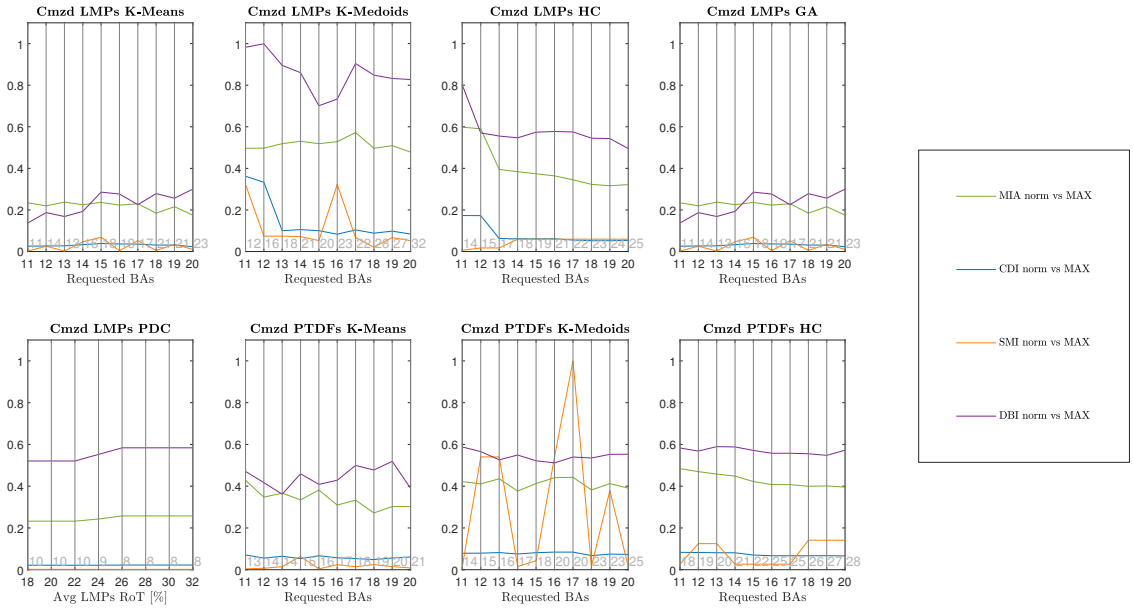

Figure 4.11: Clustering validity indicators' trends from 11 to 20 *BAs* and from 18% to 32% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms.

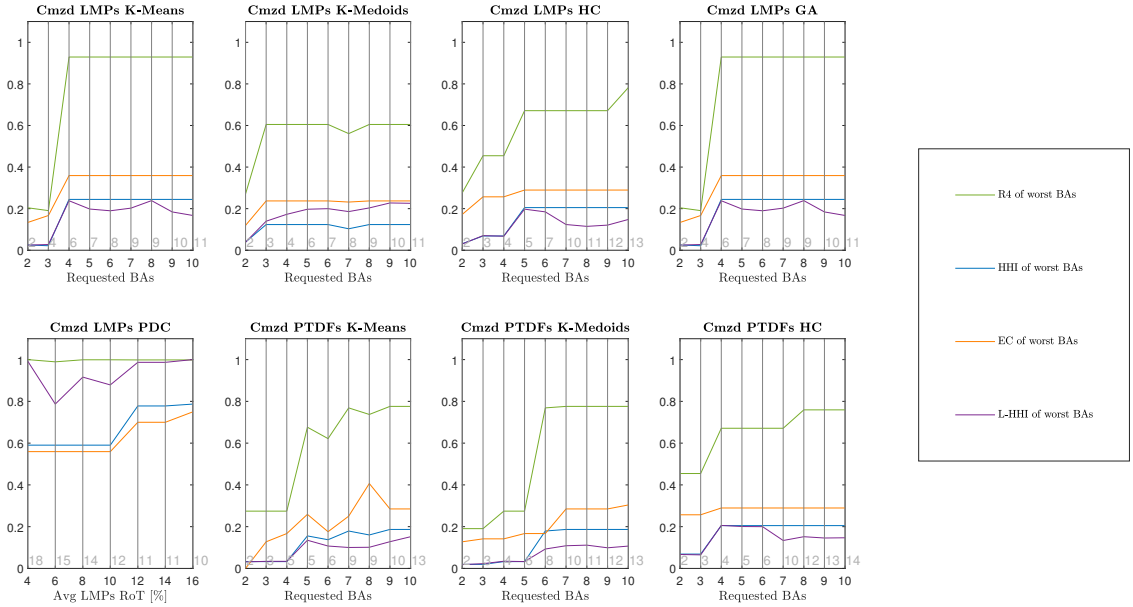

Figure 4.12: Economic efficiency indicators' trends from 2 to 10 *BAs* and from 4% to 16% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms.

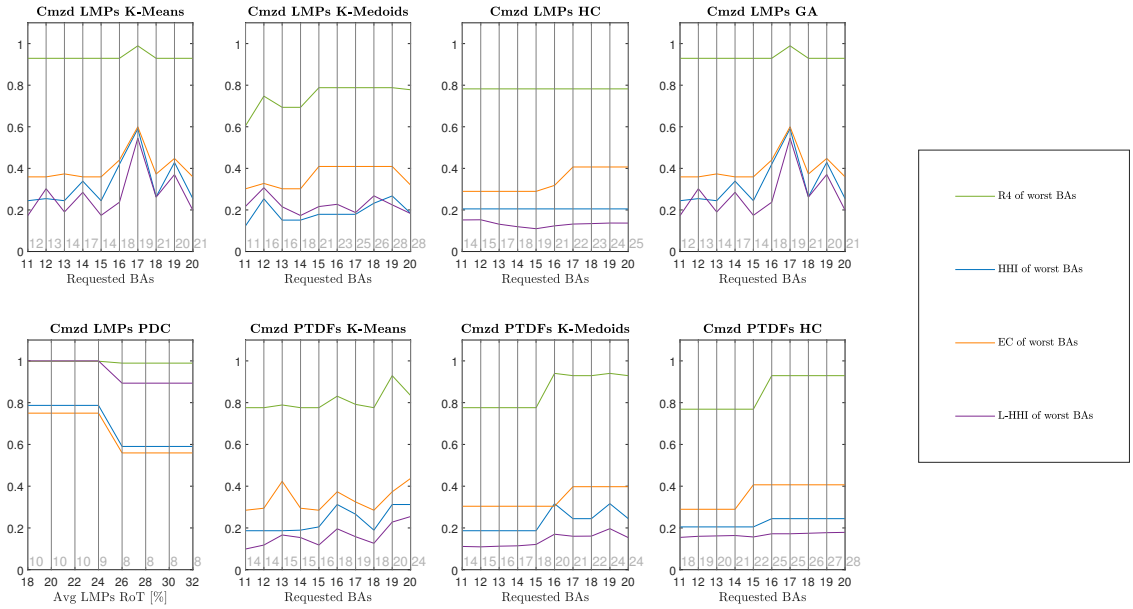

Figure 4.13: Economic efficiency indicators' trends from 11 to 20 *BAs* and from 18% to 32% average *LMPs* tolerance for zonal configurations coming from methodology's customized algorithms.

For the aforementioned reasons, since all the zonal configurations' assessment indicators reveal better results the more they are close to zero, the just reported trends have to be investigated towards their minimums, in order to find reasonably optimal *BAs* number for each of the methodology's clustering algorithms, or a reasonably optimal average *LMPs* tolerance for the *PDC*. Therefore, Table 4.1 firstly contains the overall minimum points for all the Matlab suitable clustering algorithms, according to both their clustering validity indicators trends and economic efficiency ones, and then it shows the same points for all the customized suitable clustering algorithms.

Table 4.1: Overall minimum points of the zonal configurations' assessment indicators trends, as a function of the *BAs* number.

Clustering algorithm	**CVIs' Minimum**	**EEIs' Minimum**
Mtlb LMPs K-means	7 Requested *BAs*	2 Requested *BAs*
Mtlb LMPs K-medoids	3 Requested *BAs*	2 Requested *BAs*
Mtlb LMPs HC	3 Requested *BAs*	2 Requested *BAs*
Mtlb PTDFs K-means	14 Requested *BAs*	3 Requested *BAs*
Mtlb PTDFs K-medoids	14 Requested *BAs*	2 Requested *BAs*
Mtlb PTDFs HC	13 Requested *BAs*	2 Requested *BAs*
Cmzd LMPs K-means	11 Requested *BAs*	2 Requested *BAs*
Cmzd LMPs K-medoids	15 Requested *BAs*	2 Requested *BAs*
Cmzd LMPs HC	20 Requested *BAs*	2 Requested *BAs*
Cmzd LMPs GA	11 Requested *BAs*	2 Requested *BAs*
Cmzd LMPs PDC	18% Avg *LMPs* RoT	6% Avg *LMPs* RoT
Cmzd PTDFs K-means	20 Requested *BAs*	2 Requested *BAs*
Cmzd PTDFs K-medoids	5 Requested *BAs*	2 Requested *BAs*
Cmzd PTDFs HC	16 Requested *BAs*	3 Requested *BAs*

By looking at the overlying table, the first thing that stands out is the huge concentration of *EEIs*' minimum points around 2 and 3 requested *BAs*. This was predictable, since all the economic efficiency indicators are respectively close to the maximum 1 or close to the minimum 0 whether the evaluated zonal configuration is similar to a monopoly or a perfect competition. And hence, if the whole reduced model of the European transmission network is divided into only 2 or 3 zones, each of these last unavoidably results to be composed of a high number of producers. This holds likely a limited market share and consequently a nearly null market power, so ending up giving to their respective zonal-based market an almost complete perfect competition. However, since having just so few *BAs* would realistically be unacceptable for a such vast territory as the European continent, the 2 or 3 requested *BAs* in question have always to be refused, necessarily in favor of at least 5 requested *BAs*. In other words, inside *Section 3.2.2* it has been chosen 2 as lower boundary of the user-defined *BAs* number's extremes, intended to make the methodology's algorithms comply with the 6^{th} zonal configurations' optimality requirement. But

actually, this number can only be the theoretical lower limit of the aforementioned range. In fact, a such vast territory as the European continent must ineluctably be divided into a minimum of 5 price zones, at least for obvious bureaucratic reasons linked to their management. For the same reason, the couple of *CVIs*' minimums in correspondence of 3 requested *BAs*, deriving from Matlab LMPs-based K-medoids and *HC*, has to be equally refused in favor of a greater number, which will be afterwards investigated.

At this point, that both the *CVIs*' minimums and the *EEIs*' ones have been evaluated for all the methodology's suitable clustering algorithms, it is necessary to intersect the above findings, in order to find a sole reasonably optimal user-defined *BAs* number for each of the partitioning methods. Or a sole reasonably optimal average *LMPs* tolerance for the *PDC*. Therefore, the following bulleted list contains some comments for each of the clustering techniques, and eventually gives a trade-off number for each of them, which then are eventually summed up inside Table 4.2.

Mtlb LMPs K-means. The *CVIs*' minimum would indicate 7 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 3 and 18 requested *BAs* the *EEIs* remain almost constant, the final choice falls on 7 *BAs*, because instead the *CVIs* have small values only in correspondence of few cases, among which there is 7 requested *BAs*.

Mtlb LMPs K-medoids. The *CVIs*' minimum would indicate 3 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since both of these values must be refused for the aforementioned reason, the *EEIs* remain almost constant from 3 to 15 requested *BAs*, hence not giving a so relevant contribution for the choice, and the *CVIs* do not vary so much from 3 to 6 requested *BAs*, the final choice falls on this latter number. Namely 6 requested *BAs*, acceptable because greater than the above mentioned realistic lower limit of 5 price zones.

Mtlb LMPs HC. The *CVIs*' minimum would indicate 3 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore again, since both of these values must be refused for the aforementioned reason, the *EEIs* remain almost constant from 3 to 20 requested *BAs*, hence not giving a so relevant contribution for the choice, and the *CVIs* do not vary so much from 3 to 7 requested *BAs*, the final choice falls on 6 requested *BAs*, to both share the number of the previous clustering algorithm and leave as little as possible the initial minimum points of *EEIs* and *CVIs*.

Mtlb PTDFs K-means. The *CVIs*' minimum would indicate 14 requested *BAs*, while the *EEIs*' one would say 3 requested *BAs*. Therefore, since the 3 requested *BAs* must be refused for the aforementioned reason and between 12 and 15 requested *BAs* the *EEIs* remain almost constant, the final choice falls on 14 *BAs*, because instead the *CVIs* have particularly minimum values in correspondence of 14 requested *BAs*, while all the surrounding cases have higher values.

Mtlb PTDFs K-medoids. The *CVIs*' minimum would indicate 14 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 13 and 20 requested *BAs* the *EEIs* remain almost constant apart from the 15 *BAs* case, the final choice falls on 14 *BAs*, because instead the *CVIs* have particularly minimum values in correspondence of 14 requested *BAs*, while all the surrounding cases have higher values.

Mtlb PTDFs HC. The *CVIs*' minimum would indicate 13 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 8 and 13 requested *BAs* the *EEIs* remain almost constant, the final choice falls on 13 *BAs*, because instead the *CVIs* have minimum values only from 13 to 15 requested *BAs*, while all the surrounding cases have higher values.

Cmzd LMPs K-means. The *CVIs*' minimum would indicate 11 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 4 and 11 requested *BAs* the *EEIs* remain almost constant, the final choice falls on 11 *BAs*, because instead the *CVIs* have minimum values only from 11 to 13 requested *BAs*, while all the surrounding cases have higher values.

Cmzd LMPs K-medoids. The *CVIs*' minimum would indicate 15 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 13 and 14 requested *BAs* the *EEIs* have relatively small values, the final choice falls on 14 *BAs*, because from 15 to 14 requested *BAs* the *CVIs* reveal only a limited increase, which can be endured to find a trade-off value.

Cmzd LMPs HC. The *CVIs*' minimum would indicate 20 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 13 and 20 the *CVIs* remain almost constant, the final choice falls on 14 *BAs*, because instead the *EEIs* have relatively small values from 10 to 16 requested *BAs*, and then reveal higher values.

Cmzd LMPs GA. The *CVIs*' minimum would indicate 11 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and between 4 and 11 requested *BAs* the *EEIs* remain almost constant, the final choice falls on 11 *BAs*, because instead the *CVIs* have minimum values only from 11 to 13 requested *BAs*, while all the surrounding cases have higher values.

Cmzd LMPs PDC. The *CVIs*' minimum would indicate 18% Avg *LMPs* RoT, while the *EEIs*' one would suggest 6% Avg *LMPs* RoT. Therefore, since from 16% to 32% of Avg *LMPs* RoT the *CVIs* remain almost constant and between 26% and 32% of Avg *LMPs* RoT the *EEIs* have values comparable to their own ones from 4%

to 10%, the final choice falls on 26% of Avg *LMPs* RoT, because it is the trade-off value nearest to both the *CVIs*' minimum and the *EEIs*' one.

Cmzd PTDFs K-means. The *CVIs*' minimum would indicate 20 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason, from 20 to 15 requested *BAs* the *CVIs* show just a slight increase of *MIA* and in correspondence of 15 requested *BAs* the *EEIs* have relatively small values, the final choice falls on this latter number. Namely 15 requested *BAs*.

Cmzd PTDFs K-medoids. The *CVIs*' minimum would indicate 5 requested *BAs*, while the *EEIs*' one would suggest 2 requested *BAs*. Therefore, since the 2 requested *BAs* must be refused for the aforementioned reason and the *EEIs*' values do not vary so much from 2 to 5 requested *BAs*, the final choice falls on 5 requested *BAs* as required by *CVIs*' minimum.

Cmzd PTDFs HC. The *CVIs*' minimum would indicate 16 requested *BAs*, while the *EEIs*' one would suggest 3 requested *BAs*. Therefore, since the 3 requested *BAs* must be refused for the aforementioned reason and between 14 and 17 the *CVIs* remain almost constant, the final choice falls on 14 *BAs*, because instead the *EEIs* have intermediate values from 8 to 14 requested *BAs*, and then reveal higher values.

Table 4.2: Trade-off user-defined inputs of the methodology's clustering algorithms, in terms of *BAs* numbers or Avg *LMPs* RoT.

Clustering algorithm	**Trade-off user-defined input**
Mtlb LMPs K-means	7 Requested *BAs*
Mtlb LMPs K-medoids	6 Requested *BAs*
Mtlb LMPs HC	6 Requested *BAs*
Mtlb PTDFs K-means	14 Requested *BAs*
Mtlb PTDFs K-medoids	14 Requested *BAs*
Mtlb PTDFs HC	13 Requested *BAs*
Cmzd LMPs K-means	11 Requested *BAs*
Cmzd LMPs K-medoids	14 Requested *BAs*
Cmzd LMPs HC	14 Requested *BAs*
Cmzd LMPs GA	11 Requested *BAs*
Cmzd LMPs PDC	26% Avg *LMPs* RoT
Cmzd PTDFs K-means	15 Requested *BAs*
Cmzd PTDFs K-medoids	5 Requested *BAs*
Cmzd PTDFs HC	14 Requested *BAs*

Therefore, having stated a trade-off user-defined input for each of the methodology's algorithms, it is now necessary to list the necessary tests, which unavoidably require to be executed to locate the best partitioning techniques. Each of these last indeed, derives from one of the aforementioned trade-off user-defined inputs and hence has a physical foundation, since it represents the condition in which at least one of the adopted clustering algorithms has had particularly good performance in terms of zonal configurations' assessment indicators. Moreover, it permits an in-depth comparison among all the methodology's partitioning techniques, both coming from Matlab commands and customized codes. As a result, the following Table 4.3 contains the tests in question, which will be subsequently shown and commented inside *Section 4.3.2*.

Table 4.3: Trade-off user-defined inputs of the methodology's clustering algorithms, in terms of *BAs* numbers or Avg *LMPs* RoT.

Number of test	**Requested BAs**	**Optimal clustering algorithms in that condition**
1^{st}	5	Cmzd PTDFs K-medoids Cmzd LMPs PDC
2^{nd}	6	Mtlb LMPs K-medoids Mtlb LMPs HC Cmzd LMPs PDC
3^{rd}	7	Mtlb LMPs K-means Cmzd LMPs PDC
4^{th}	11	Cmzd LMPs K-means Cmzd LMPs GA Cmzd LMPs PDC
5^{th}	13	Mtlb PTDFs HC Cmzd LMPs PDC
6^{th}	14	Mtlb PTDFs K-means Mtlb PTDFs K-medoids Cmzd LMPs K-medoids Cmzd LMPs HC Cmzd PTDFs HC Cmzd LMPs PDC
7^{th}	15	Customized PTDFs K-means Cmzd LMPs PDC

The customized LMPs-based *PDC* has been inserted inside all the tests. Because its user-defined input is the average *LMPs* tolerance. Which is linked to the *BAs* number required by other algorithms, but numerically it is different from this latter. Therefore, it can always be used as term of comparison during the tests in question. And so it is done by time after time setting the Avg *LMPs* RoT which gives to the *PDC*'s zonal

configuration the "Final *BAs*" as near as possible to the test's requested one (assessable from the semitransparent vertical lines of Fig. 4.10, Fig. 4.11, Fig. 4.12 and Fig. 4.13). So as to evaluate the *PDC* in different situations like all the other methodology's algorithms. Not only where it has the best performance, namely with 26% of Avg *LMPs* RoT as previously reported inside Table 4.2.

4.3.2 Resulting zonal configurations and their assessment

This section contains the 7 tests previously decided inside *Section 4.3.1*. Therefore, for each of them it will be provided 4 figures respectively containing:

- The geographical representation of the Matlab algorithms' zonal configurations.
- The geographical representation of the customized algorithms' zonal configurations.
- The all zonal configurations' assessment through the *CVIs*.
- The all zonal configurations' assessment through the *EEIs*.

Test 1: 5 Requested BAs and 26% of Avg LMPs RoT

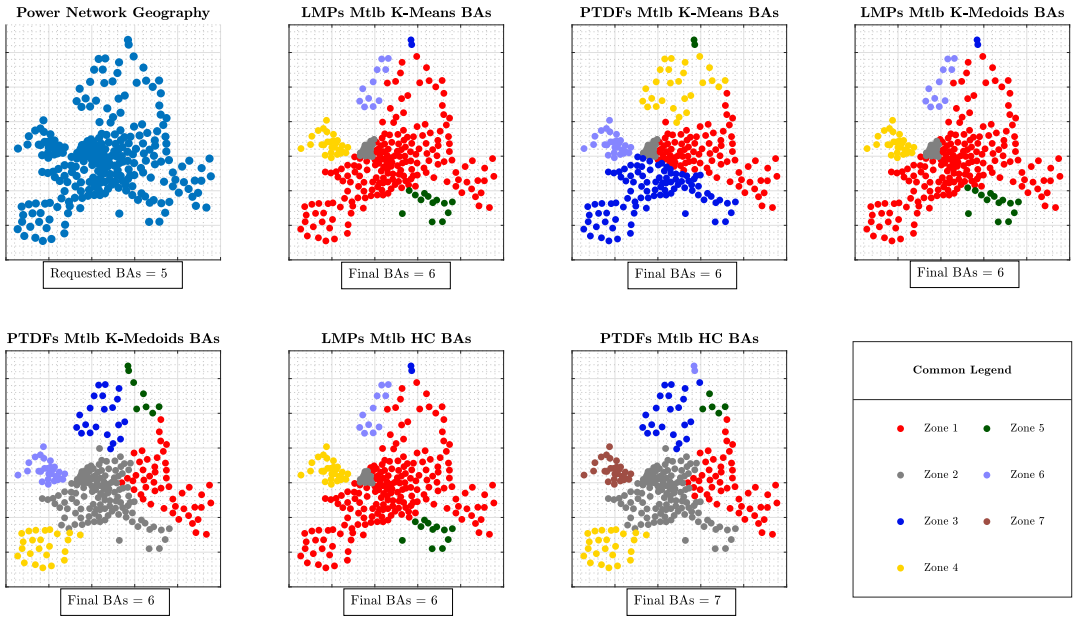

Figure 4.14: Geographical representation of the Matlab algorithm's zonal configurations which result from 5 requested *BAs*.

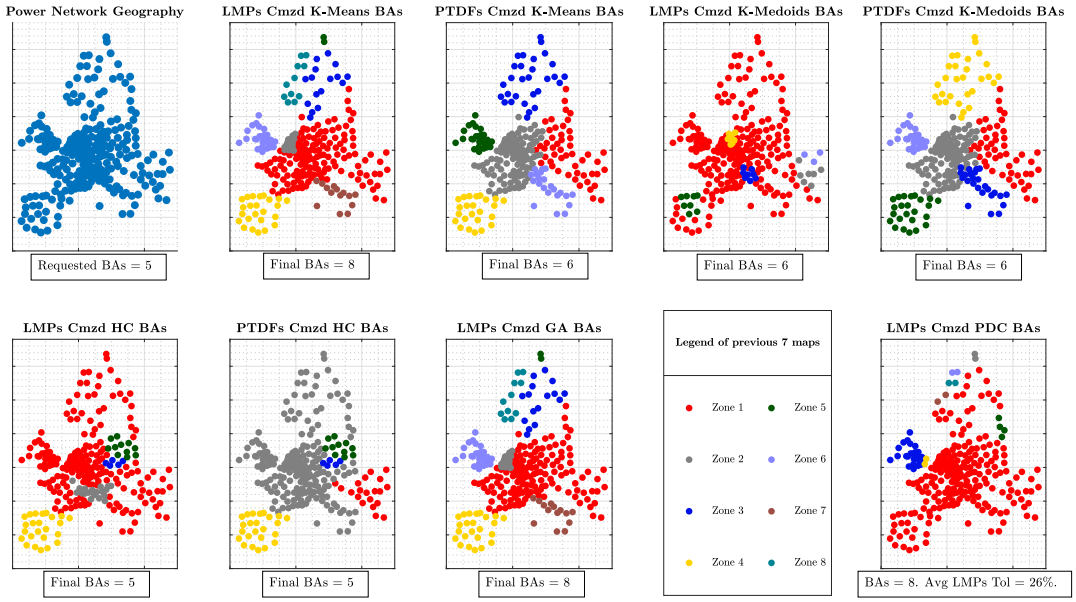

Figure 4.15: Geographical representation of the customized algorithm's zonal configurations which result from 5 requested *BAs* and 26% of average *LMPs* tolerance.

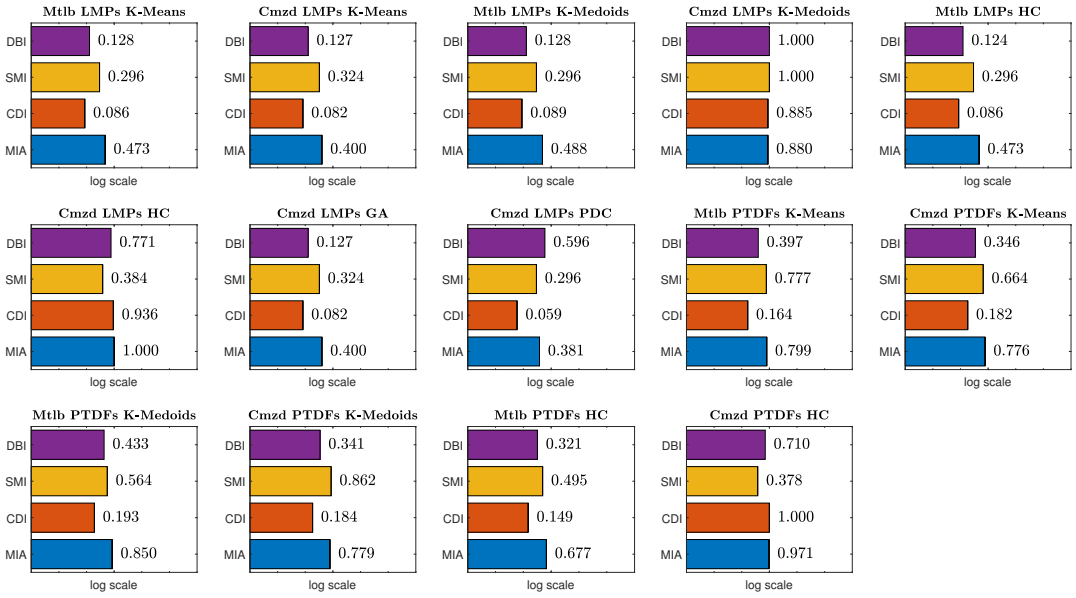

Figure 4.16: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance.

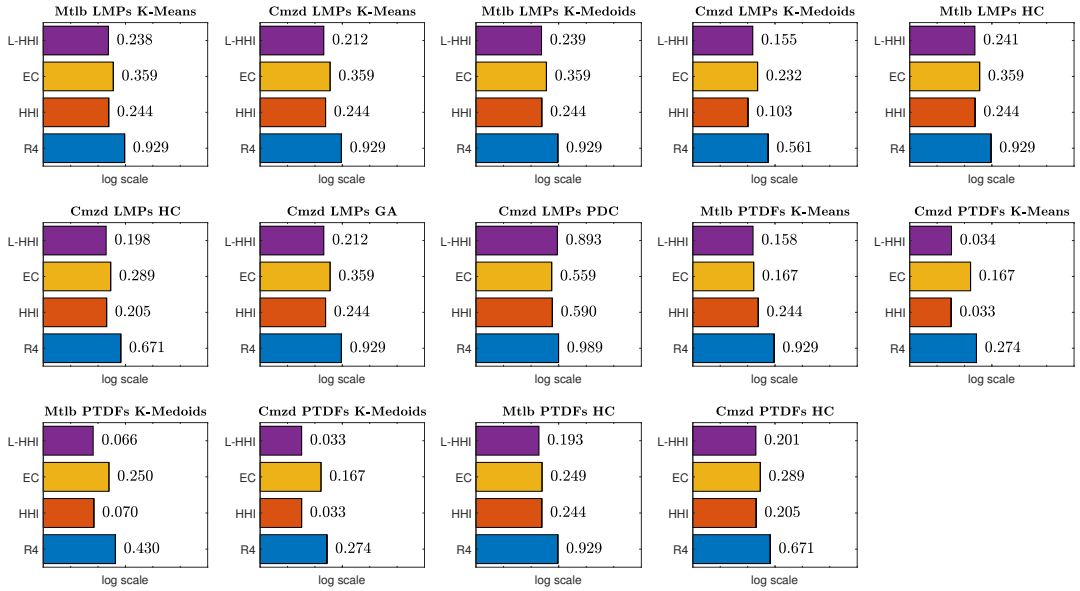

Figure 4.17: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.4 provides the information for each assessment indicator, with more than one clustering algorithm if there is a dead heat.

Table 4.4: Methodology's best clustering algorithms of test 1.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs HC
SMI	Mtlb LMPs K-means, Mtlb LMPs K-medoids, Mtlb LMPs HC, Cmzd LMPs PDC
CDI	Cmzd LMPs PDC
MIA	Cmzd LMPs PDC
L-HHI	Cmzd PTDFs K-medoids
EC	Mtlb PTDFs K-means, Cmzd PTDFs K-means, Cmzd PTDFs K-medoids
HHI	Cmzd PTDFs K-means, Cmzd PTDFs K-medoids
R_4	Cmzd PTDFs K-means, Cmzd PTDFs K-medoids

Test 2: 6 Requested BAs and 26% of Avg LMPs RoT

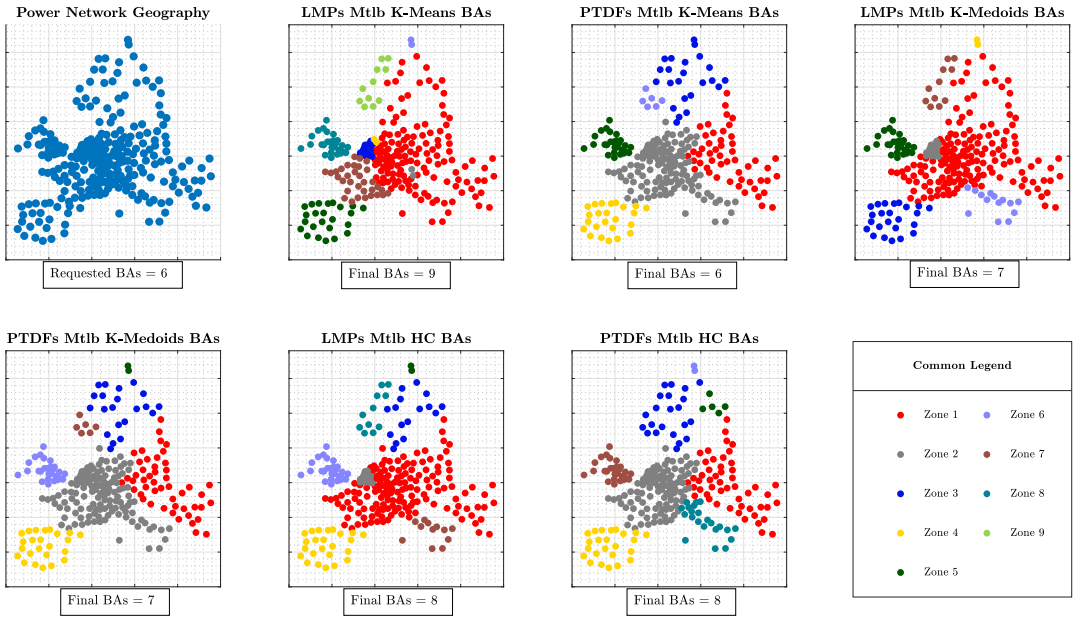

Figure 4.18: Geographical representation of the Matlab algorithm's zonal configurations which result from 6 requested *BAs*.

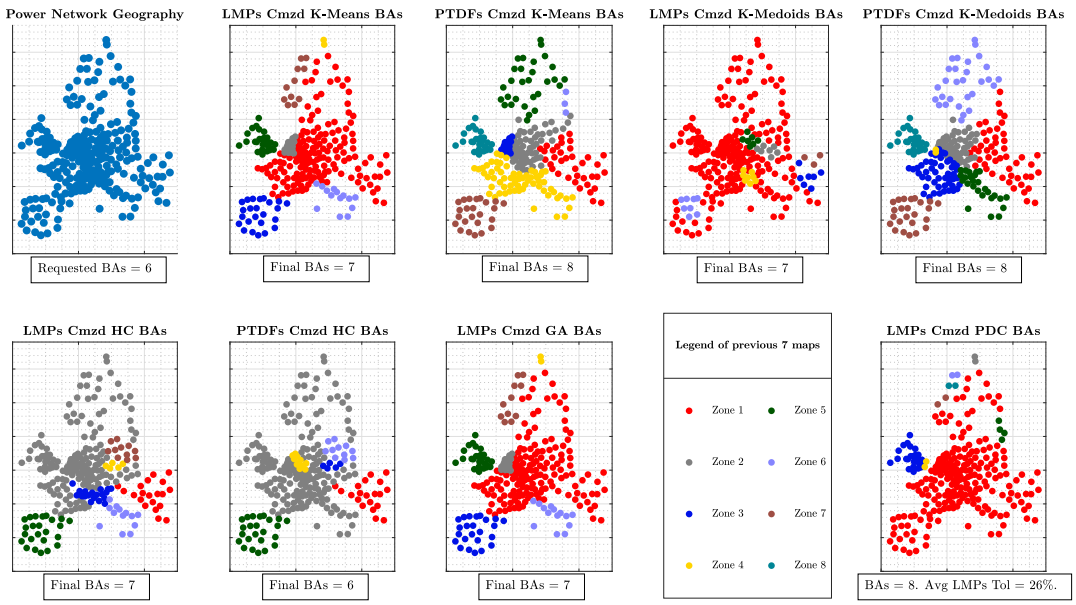

Figure 4.19: Geographical representation of the customized algorithm's zonal configurations which result from 6 requested *BAs* and 26% of average *LMPs* tolerance.

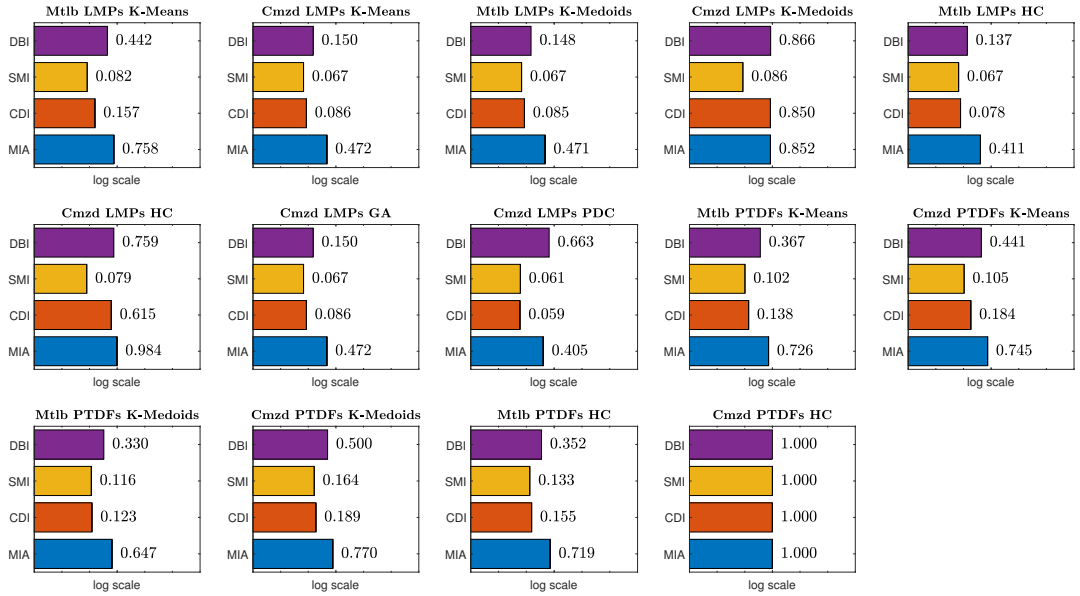

Figure 4.20: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 6 requested *BAs* and 26% of average *LMPs* tolerance.

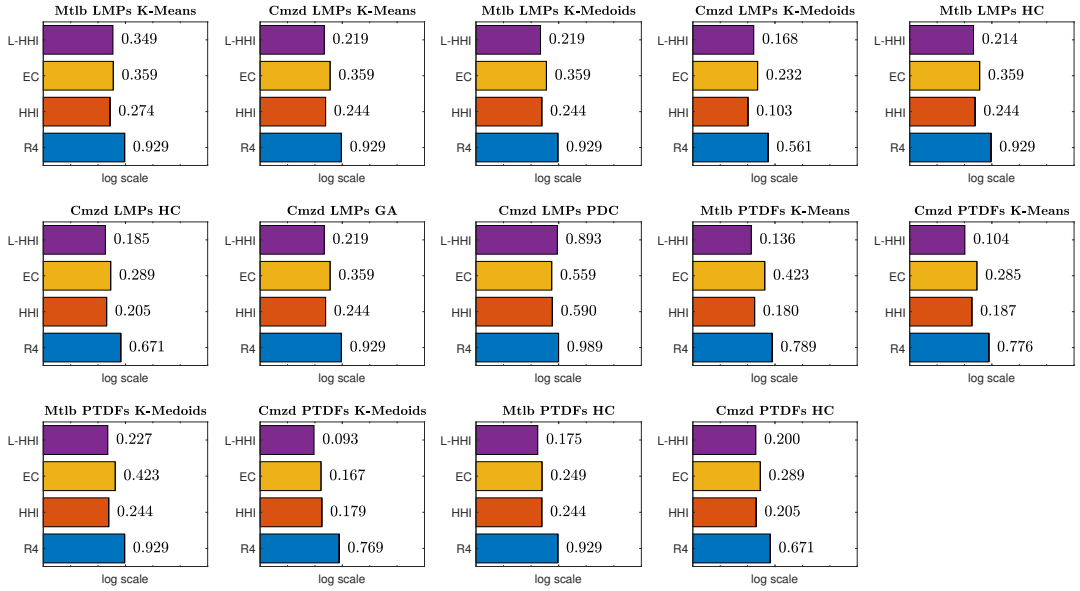

Figure 4.21: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 5 requested *BAs* and 26% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.5 provides the information for each assessment indicator.

Table 4.5: Methodology's best clustering algorithms of test 2.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs HC
SMI	Cmzd LMPs PDC
CDI	Cmzd LMPs PDC
MIA	Cmzd LMPs PDC
L-HHI	Cmzd PTDFs K-medoids
EC	Cmzd PTDFs K-medoids
HHI	Cmzd LMPs K-medoids
R_4	Cmzd LMPs K-medoids

Test 3: 7 Requested BAs and 26% of Avg LMPs RoT

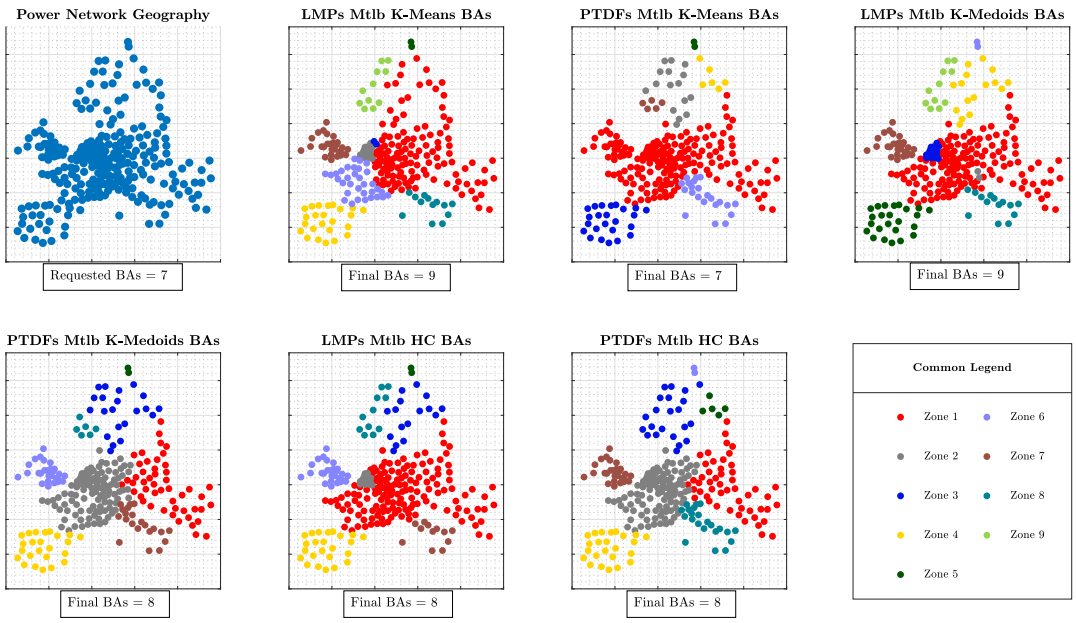

Figure 4.22: Geographical representation of the Matlab algorithm's zonal configurations which result from 7 requested *BAs*.

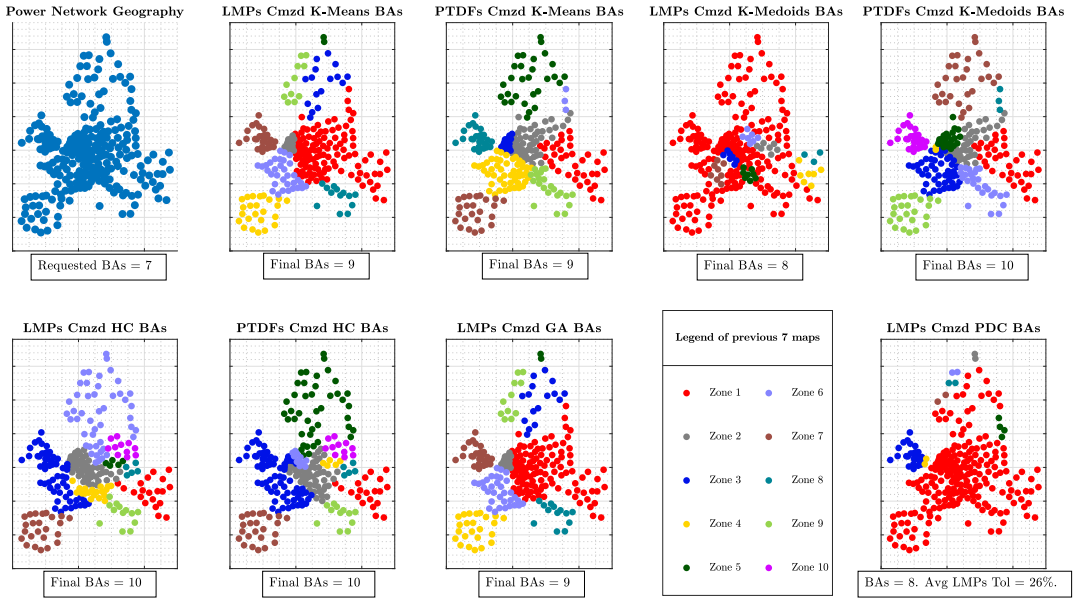

Figure 4.23: Geographical representation of the customized algorithm's zonal configurations which result from 7 requested *BAs* and 26% of average *LMPs* tolerance.

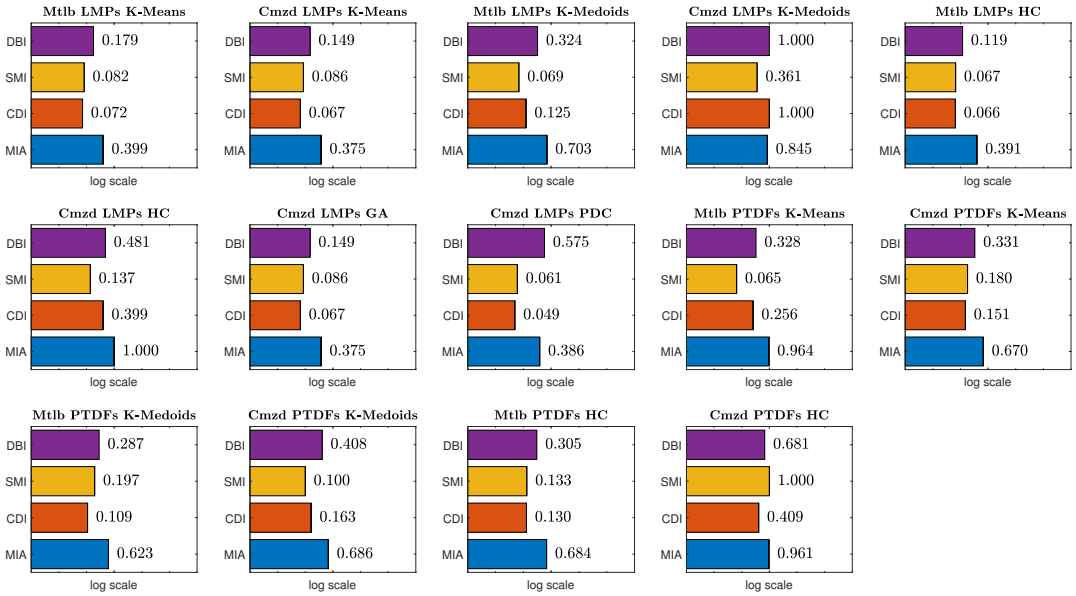

Figure 4.24: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 7 requested *BAs* and 26% of average *LMPs* tolerance.

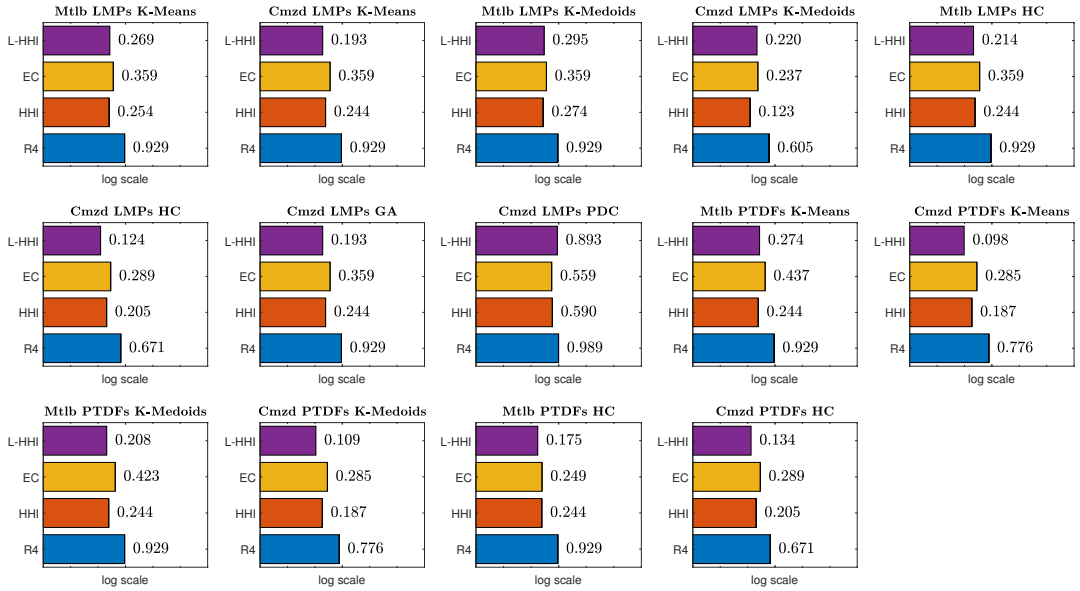

Figure 4.25: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 7 requested *BAs* and 26% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.6 provides the information for each assessment indicator, with more than one clustering algorithm if there is a dead heat.

Table 4.6: Methodology's best clustering algorithms of test 3.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs HC
SMI	Cmzd LMPs PDC
CDI	Cmzd LMPs PDC
MIA	Cmzd LMPs K-means, Cmzd LMPs GA
L-HHI	Cmzd PTDFs K-means
EC	Cmzd LMPs K-medoids
HHI	Cmzd LMPs K-medoids
R_4	Cmzd LMPs K-medoids

Test 4: 11 Requested BAs and 12% of Avg LMPs RoT

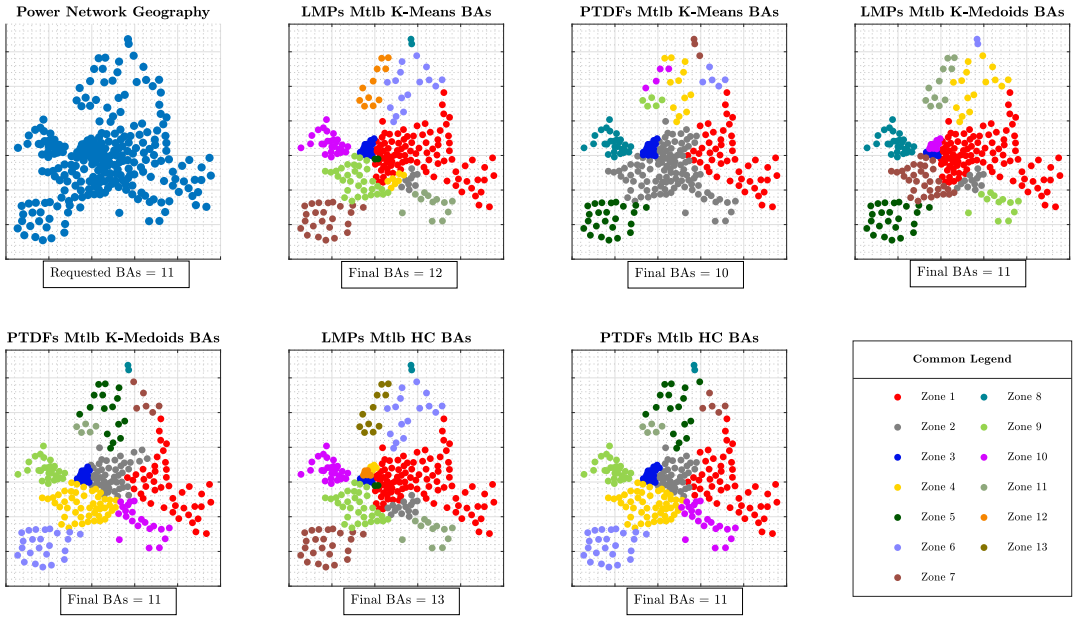

Figure 4.26: Geographical representation of the Matlab algorithm's zonal configurations which result from 11 requested *BAs*.

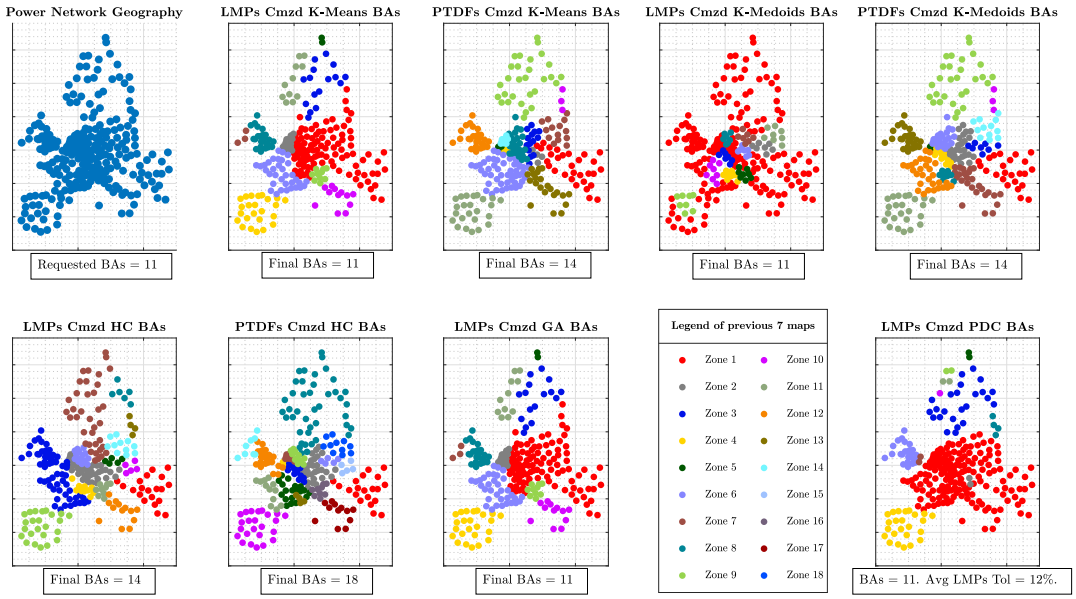

Figure 4.27: Geographical representation of the customized algorithm's zonal configurations which result from 11 requested *BAs* and 12% of average *LMPs* tolerance.

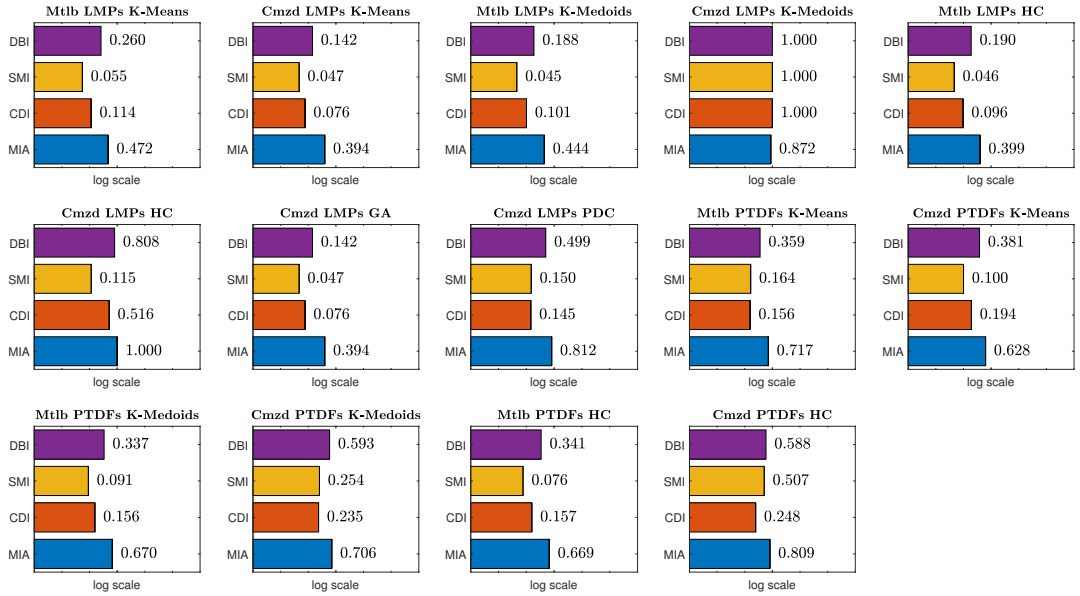

Figure 4.28: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 11 requested *BAs* and 12% of average *LMPs* tolerance.

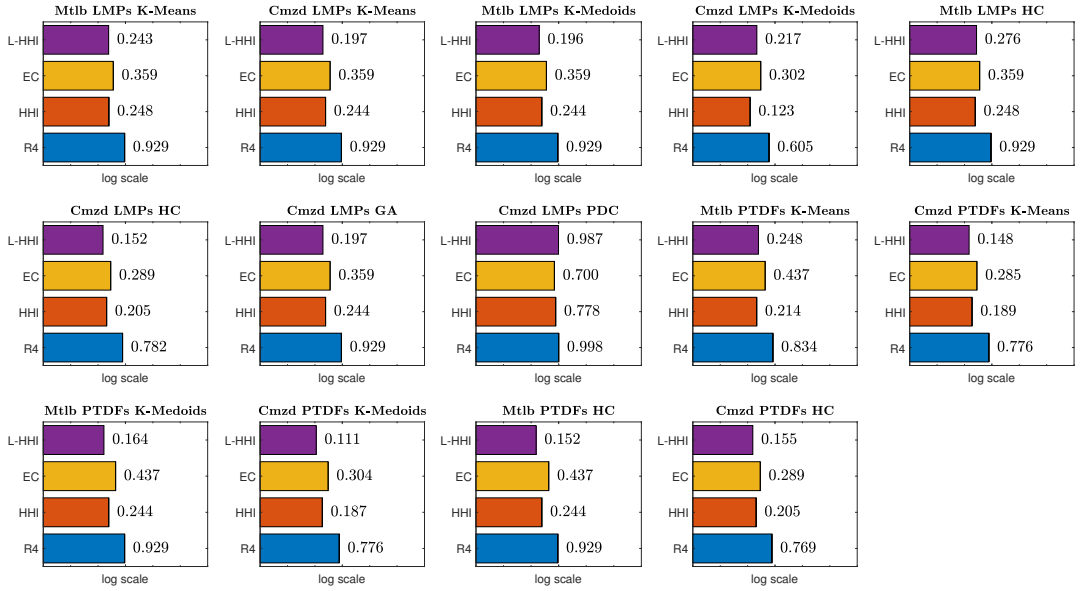

Figure 4.29: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 11 requested *BAs* and 12% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.7 provides the information for each assessment indicator, with more than one clustering algorithm if there is a dead heat.

Table 4.7: Methodology's best clustering algorithms of test 4.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Cmzd LMPs K-means, Cmzd LMPs GA
SMI	Mtlb LMPs K-medoids
CDI	Cmzd LMPs K-means, Cmzd LMPs GA
MIA	Cmzd LMPs K-means, Cmzd LMPs GA
L-HHI	Cmzd PTDFs K-medoids
EC	Cmzd PTDFs K-means
HHI	Cmzd LMPs K-medoids
R_4	Cmzd LMPs K-medoids

Test 5: 13 Requested BAs and 10% of Avg LMPs RoT

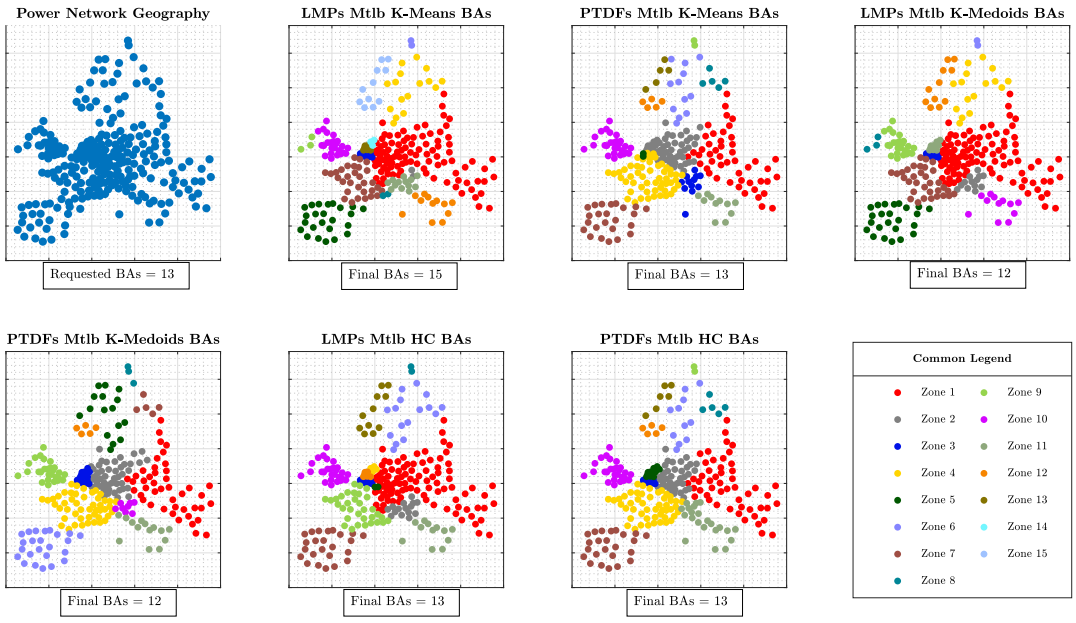

Figure 4.30: Geographical representation of the Matlab algorithm's zonal configurations which result from 13 requested *BAs*.

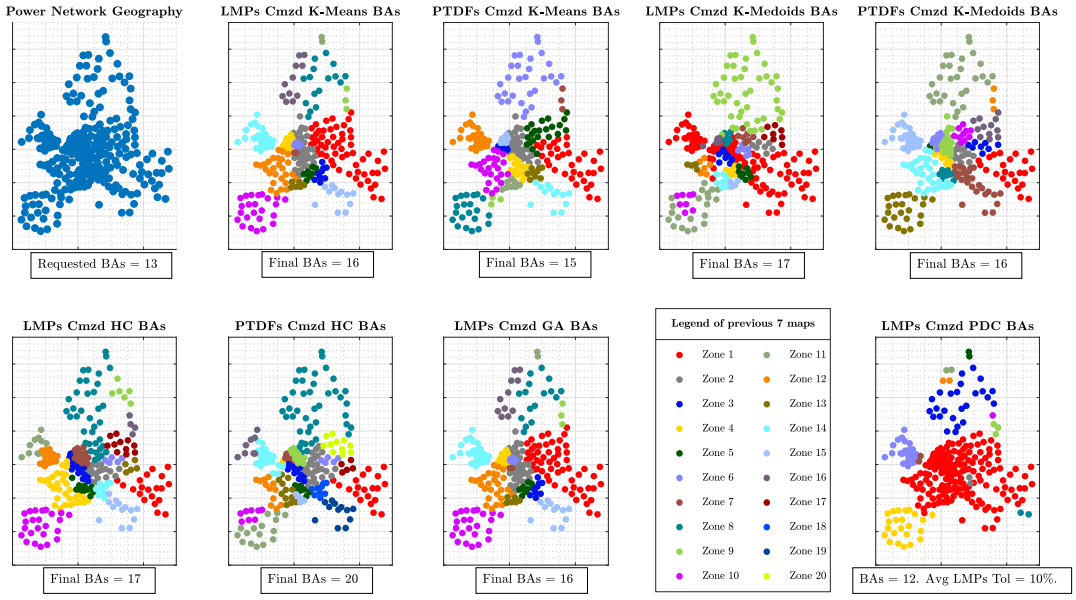

Figure 4.31: Geographical representation of the customized algorithm's zonal configurations which result from 13 requested *BAs* and 10% of average *LMPs* tolerance.

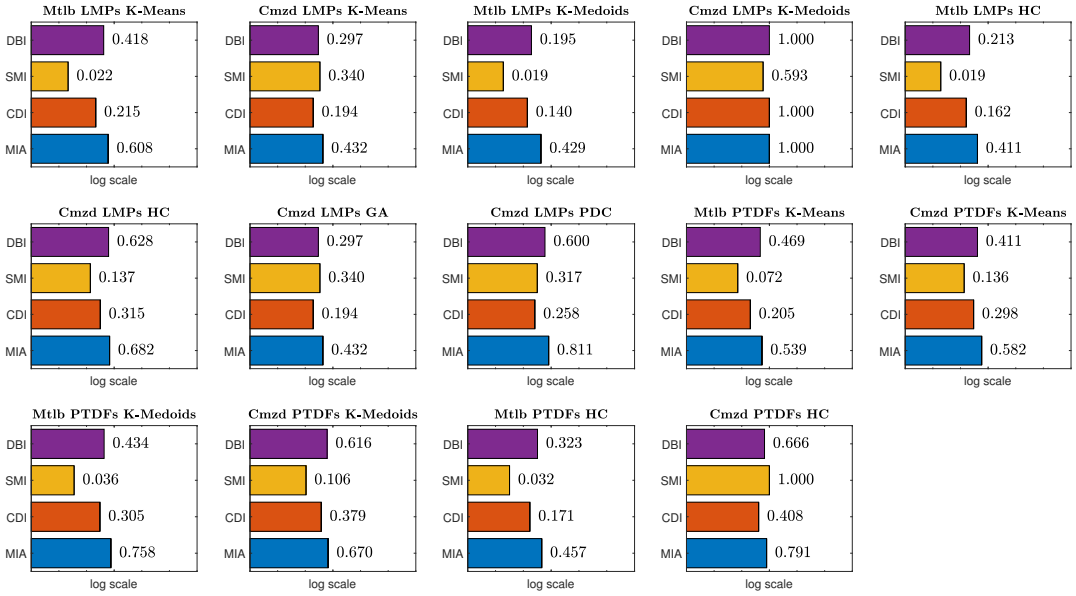

Figure 4.32: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 13 requested *BAs* and 10% of average *LMPs* tolerance.

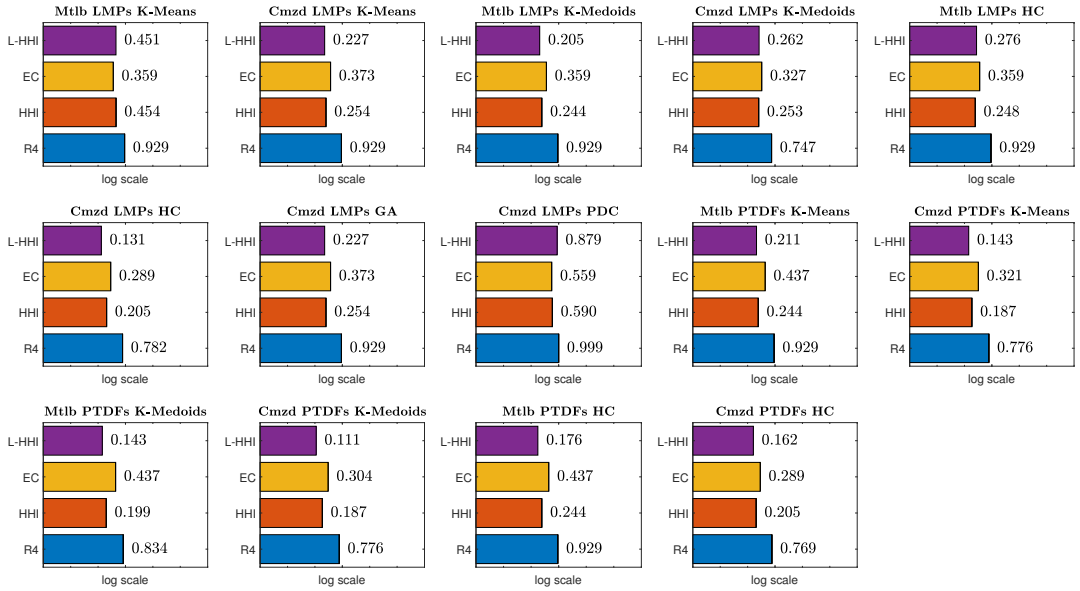

Figure 4.33: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 13 requested *BAs* and 10% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.8 provides the information for each assessment indicator, with more than one clustering algorithm if there is a dead heat.

Table 4.8: Methodology's best clustering algorithms of test 5.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs K-medoids
SMI	Mtlb LMPs K-medoids, Mtlb LMPs HC
CDI	Mtlb LMPs K-medoids
MIA	Mtlb LMPs HC
L-HHI	Cmzd PTDFs K-medoids
EC	Cmzd LMPs HC, Cmzd PTDFs HC
HHI	Cmzd PTDFs K-means, Cmzd PTDFs K-medoids
R_4	Cmzd LMPs K-medoids

Test 6: 14 Requested BAs and 8% of Avg LMPs RoT

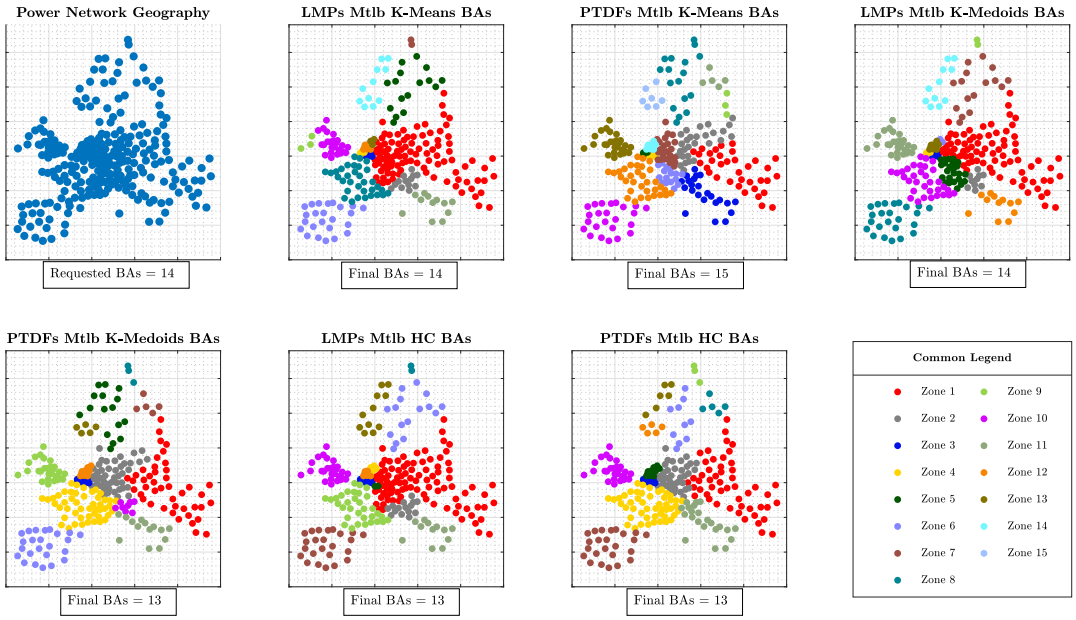

Figure 4.34: Geographical representation of the Matlab algorithm's zonal configurations which result from 14 requested *BAs*.

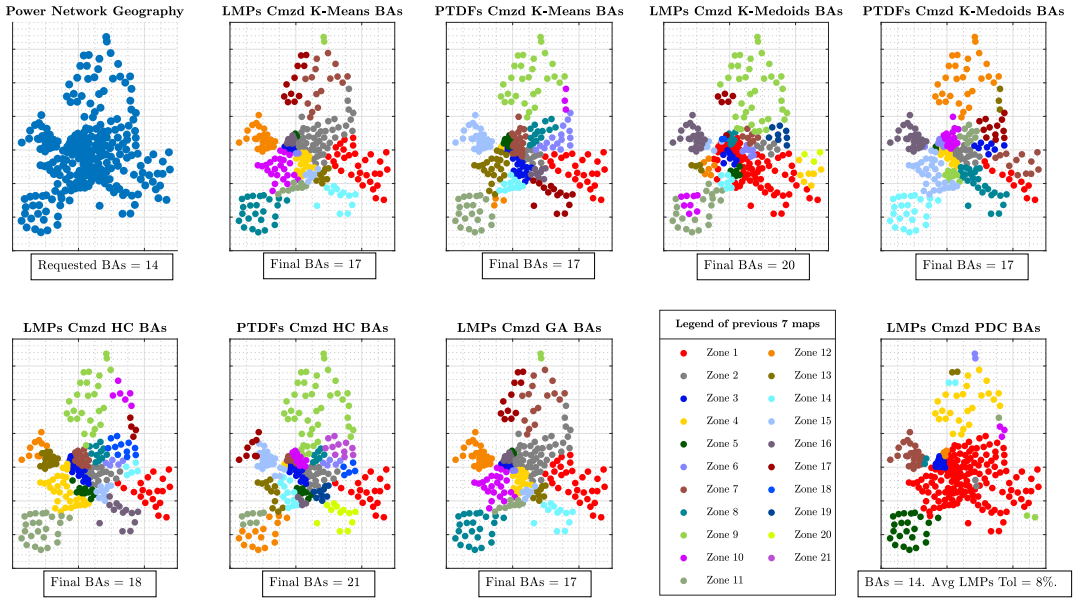

Figure 4.35: Geographical representation of the customized algorithm's zonal configurations which result from 14 requested *BAs* and 8% of average *LMPs* tolerance.

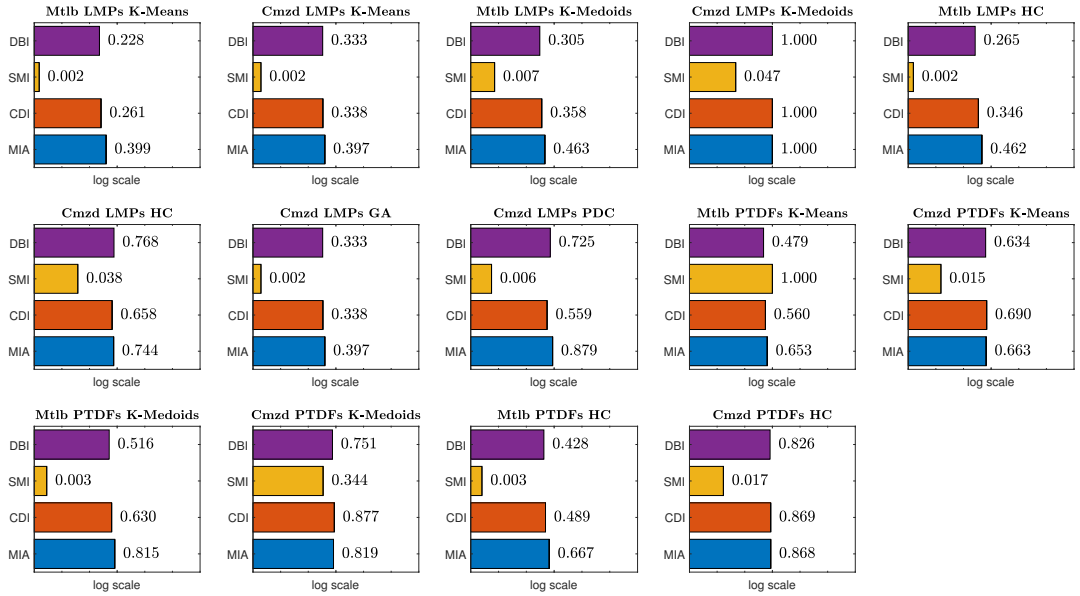

Figure 4.36: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 14 requested *BAs* and 8% of average *LMPs* tolerance.

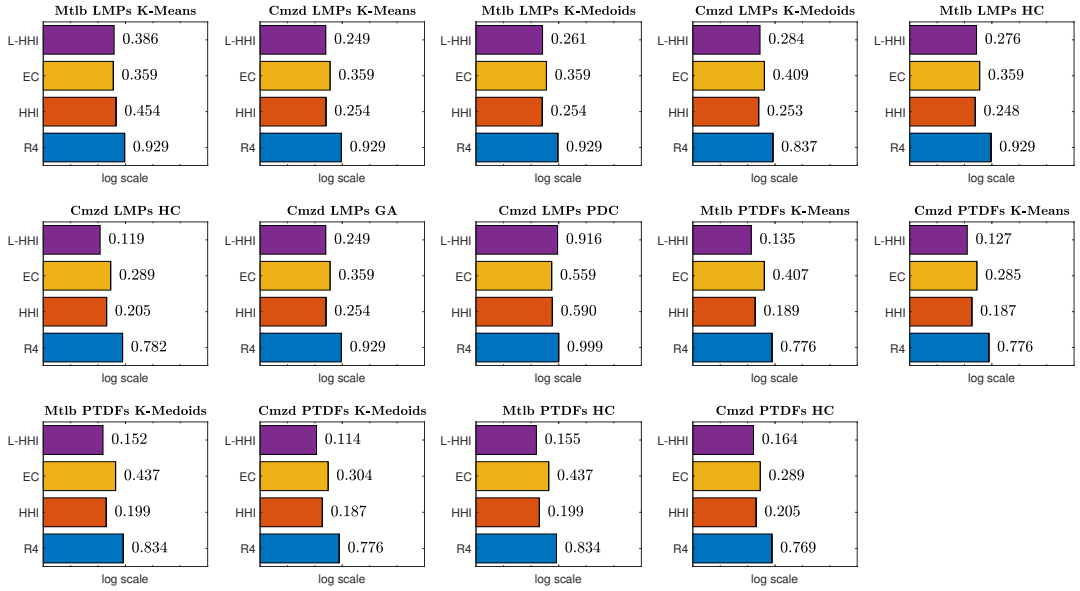

Figure 4.37: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 14 requested *BAs* and 8% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.9 provides the information for each assessment indicator.

Table 4.9: Methodology's best clustering algorithms of test 6.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs K-means
SMI	Mtlb LMPs K-means, Cmzd LMPs K-means, Mtlb LMPs HC, Cmzd LMPs GA
CDI	Mtlb LMPs K-means
MIA	Cmzd LMPs K-means, Cmzd LMPs GA
L-HHI	Cmzd PTDFs K-medoids
EC	Cmzd PTDFs K-means
HHI	Cmzd PTDFs K-means, Cmzd PTDFs K-medoids
R_4	Cmzd PTDFs HC

Test 7: 15 Requested BAs and 6% of Avg LMPs RoT

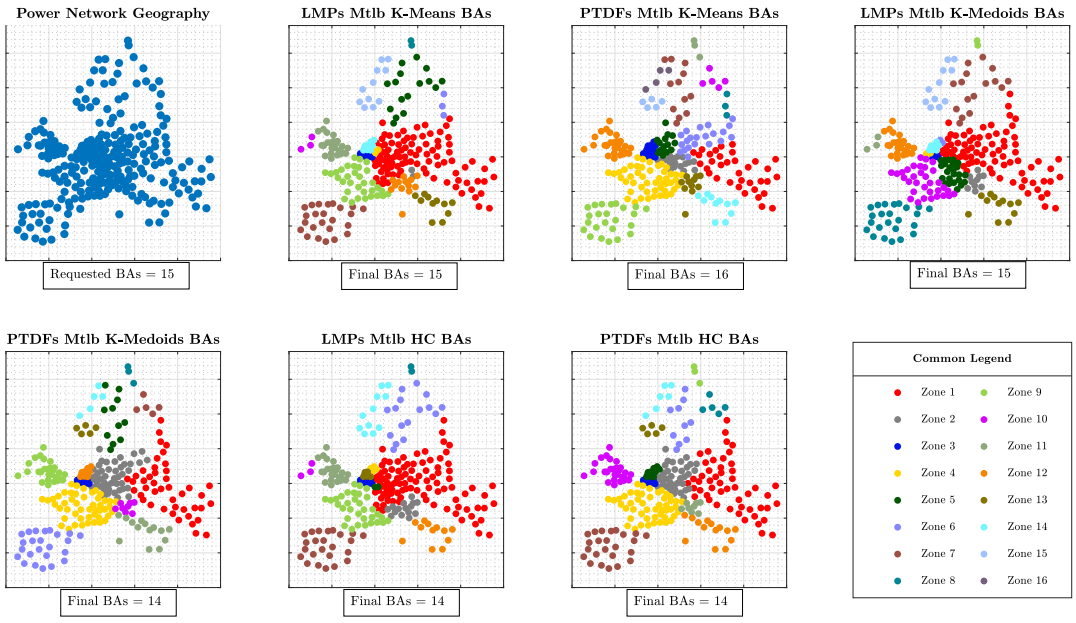

Figure 4.38: Geographical representation of the Matlab algorithm's zonal configurations which result from 15 requested *BAs*.

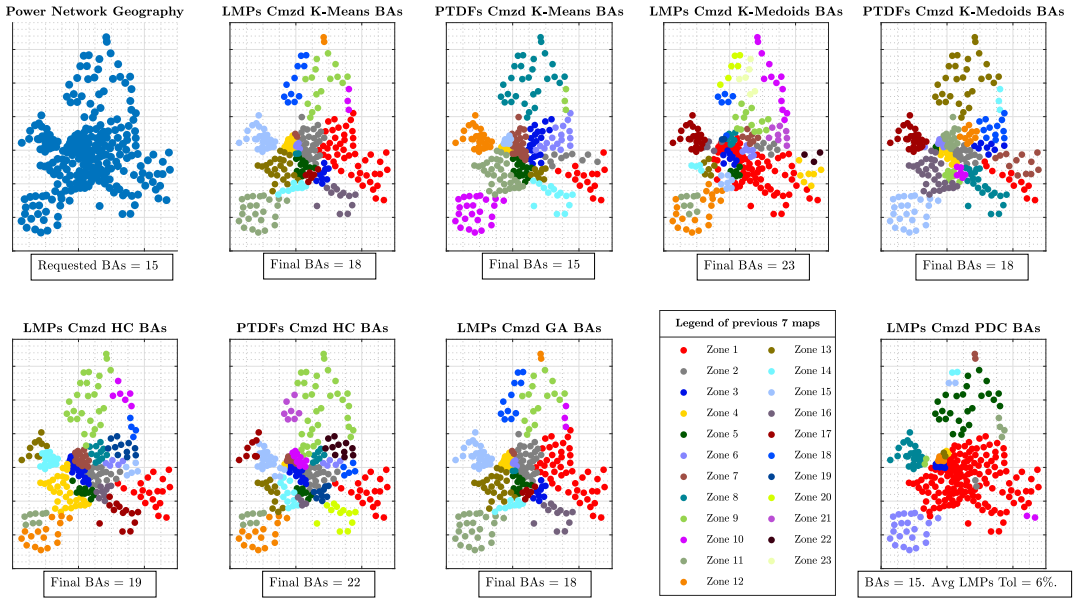

Figure 4.39: Geographical representation of the customized algorithm's zonal configurations which result from 15 requested *BAs* and 6% of average *LMPs* tolerance.

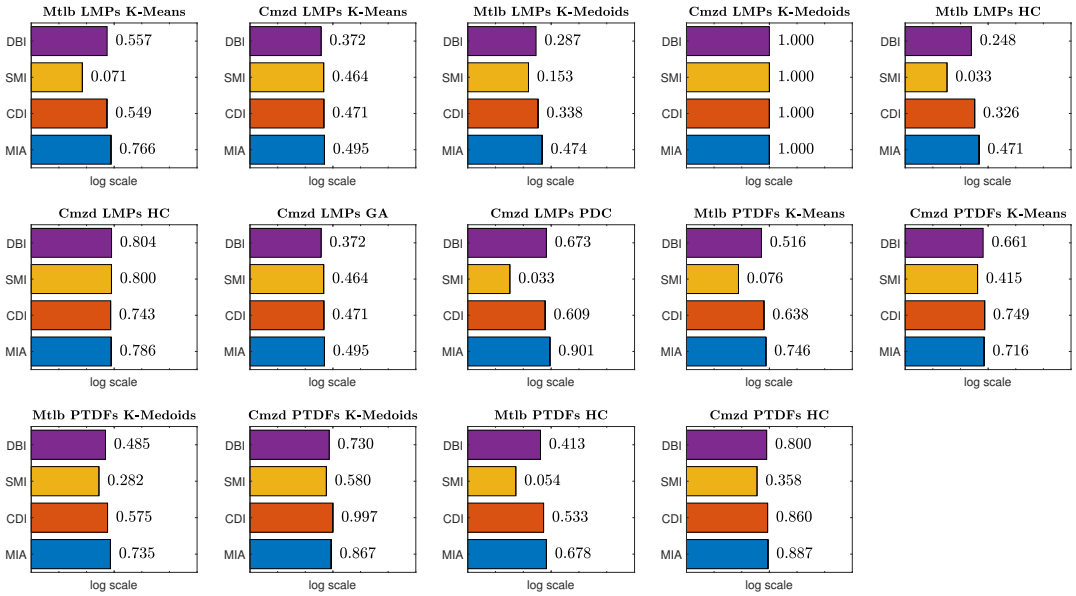

Figure 4.40: *CVIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 15 requested *BAs* and 6% of average *LMPs* tolerance.

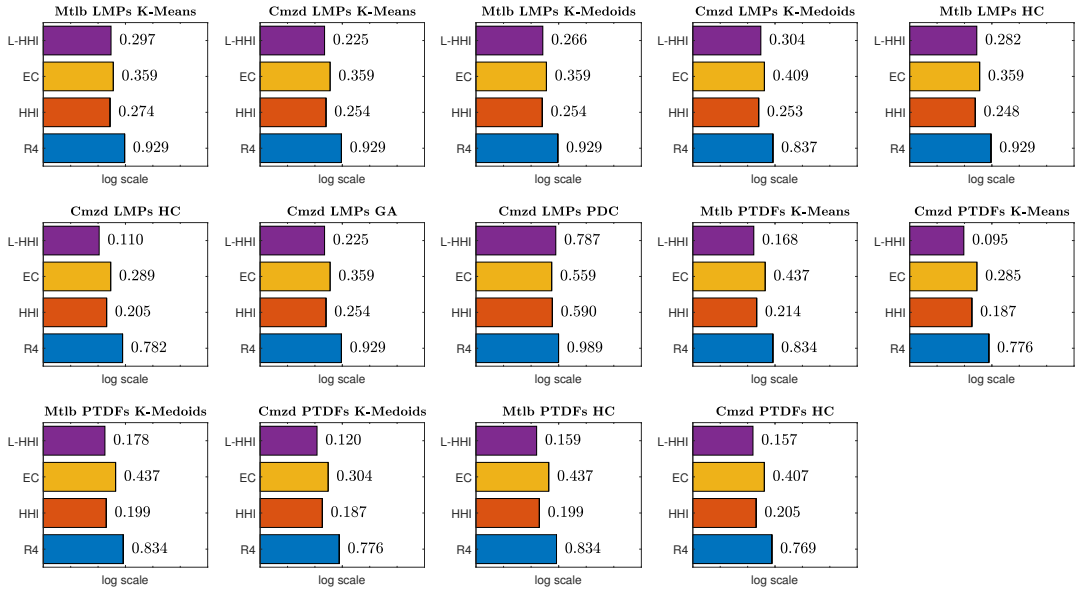

Figure 4.41: *EEIs* of the zonal configurations produced by both Matlab and customized clustering algorithms, with 15 requested *BAs* and 6% of average *LMPs* tolerance.

Therefore, remembering that for all the aforementioned zonal configurations' assessment indicators the more they are close to zero the more optimal is the judged *BAs* set, it is interesting to look for the minimum values of both the *CVIs* and the *EEIs*. They reveal the methodology's best algorithms of this test, according to the adopted evaluation criteria. Table 4.10 provides the information for each assessment indicator, with more than one clustering algorithm if there is a dead heat.

Table 4.10: Methodology's best clustering algorithms of test 7.

Assessment indicator	Best clustering algorithm according to the indicator
DBI	Mtlb LMPs HC
SMI	Mtlb LMPs HC, Cmzd LMPs PDC
CDI	Mtlb LMPs HC
MIA	Mtlb LMPs HC
L-HHI	Cmzd PTDFs K-means
EC	Cmzd PTDFs K-means
HHI	Cmzd PTDFs K-means, Cmzd PTDFs K-medoids
R_4	Cmzd PTDFs HC

4.3.3 Methodology's best suitable clustering algorithms

This section contains the best methodology's algorithms emerged by the previous zonal configurations assessment which has been executed inside *Section 4.3.2*, by using both clustering validity indicators and economic efficiency ones. These most performing partitioning techniques have been distinguished into different categories, according to the criterion used to declare their supremacy over the others. Therefore, the following sections contain and comment:

- The best methodology's clustering algorithm according to the *CVIs*.
- The best methodology's clustering algorithm according to the *EEIs*.
- The best methodology's clustering algorithm according to the *CVIs* and *EEIs* together.
- The best methodology's clustering algorithm according to the sum of all the *CVIs* and *EEIs* for each clustering algorithm over the 7 tests.

After these rankings, the final part of the section provides the Pareto front, which puts together, and hence permits to compare, all the zonal configurations previously emerged by the 14 methodology's algorithms during the 7 tests. In fact, these zonal configurations constitute 98 points, which are placed on a Cartesian plane with *CVIs* sum on the horizontal axis and *EEIs* sum on the vertical one. The points are shown in two versions, in order to ease the understanding to the reader:

- In the first version, the 98 zonal configurations' points are colored depending on the origin test.
- In the second version, the 98 zonal configurations' points are colored according to the clustering algorithm.

The clustering validity indicators' best

This category orders the methodology's clustering algorithms depending on how many times they have been the best within the *CVI* assessment. As a result, Table 4.11 provides the ranking in question.

Table 4.11: Ranking of the methodology's clustering algorithms over the 7 tests. According to *CVIs*.

Ranking	Number of times best	Clustering algorithm
1^{st}	11	Mtlb LMPs HC
2^{nd}	9	Cmzd LMPs PDC
3^{rd}	6	Cmzd LMPs K-means
3^{rd}	6	Cmzd LMPs GA
4^{th}	5	Mtlb LMPs K-medoids
5^{th}	4	Mtlb LMPs K-means
6^{th}	0	Cmzd LMPs K-medoids
6^{th}	0	Cmzd LMPs HC
6^{th}	0	Mtlb PTDFs K-means
6^{th}	0	Cmzd PTDFs K-means
6^{th}	0	Mtlb PTDFs K-medoids
6^{th}	0	Cmzd PTDFs K-medoids
6^{th}	0	Mtlb PTDFs HC
6^{th}	0	Cmzd PTDFs HC

Therefore, according to *CVIs*, the methodology's suitable clustering algorithms which have been deployed during the previous seven tests are roughly discernible in three groups:

1. The partitioning methods that outperform all the others, namely the Matlab LMPs-based *HC* and the customized LMPs *PDC*.

2. The methods with intermediate performance, from the 3^{th} to the 5^{th} position, i.e. the customized LMPs-based K-means, the customized LMPs-based *GA*, the Matlab LMPs-based K-medoids and the Matlab LMPs-based K-means.

3. All the other methods, namely, the algorithms tied for the 6^{th} place, whose zonal configurations reveal the weakest optimality by being 0 times the best ones.

Moreover, it is glaring to observe that the PTDFs-based clustering algorithms never manage to be the best according to *CVI* assessment. In fact, all of them are equal and are assigned the 6^{th} position with number of times best null, together with the Matlab and the customized PTDFs-based K-means.

The economic efficiency indicators' best

This category orders the methodology's clustering algorithms depending on how many times they have been the best within the *EEI* assessment. As a result, Table 4.12 provides the ranking in question.

Table 4.12: Ranking of the methodology's clustering algorithms over the 7 tests. According to *EEIs*.

Ranking	Number of times best	Clustering algorithm
1^{st}	12	Cmzd PTDFs K-medoids
2^{nd}	11	Cmzd PTDFs K-means
3^{rd}	8	Cmzd LMPs K-medoids
4^{th}	3	Cmzd PTDFs HC
5^{th}	1	Mtlb PTDFs K-means
5^{th}	1	Cmzd LMPs HC
6^{th}	0	Mtlb LMPs K-means
6^{th}	0	Cmzd LMPs K-means
6^{th}	0	Mtlb LMPs K-medoids
6^{th}	0	Mtlb LMPs HC
6^{th}	0	Cmzd LMPs GA
6^{th}	0	Cmzd LMPs PDC
6^{th}	0	Mtlb PTDFs K-medoids
6^{th}	0	Mtlb PTDFs HC

Therefore, according to *EEIs*, the methodology's suitable clustering algorithms which have been deployed during the previous seven tests are roughly discernible in three groups:

1. The partitioning methods that outperform all the others, namely the customized PTDFs-based K-medoids, the customized PTDFs-based K-means and the customized LMPs-based K-medoids.

2. The methods with low performance, from the 4^{th} to the 5^{th} position, i.e. the customized PTDFs-based *HC*, the Matlab PTDFs-based K-means and the customized LMPs-based *HC*.

3. All the other methods, namely, the algorithms tied for the 6^{th} place, whose zonal configurations reveal the weakest optimality by being 0 times the best ones.

Moreover, apart from rare exceptions like the customized LMPs-based K-medoids and the *HC*, it is glaring to observe that the LMPs-based clustering algorithms never manage to be the best according to *EEI* assessment. In fact, all of them are equal and are assigned the 6^{th} position with number of times best null, together with the Matlab PTDFs-based K-medoids and *HC*.

The best of all the zonal configurations' assessment criteria

This category orders the methodology's clustering algorithms depending on how many times they have been the best within the *CVI* and *EEI* assessments together. As a result, Table 4.13 provides the ranking in question.

Table 4.13: Ranking of the methodology's clustering algorithms over the 7 tests. According to *CVIs* and *EEIs* together.

Ranking	Number of times best	Clustering algorithm
1^{st}	12	Cmzd PTDFs K-medoids
2^{nd}	11	Mtlb LMPs HC
2^{nd}	11	Cmzd PTDFs K-means
3^{rd}	9	Cmzd LMPs PDC
4^{th}	8	Cmzd LMPs K-medoids
5^{th}	6	Cmzd LMPs K-means
5^{th}	6	Cmzd LMPs GA
6^{th}	5	Mtlb LMPs K-medoids
7^{th}	4	Mtlb LMPs K-means
8^{th}	3	Cmzd PTDFs HC
9^{th}	1	Cmzd LMPs HC
9^{th}	1	Mtlb PTDFs K-means
10^{th}	0	Mtlb PTDFs K-medoids
10^{th}	0	Mtlb PTDFs HC

Therefore, according to the *CVIs* and the *EEIs* together, the methodology's suitable clustering algorithms which have been deployed during the previous seven tests are roughly discernible in four groups:

1. The partitioning methods that outperform all the others placed from the 1^{st} to the 4^{th} position, i.e. the customized PTDFs-based K-medoids, the Matlab LMPs-based *HC*, the customized PTDFs-based K-means, the customized LMPs-based *PDC*, and the customized LMPs-based K-medoids.

2. Those which have intermediate performance, from the 5^{th} to the 7^{th} position, namely, the customized LMPs-based K-means, the customized LMPs-based *GA*, the Matlab LMPs-based K-medoids and the Matlab LMPs-based K-means.

3. Those which have low performance, represented by the 8^{th} and the 9^{th} position.

4. All the other methods tied for the 10^{th} place, whose zonal configurations reveal the weakest optimality by being 0 times the best ones.

The best of the sum of all the zonal configurations' assessment criteria

This category orders the methodology's clustering algorithms depending on the sum of all their *CVIs* and *EEIs* during the previously shown 7 tests. This is an interesting sorting, since for all the zonal configurations' assessment indicators, the more they are small, the more the judged *BAs* set is optimal, and hence the associated clustering algorithm is efficient in power networks partitioning. As a result, Table 4.14 provides the ranking in question.

Table 4.14: Ranking of the methodology's clustering algorithms over the 7 tests. According to the sum of both *CVIs* and *EEIs* for each partitioning technique.

Ranking	Sum result	Clustering algorithm
1^{st}	18.5251	Mtlb LMPs HC
2^{nd}	19.4009	Mtlb LMPs K-medoids
3^{rd}	19.4728	Cmzd LMPs K-means
3^{rd}	19.4728	Cmzd LMPs GA
4^{th}	20.7950	Cmzd PTDFs K-means
5^{th}	21.3574	Mtlb PTDFs HC
6^{th}	21.6084	Mtlb LMPs K-means
7^{th}	22.4036	Mtlb PTDFs K-medoids
8^{th}	23.2655	Cmzd PTDFs K-medoids
9^{th}	24.0593	Mtlb PTDFs K-means
10^{th}	26.7088	Cmzd LMPs HC
11^{th}	30.4496	Cmzd PTDFs HC
12^{th}	33.1477	Cmzd LMPs PDC
13^{th}	33.8623	Cmzd LMPs K-medoids

Therefore, according to the sum of both *CVIs* and *EEIs* for each partitioning technique over the 7 tests, the methodology's suitable clustering algorithms which have been deployed during the previous seven tests are roughly discernible in three groups:

1. The top three algorithms of the ranking, which outperform the others having a sum result lower than 20, i.e. the Matlab LMPs-based *HC*, the Matlab LMPs-based K-medoids, the customized LMPs-based K-means and the customized LMPs-based *GA*.

2. The partitioning methods whose zonal configurations reveal an intermediate optimality, namely those that have a sum result between 20 and 25.

3. All the other methods, having a final sum greater than 25, show the worst performance.

Combined view with colors by origin test

The combined view of the two types of indicators gathers, and hence permits to compare, all the 98 zonal configurations previously produced by the 14 methodology's algorithms during the 7 tests. In this first version the *BAs* sets are colored depending on the origin test, in order to ease the understanding of the reader. The non-dominated solutions found in this plot form the Pareto front. The 7 non-dominated points are specifically shown in Fig. 4.43.

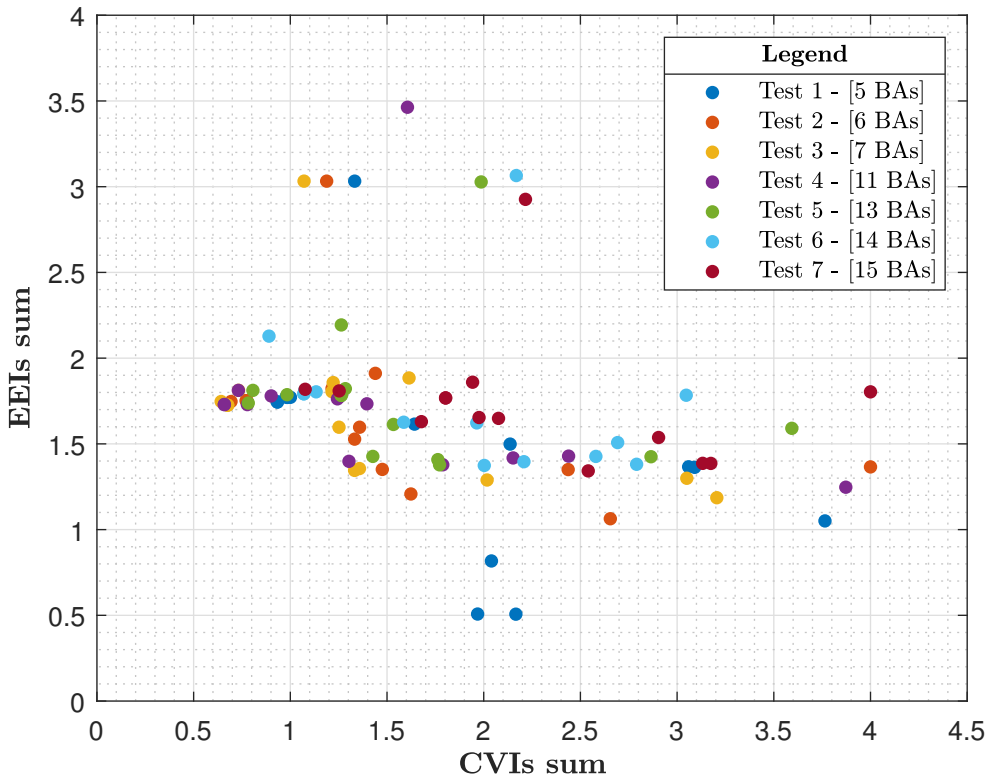

Figure 4.42: Combined view where the 98 zonal configurations' points are colored depending on the origin test.

As a result:

- The best zonal configurations are concentrated in the leftmost part of the points cloud and derive from different tests. The two solutions belonging to the Pareto front located in that area correspond to 7 and 11 requested *BAs*.

- The points of the second group of good zonal configurations stand on the central part of the Pareto front. Inside it there are *BAs* sets made up again of 7 and 11 requested *BAs*, plus a solution with 6 requested *BAs*.

- The third group of good zonal configurations, which closes the Pareto front, is placed in the lower part of the points cloud, roughly on the coordinates (2,0.5). It contains a sole zonal configuration coming from the first test, made up of 5 requested *BAs*.

For these reasons, it is interesting to note the absence from the Pareto front of any zonal configuration composed of 13, 14 or 15 requested *BAs*, whichever is the clustering algorithm that produce it.

The Fig. 4.43 provides an enlargement of this first colored version of the Pareto front, to facilitate its reading.

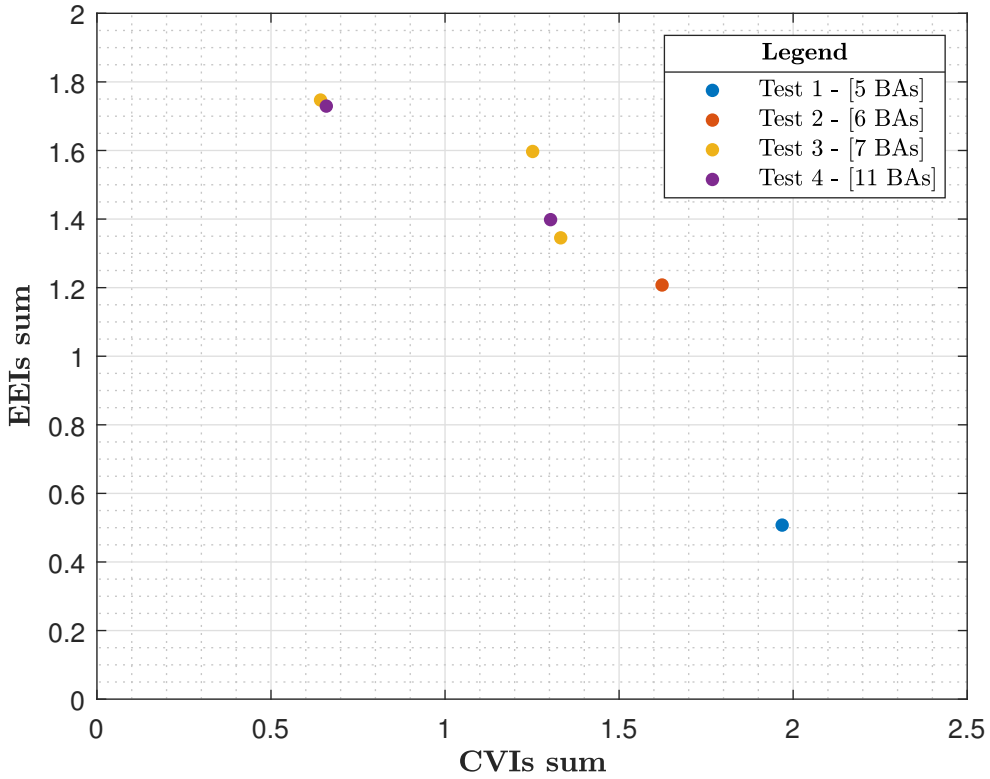

Figure 4.43: Enlargement of the Pareto front where the 98 zonal configurations' points are colored depending on the origin test.

Combined view with colors by clustering algorithm

The combined view of the two types of indicators gathers, and hence permits to compare, all the 98 zonal configurations previously produced by the 14 methodology's algorithms during the 7 tests. In this second version the *BAs* sets are colored according to the clustering algorithm, in order to ease the understanding of the reader. The Pareto front is the same as before, its 7 non-dominated points are specifically shown in Fig. 4.45 by using the coloration of this section.

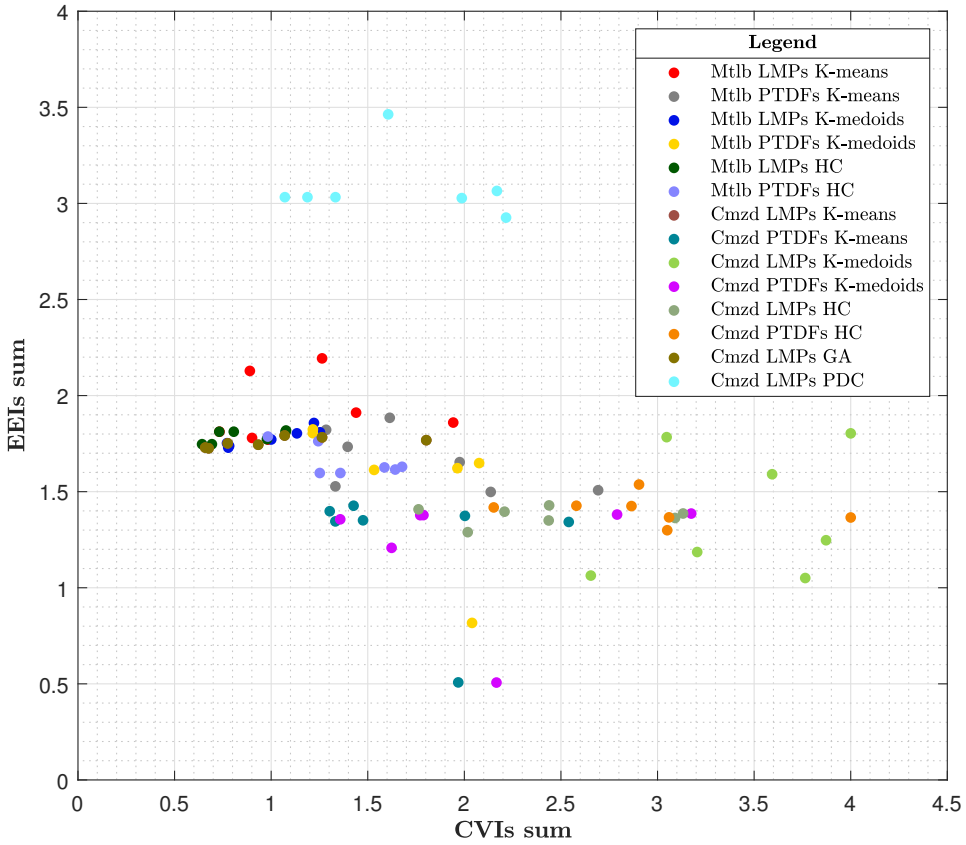

Figure 4.44: Combined view where the 98 zonal configurations' points are colored according to the clustering algorithm.

As a result:

- The best zonal configurations are concentrated in the leftmost part of the points cloud and derive from different tests. The two solutions belonging to the Pareto front located in that area are respectively produced by the Matlab LMPs-based *HC* and the customized LMPs-based *GA*. Nevertheless, the customized LMPs-based *GA* must be rejected as non-suitable clustering algorithm for defining optimal zonal

configurations, and must be substituted by the customized LMPs-based K-means, due to reasons that will be afterwards described inside *Section 5.1*.

- The points of the second group of good zonal configurations stand on the central part of the Pareto front. Inside it there is a *BAs* set produced by the Matlab PTDFs-based *HC*, two *BAs* sets created by the customized PTDFs-based K-means, and a *BAs* set coming from the customized PTDFs-based K-medoids.
- The third group of good zonal configurations, which closes the Pareto front, is placed in the lower part of the points cloud, roughly on the coordinates (2,0.5). It contains a sole zonal configuration produced by the customized PTDFs-based K-means.

For these reasons, it is interesting to note an affinity between LMPs-based clustering algorithms and *CVI* assessment or between PTDFs-based clustering algorithms and *EEI* assessment. This phenomenon will be subsequently described more in-depth inside *Section 5.3*, but anyway it is already visible inside Fig. 4.44. Since in that graph the LMPs-based zonal configurations and the PTDFs-based zonal configurations are respectively placed in the leftmost part of the points cloud, namely, where the *CVIs* sum is minimum, and in the lowest part of the points cloud, namely, where the *EEIs* sum is minimum.

The Fig. 4.45 provides an enlargement of this second colored version of the Pareto front, to facilitate its reading.

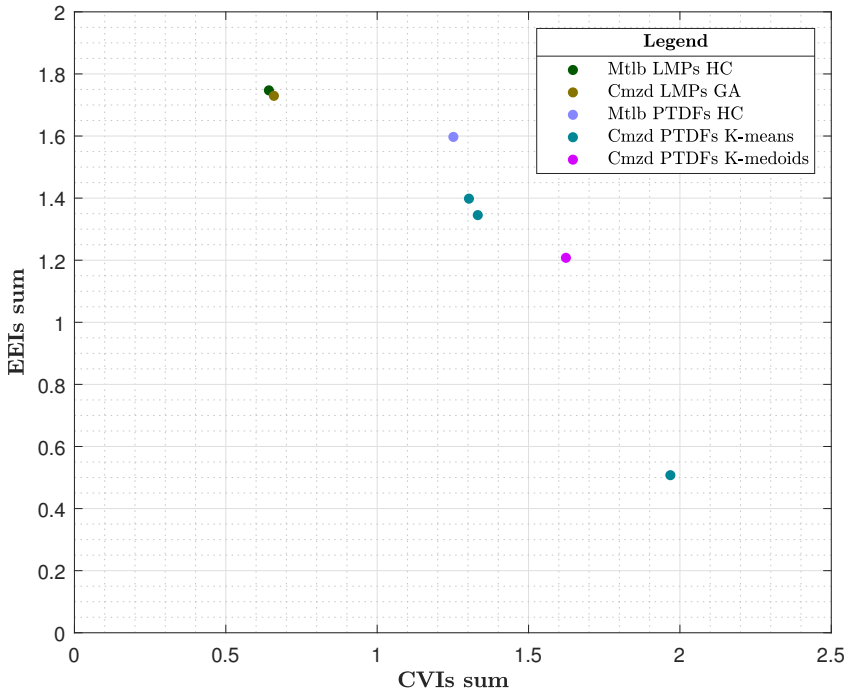

Figure 4.45: Enlargement of the Pareto front where the 98 zonal configurations' points are colored according to the clustering algorithm.

Eventually, by joining the considerations deriving from Fig. 4.44 with the ones previously done for the first version of the Pareto front portrayed inside Fig. 4.42, it results a group of compromise solutions which could be passed to a decision maker for the final choice. These zonal configurations, from left to right, are summed up into Table 4.15.

Table 4.15: Summary of compromise zonal configurations.

Number	**2^{nd} version information**		**1^{st} version information**
P1	Mtlb LMPs-based *HC*	with	7 requested *BAs*
P2	Cmzd LMPs-based K-means	with	11 requested *BAs*
P3	Mtlb PTDFs-based *HC*	with	7 requested *BAs*
P4	Cmzd PTDFs-based K-means	with	11 requested *BAs*
P5	Cmzd PTDFs-based K-means	with	7 requested *BAs*
P6	Cmzd PTDFs-based K-medoids	with	6 requested *BAs*
P7	Cmzd PTDFs-based K-means	with	5 requested *BAs*

Moreover, there are mathematical mechanisms which can help the decision maker during his/her choice. One of these mechanisms is the Analytic Hierarchy Process (*AHP*), which is subsequently applied to the just described Pareto front. The theoretical foundations of the following process come from reference [61].

The here considered Pareto front is made up of 7 points, previously summed up in Table 4.15 and recalled inside Fig. 4.46, to better understand the next steps.

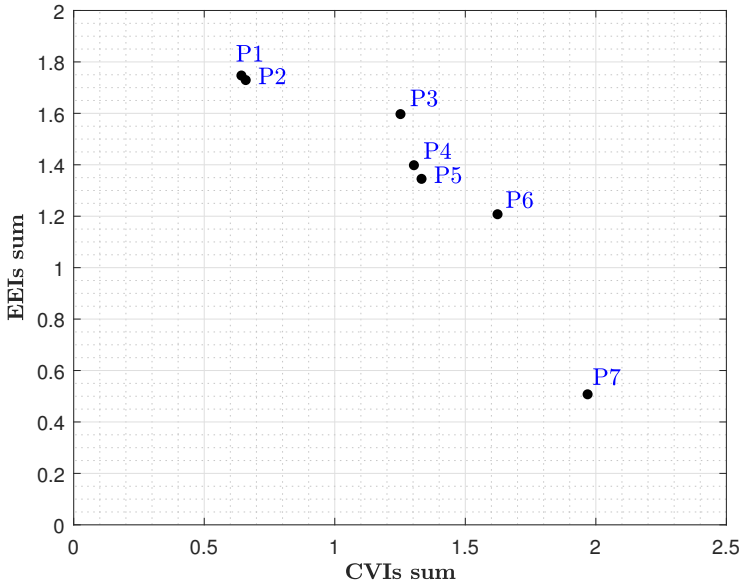

Figure 4.46: Pareto front recall, for the analytic hierarchy process

These 7 points have (*x*,*y*) coordinates contained into (4.1), which constitute the **B** matrix of the *AHP*.

$$\mathbf{B} = \begin{bmatrix} 0.6424 & 1.7471 \\ 0.6589 & 1.7294 \\ 1.2517 & 1.5971 \\ 1.3032 & 1.3983 \\ 1.3323 & 1.3453 \\ 1.6235 & 1.2077 \\ 1.9684 & 0.5075 \end{bmatrix} \tag{4.1}$$

Each column of the **B** matrix is thus associated to an objective. In particular, the *CVIs* sum objective is contained into the first column, whereas the *EEIs* sum objective is represented by the second column. It is necessary to look for both the minimum and the maximum value for each of these two columns, in order to pass from the **B** matrix to the **B'** one. Therefore, the minimum and maximum values for the two objectives are:

- $b_{min}^{CVI} = 0.6424$
- $b_{max}^{CVI} = 1.9684$
- $b_{min}^{EEI} = 0.5075$
- $b_{max}^{EEI} = 1.7471$

These values, respectively using the equation (4.2) and the equation (4.3) for the first and the second column of the **B** matrix, namely, for the *CVIs* sum objective and for the *EEIs* sum one, permit to pass from the **B** matrix to the **B'** one, which is shown inside (4.4).

$$\begin{cases} b'_{i1} = 1 + \frac{8*(b_{i1} - b_{min}^{CVI})}{b_{max}^{CVI} - b_{min}^{CVI}} \\ i = 1, \ldots, 7 \end{cases} \tag{4.2}$$

$$\begin{cases} b'_{i2} = 1 + \frac{8*(b_{i2} - b_{min}^{EEI})}{b_{max}^{EEI} - b_{min}^{EEI}} \\ i = 1, \ldots, 7 \end{cases} \tag{4.3}$$

$$\mathbf{B'} = \begin{bmatrix} 1.0000 & 9.0000 \\ 1.0999 & 8.8857 \\ 4.6759 & 8.0318 \\ 4.9865 & 6.7489 \\ 5.1625 & 6.4066 \\ 6.9189 & 5.5188 \\ 9.0000 & 1.0000 \end{bmatrix} \tag{4.4}$$

By using the **B'** matrix, it is possible to obtain the "pair comparison matrix" **D** for each objective. Therefore, the $\mathbf{D_{CVI}}$ and the $\mathbf{D_{EEI}}$ are respectively provided within (4.5) and (4.6).

$$\mathbf{D_{CVI}} = \begin{bmatrix} 1 & 1 & 1/5 & 1/5 & 1/5 & 1/7 & 1/9 \\ 1 & 1 & 1/4 & 1/5 & 1/5 & 1/6 & 1/8 \\ 5 & 4 & 1 & 1 & 1 & 1 & 1/2 \\ 5 & 5 & 1 & 1 & 1 & 1 & 1/2 \\ 5 & 5 & 1 & 1 & 1 & 1 & 1/2 \\ 7 & 6 & 1 & 1 & 1 & 1 & 1 \\ 9 & 8 & 2 & 2 & 2 & 1 & 1 \end{bmatrix} \tag{4.5}$$

$$\mathbf{D_{EEI}} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 2 & 9 \\ 1 & 1 & 1 & 1 & 1 & 2 & 9 \\ 1 & 1 & 1 & 1 & 1 & 1 & 8 \\ 1 & 1 & 1 & 1 & 1 & 1 & 7 \\ 1 & 1 & 1 & 1 & 1 & 1 & 6 \\ 1/2 & 1/2 & 1 & 1 & 1 & 1 & 6 \\ 1/9 & 1/9 & 1/8 & 1/7 & 1/6 & 1/6 & 1 \end{bmatrix} \tag{4.6}$$

The maximum eigenvalue of the $\mathbf{D_{CVI}}$ matrix is 7.0511. Whereas, the maximum eigenvalue of the $\mathbf{D_{EEI}}$ matrix is 7.0776. The eigenvectors corresponding to these eigenvalues are used to produce the decision matrix **A**, which is provided inside (4.7).

$$\mathbf{A} = \begin{bmatrix} 0.0704 & 0.4579 \\ 0.0756 & 0.4579 \\ 0.3540 & 0.4044 \\ 0.3647 & 0.3968 \\ 0.3647 & 0.3892 \\ 0.4405 & 0.3245 \\ 0.6356 & 0.0540 \end{bmatrix} \tag{4.7}$$

Eventually, by giving a weight coefficient equal to 0.5 to both the *CVIs* sum objective and the *EEIs* sum one, the $\overrightarrow{z}$ vector is calculated and shown within (4.8).

$$\overrightarrow{z} = \begin{bmatrix} 0.2641 \\ 0.2668 \\ 0.3792 \\ 0.3808 \\ 0.3770 \\ 0.3825 \\ 0.3448 \end{bmatrix} \tag{4.8}$$

The minimum value of vector $\overrightarrow{z}$ is 0.2641, which suggests to the decision maker the solution "P1" as the best one, namely, the zonal configuration created by the Matlab LMPs-based *HC* with 7 requested *BAs*. In order to strengthen this point of view, it is interesting to use also another mathematical mechanism called Technique for Order

of Preference by Similarity to Ideal Solution (*TOPSIS*). Therefore, the following lines present this mechanism's process, whose theoretical foundations are again contained inside reference [61]. The *TOPSIS* multi-criteria decision analysis method starts from the normalized decision matrix **R**, which can be easily taken equal to the previous **A** matrix of the *AHP*, since this latter was already normalized.

$$\mathbf{R} = \mathbf{A} = \begin{bmatrix} 0.0704 & 0.4579 \\ 0.0756 & 0.4579 \\ 0.3540 & 0.4044 \\ 0.3647 & 0.3968 \\ 0.3647 & 0.3892 \\ 0.4405 & 0.3245 \\ 0.6356 & 0.0540 \end{bmatrix} \tag{4.9}$$

Starting from the **R** matrix, which is provided into (4.9), it is created the auxiliary decision matrix **Z**, that is shown within (4.10). This passage is done by multiplying the two columns of the **R** matrix, which respectively stand for the *CVIs* sum objective and the *EEIs* sum one, for the respective weight coefficient, which are both set equal to 0.5 as already mentioned in the overlying lines.

$$\mathbf{Z} = \begin{bmatrix} 0.0352 & 0.2289 \\ 0.0378 & 0.2289 \\ 0.1770 & 0.2022 \\ 0.1824 & 0.1984 \\ 0.1824 & 0.1946 \\ 0.2202 & 0.1622 \\ 0.3178 & 0.0270 \end{bmatrix} \tag{4.10}$$

Inside the **Z** matrix the highest values of each column constitute the ideal positive solutions for each objective, which are gathered inside $\overrightarrow{z^+}$ vector provided within (4.11), whereas the lowest values of each column constitute the ideal negative solutions for each objective, which are collected inside $\overrightarrow{z^-}$ vector shown into (4.12).

$$\overrightarrow{z^+}^T = \begin{bmatrix} 0.0352 & 0.0270 \end{bmatrix} \tag{4.11}$$

$$\overrightarrow{z^-}^T = \begin{bmatrix} 0.3178 & 0.2289 \end{bmatrix} \tag{4.12}$$

The Euclidean distances from the ideal positive solutions, contained inside $\overrightarrow{z^+}$ vector, are calculated for each point by summing the Euclidean distances obtained for each objective, as shown within (4.13). In this way it is created the $\overrightarrow{\delta^+}$ vector, which is provided into (4.14).

$$\begin{cases} \delta_i^+ = \sqrt{\sum_{j=1}^{2} *(z_{ij} - z_j^+)^2} \\ i = 1, \ldots, 7 \end{cases} \tag{4.13}$$

$$\overrightarrow{\delta^+} = \begin{bmatrix} 0.2020 \\ 0.2020 \\ 0.2254 \\ 0.2259 \\ 0.2231 \\ 0.2292 \\ 0.2826 \end{bmatrix} \tag{4.14}$$

In the same way, the Euclidean distances from the ideal negative solutions, contained inside $\overrightarrow{z^-}$ vector, are calculated for each point by summing the Euclidean distances obtained for each objective, as shown within (4.15). In this way it is created the $\overrightarrow{\delta^-}$ vector, which is provided into (4.16).

$$\begin{cases} \delta_i^- = \sqrt{\sum_{j=1}^{2} *(z_{ij} - z_j^-)^2} \\ i = 1, \ldots, 7 \end{cases} \tag{4.15}$$

$$\overrightarrow{\delta^-} = \begin{bmatrix} 0.2826 \\ 0.2800 \\ 0.1433 \\ 0.1388 \\ 0.1397 \\ 0.1182 \\ 0.2020 \end{bmatrix} \tag{4.16}$$

Eventually, the normalized distances from the ideal negative solution are calculated for all the points, by using the equation contained inside (4.17). In this way it is created the final vector $\overrightarrow{\delta_{rel}^-}$, which is provided within (4.18).

$$\begin{cases} \delta_{i,rel}^- = \frac{\delta_i^-}{\delta_i^- + \delta_i^+} \\ i = 1, \ldots, 7 \end{cases} \tag{4.17}$$

$$\overrightarrow{\delta_{rel}^-} = \begin{bmatrix} 0.5832 \\ 0.5809 \\ 0.3886 \\ 0.3806 \\ 0.3851 \\ 0.3402 \\ 0.4168 \end{bmatrix} \tag{4.18}$$

Inside $\overrightarrow{\delta_{rel}^-}$ vector the maximum value indicates the solution that has the highest normalized distance from the ideal negative solution. This latter solution is the "best among the worst", which thereby becomes the best solution according to *TOPSIS* procedure. Therefore, in this case the *TOPSIS* process suggests to the decision maker the solution "P1" as the best one, namely, again the zonal configuration created by the Matlab LMPs-based *HC* with 7 requested *BAs*, as already stated by the previous *AHP* method.

Chapter 5

Conclusions

This chapter contains both the findings emerged by the application of this thesis' methodology on a real case study, which can be ascertained in the previous *Chapter 4*, and the suggestions for a possible future development of this work.

5.1 The *GA*'s false optimality

According to the rankings of methodology's clustering algorithms previously reported within *Section 4.3.3*, the customized LMPs-based *GA* is one of the most performing partitioning techniques. In fact, it reaches the 3^{rd} position inside Table 4.14 based on the sum of both *CVIs* and *EEIs*, which will be subsequently proved to be the most relevant ranking among the available ones for what concerns the higher placements. Consequently, it should be considered as one of the best clustering algorithms that have been used. Nevertheless, the *GA*'s good performance along the aforementioned 7 tests must be actually rejected as fake ones, and thereby the algorithm in question must be labeled as non-good at all.

This because, into the previous *Section 4.2.2* it has been stated the inadequacy of the *GA*'s starting population random initialization. Therefore, from there on out it has been decided to steer its initial exploration of the solutions' space, by half initializing its starting population through the zonal configuration coming from the customized LMPs-based K-means (before the application of both the additional handwritten functions "NoSingleNodeBAs" and "CheckBAsConnection"). This action was hence aimed at finally making the *GA* a performing clustering algorithm, without setting its stop criteria to very huge values. But actually, it just revealed to be the copy and paste of the customized LMPs-based K-means' zonal configurations to the customized LMPs-based *GA*'s ones. This can be easily seen by the previous Fig. 4.10, Fig. 4.11, Fig. 4.12 and Fig. 4.13, which contain the *CVIs* and *EEIs*' trends of customized algorithms' zonal configurations from 2 to 20 requested *BAs*, that reveal totally equal patterns for the two above mentioned algorithms. This is also seen in any of the figures provided within the 7 tests, where the *GA* shows always the same values and zonal configurations of its starting population's initializer. In other words, if the customized LMP-based *GA*'s starting population is randomly initialized, its zonal configurations become totally senseless as previously proven

inside *Section 4.3.3*. But meanwhile, if the same starting population is not randomly initialized, the customized LMPs-based *GA* does not add anything respect to the clustering algorithm which partly initializes its population. Thereby, its resulting price zones still remain useless and devoid of any interest.

At this point, it could be asserted that also this latter problem may be solved by setting huge stop criteria to the *GA*, just like already said for the first problem linked to the population random initialization. Nevertheless, it seems more likely to see the *GA* as a good *OP* solver but a bad clustering algorithm. Because experimentally, during all the tests which have been done during these months of research, the *GA* used as clustering algorithm has never created particularly optimal zonal configurations. Indeed, it has always revealed very poor performance, whether randomly initialized or not. Therefore, the *GA* understood as this thesis' one must realistically be refused as non-suitable technique for defining optimal zonal configurations. The only possibility to re-evaluate it as clustering algorithm for the aforementioned purpose may come from some modifications, like for instance:

- The mutation process change, so that it becomes a genetic operator working only on *BAs*' border nodes.
- The crossover process change, so that it becomes a genetic operator moving physically cohesive *BAs*' portions from one prize zone to another adjacent one.
- The crossover probability change, so that the associated genetic operator becomes less probable than the current high value of 90%.

Using a *GA* customized this way would surely give better results than the ones obtained through this thesis' methodology because, thanks to these changes, the algorithm's randomness would be both limited to avoid defining unfeasible zonal configurations, similar to color palettes, and effectively used to explore the solutions' space, to try finding better zonal configurations.

5.2 The outsider *PDC*

This section has the overlying strange title because, as already explained within the previous *Section 3.2*, the *PDC* is the clustering algorithm which tries to obtain an optimal zonal configuration in the easiest possible way. Namely, it merges the power network nodes when both their average *LMPs* difference drops below a certain user-defined range and they result to be physically connected. Therefore, due to its extreme simplicity, this partitioning technique could have been even regarded as an outsider at the beginning of this thesis.

Made this clarification, the ranking included inside Table 4.13 points out the *PDC* as one of the methodology's best algorithms. But actually, since this latter sorting will be afterwards labeled as non-accurate for what concerns the higher positions in favor of the one contained into Table 4.14, the 3^{rd} place in question must be rejected as non-relevant one, and must be substituted with the algorithm's 12^{th} placement of Table 4.14. As a

result, the *PDC*'s bad positioning inside this latter ranking becomes the first reason of its inaccuracy.

Moreover, by looking at the geographical representations of the *PDC*'s zonal configurations contained into the 7 tests, these last never reveal *BAs* sets which may be effectively applicable. Because from having 8 price zones, associated to a 26% of average *LMPs* tolerance, to having 15 of them, associated to a 6% of the same parameter, there is always an excessive contrast between a too high majority of very small *BAs* (especially concentrated into the higher part of Scandinavian countries) and a too small quantity of big price zones. In other words, the *BAs*' sizes sensibility towards the user-defined average *LMPs* tolerance is too low. And consequently, he cannot prevent the *PDC* from defining too different price zones in terms of dimensions. This condition, which is obviously non-optimal for many reasons, can be avoided only naturally thanks to the intrinsic features of the partitioned power network (thing that does not happen inside this case study's reduced model of the European transmission network). Therefore, this is the second reason of the *PDC*'s inaccuracy, which furthermore had already been noted within reference [17] as previously reported inside pros and cons of *Chapter 2*.

Eventually, as can be seen from the Pareto front portrayed within Fig. 4.44, the *PDC*'s zonal configurations obtained during the 7 tests outlined in *Section 4.3.2* are markedly worse than all the other clustering algorithms' *BAs* sets, whichever is the requested *BAs* number. For these three reasons, it is reasonable to refuse the *PDC* as non-suitable technique for defining optimal zonal configurations.

5.3 PTDFs-based algorithms vs LMPs-based ones

It is interesting to note that the LMPs-based clustering algorithms, both coming from Matlab commands or customized source codes, create zonal configurations with better optimality performance from the *CVIs*' point of view. Conversely, the Matlab and customized PTDFs-based clustering algorithms produce *BAs* sets with better optimality performance from the *EEIs*' point of view. This can be easily seen by the previous Table 4.11 and Table 4.12. In the first one, all the partitioning techniques which have been the best for at least once are LMPs-based clustering algorithms. In the second one, the same bests for at least once are PTDFs-based clustering algorithms 4 times out of 6.

It is difficult to give a reason to this phenomenon, but empirically it seems that using the nodal *PTDFs* of most congestible lines as clustering feature makes the respective algorithm produce price zones with lower market power and hence higher competition level, which thereby are more performing from the *EEIs*' point of view. On the contrary, using the *LMP* hourly trends as clustering feature permits to obtain *BAs* that retain better the nodal prices' benchmark economic signals. This is proven by their small values of *CVIs*, which are remembered to be always computed on *LMP* hourly trends, whether the judged algorithm is LMPs-based or PTDFs-based, and consequently they give an estimation of the within-clusters *LMP* variance.

5.4 The optimality exclusivity

By looking at the previous Table 4.11 and Table 4.12, it emerges that there is no methodology's suitable clustering algorithm which has been at least once best inside both the categories of zonal configurations' assessment indicators. Namely, if a partitioning technique has been the best inside *CVIs*' evaluations for a certain number of times greater than zero, it has never been the best according to *EEIs* along all the 7 tests made. Obviously, it is the other way around for the clustering algorithms which have been the best inside *EEIs*' evaluations for at least once. Therefore, in other words, for all the methodology's partitioning techniques having been the best for even just once within one of the two assessment indicators' categories has then precluded it the possibility to be the best in the other category, during all the case study's tests.

This consideration becomes even more interesting when read as an extension of the previous one, reported inside *Section 5.3*. In fact, whether on the one side this latter concerns the affinity between LMPs-based clustering algorithms and *CVIs* or between PTDFs-based ones and *EEIs*, which anyway is not so severe as proven by the customized LMPs-based K-medoids and *HC* within Table 4.12. The here outlined observation widens the aforementioned point of view and simultaneously becomes more strict. By saying that, whichever is the nature of the partitioning technique, it can only be the best inside one of the two assessment indicators' categories, and never in the other one.

For these reasons, the ranking portrayed inside Table 4.13 slightly loses its importance. In fact, it proves to be not a classification that truly points out the overall best partitioning technique, according to both *CVIs*' evaluations and *EEIs*' ones. But instead, just a comparison among the algorithms' first places which however have been merely collected within their respective optimality fields. Therefore, from this latter point of view the best clustering algorithm can also emerge due to the remarkable inadequacy of the opponents inside the two separated optimality fields in question (namely the *CVI* assessment and the *EEI* one), rather than for its real merit within a common evaluation field, as instead it should be in order to make relevant the ranking contained into Table 4.13.

From the above discussion, it emerges that the sum of *CVIs* and the sum of *EEIs* can be considered to some extent as conflicting objectives, to be handled together by drawing the Pareto front.

5.5 The methodology's worst clustering algorithms

At the end of the above *Section 5.4* it is stated that the ranking contained into Table 4.13 is not so accurate, because of reasons explained there. Nevertheless, this is only true for the higher placements, namely to elect the overall best partitioning technique. In fact, the lower part of the ranking is completely truthful. Therefore, it is perfect to point out the methodology's worst clustering algorithms. As a result, the table in question nominates the Matlab PTDFs-based K-medoids and *HC* as worse, closely followed by the Matlab PTDFs-based K-means and the customized LMPs-based *HC*, that manage to be the best for only once during all the case study's seven tests.

Before making considerations on this result, which could be distorted by particular conditions of that specific ranking, it is better to also look for the worst clustering algorithms emerged by another sorting. Therefore, according to Table 4.14 where the final ranking has been created by summing all the zonal configurations' *CVIs* and *EEIs* along the seven tests, the worst partitioning techniques have been the customized LMPs-based K-medoids, *PDC*, *HC* and the customized PTDFs-based *HC*. It is worth remembering that it is not necessary to also look for the worst clustering algorithms inside the two Pareto fronts of Fig. 4.42 and Fig. 4.44, because these last have been created using *CVIs* sum on the horizontal axis and *EEIs* sum on the vertical one. Hence, they roughly provide the same information of Table 4.14, just represented in another way.

For these reasons, the aforementioned assertions lead to the following considerations:

- The customized LMPs-based *HC* shows poor performance according to both Table 4.13 and Table 4.14. Therefore, there are valid reasons to consider it as one of the methodology's worst clustering algorithms, together with the customized LMPs-based *GA* and *PDC*, for reasons previously reported respectively inside *Section 5.1* and *Section 5.2*.

- The clustering algorithms which outperform on the long run, namely considering the sum of all their zonal configurations' *CVIs* and *EEIs* along the case study's seven tests, typically perform worse in the one-shot bests, whose ranking is reported inside Table 4.13. This can be easily seen by looking for instance at the Matlab PTDFs-based K-medoids and *HC*. That, from the 10^{th} positions within Table 4.13, respectively pass to the 7^{th} and the 5^{th} place into Table 4.14. Or even the other way around, by looking at the customized LMPs-based K-medoids, *PDC* and *HC*. That, from the 13^{th}, 12^{th} and 11^{th} position within Table 4.13, respectively pass to the 4^{th}, 3^{rd} and 8^{th} place into Table 4.14.

5.6 The methodology's best clustering algorithms

Since the higher part of the ranking contained inside Table 4.13 must be rejected as non-accurate, for reasons previously explained inside *Section 5.4* and *Section 5.5*, the methodology's best clustering algorithms have to be nominated by looking at the sorting of Table 4.14. Thereby, this latter points out a podium made up of:

- 1^{st} place: Matlab LMPs-based *HC*
- 2^{nd} place: Matlab LMPs-based K-medoids
- 3^{rd} place: Customized LMPs-based K-means

The equal merit in the third place between the customized LMPs-based K-means and *GA* is voluntarily missing, because of *GA*'s fake optimality reasons previously explained inside *Section 5.1*. Moreover, the aforementioned podium is also confirmed by Fig. 4.44, where the best zonal configurations concentrated in the leftmost part of the points cloud actually derive from Matlab LMPs-based *HC*, K-medoids and customized LMPs-based K-means. As previously stated inside *Section 4.3.3*. Consequently, in order to comment this twice obtained result, the following sections contain some considerations.

5.6.1 Nodal *PTDFs* of most congestible lines as valid clustering feature

First of all, the aforementioned podium does not reveal an important feature, which however is worth remembering. In fact, even if not placed in the first three positions, the PTDFs-based clustering algorithms show good performance, as proved for instance by the customized K-means and the Matlab *HC* fed through this clustering feature, which manage to reach respectively the 4^{th} and the 5^{th} placement within the ranking of Table 4.14, furthermore detached of few units respect to the first three clustering algorithms. Therefore, this can be considered as the consistency proof in using the nodal *PTDFs* of most congestible lines as clustering feature, in alternative to the classical *LMP* hourly patterns. Action which has been explained previously from the theoretical point of view, within *Section 3.2.1*, and moreover has already been used for many years inside the scientific literature regarding the optimal price zones definition subject. But nevertheless, it had never been compared to LMPs-based zonal configurations to assess its effectiveness, as instead has been done inside this thesis.

5.6.2 The penalty factor technique's double face

The penalty factor technique, previously explained inside *Section 3.2.2*, has been proved to be insufficient to make the clustering algorithms comply with the zonal configurations' 9^{th} optimality requirement, within *Section 4.2.1*. As a result, from this latter section on out the methodology's customized partitioning techniques have added to their processes the downstream application of the additional handwritten function "CheckBAsConnection", in order to preserve their reason for being among the methodology's suitable clustering algorithms (apart from the customized LMPs-based *PDC*, for reasons above reported inside *Section 3.2.2*).

Made this premise, it is interesting to note how the penalty factor technique conservation within the customized partitioning techniques (apart the *PDC*) has had both good and bad consequences in terms of resulting zonal configurations' optimality, as proven by the ranking of Table 4.14. In fact, for instance, the PTDFs-based and LMPs-based K-means respectively pass from the 9^{th} and 6^{th} position to the 4^{th} and 3^{rd} one, by passing from the Matlab versions to the customized ones of their clustering algorithms. But the other way around, there are partitioning techniques like the Matlab LMPs-based *HC* and K-medoids, which respectively pass from the 1^{st} and 2^{nd} placement to the 10^{th} and 13^{th} one, by making the same change. Therefore, according to these results, on the one hand it seems that actually the K-means clustering gains benefit from the preliminary application of the penalty factor technique. Which, even if not exhaustive respect to the compliance towards the zonal configurations' 9^{th} optimality requirement, manages to render less invasive the alteration of its final *BAs* set, coming from the usage of the handwritten function "CheckBAsConnection". Whereas, on the other hand, it seems that other partitioning techniques like the *HC* and the K-medoids actually acquire drawbacks by the presence of the penalty factor technique in addition to the downstream application of the aforementioned handwritten function.

Given these facts, it is not yet possible to assert a complete affinity between clustering algorithms like *HC* or K-medoids and the preliminary usage of the penalty factor

technique, because of the exiguity of the data on which the consideration in question would be founded. Nevertheless, in addition to this technique's insufficiency in making the clustering algorithms comply with the zonal configurations' 9^{th} optimality requirement previously reported inside *Section 4.2.1*, the overlying findings prove that even its additional use ex-ante may not be always profitable to be done. Therefore, it must only be undertaken after having proved its effectiveness through comparisons like this thesis' one.

5.6.3 The podium

According to the scientific literature regarding the subject of this thesis, namely the usage of clustering algorithms for defining power networks' optimal zonal configurations, the most frequently used partitioning technique for the aforementioned purpose is the *HC*, as indicated in Table 2.2, where this latter algorithm shows to be used seven times within the papers considered in that chapter. Therefore, the 1^{st} place obtained by the Matlab LMPs-based *HC* within the ranking of Table 4.14 is surely important because it gives a practical demonstration of the validity of what is usually done in the scientific literature from many years.

Nevertheless, the 2^{nd} placement of the same ranking contains a K-medoids clustering. Which instead has been used only once inside the above mentioned documents, i.e., this thesis' references previously summed up into *Chapter 2*. Consequently, this second result becomes equally interesting, because it proves the existence of a not yet traveled direction of research that could reserve particularly interesting performance.

Thirdly, the 3^{rd} position of the customized LMPs-based K-means inside the aforementioned ranking of Table 4.14 still offers the possibility for a twofold consideration. In fact, on the one hand it demonstrates the possible advantage that a clustering algorithm can acquire from the joint use of the penalty factor technique and an ex-post control of its *BAs*' physical integrity, which however has already been outlined inside *Section 5.6.2*. On the other hand, it proves the rationale of using the K-means clustering for making optimal zonal configurations, which is an already widespread approach thanks to K-means' notoriety as proven by Table 2.2.

Eventually, it is still worth remembering that the whole of the podium's partitioning techniques is LMPs-based. Therefore, this unavoidably becomes the tangible validity proof of the highly diffused behavior of using *LMP* hourly trends as clustering feature.

5.7 Final thoughts and future developments

The ultimate goal of this thesis was not the creation of a truly optimal zonal configuration for a certain power network, but rather the search for "the most suitable technique to deterministically define an optimal zonal configuration", as already stated inside the *Summary*. In fact, it has not been especially commented any absolute best zonal configuration emerged inside *Section 4.3.3* by joint analysis of Fig. 4.44 and Fig. 4.44.

Therefore, in this sense the previous pages undertake this path by using clustering algorithms, since these last reveal to be the only rational possibility to carry out the aforementioned purpose, and hence they compare the performance of some of them through the usage of a real case study, based on a reduced model of the European transmission electricity grid. From this latter, several findings emerge, like the praise of customized LMPs-based K-means and Matlab LMPs-based *HC* and K-medoids, or the rejection of customized LMPs-based *GA* and *PDC*. But nevertheless, it is not possible to distinguish the whole of the available clustering algorithms into a multitude of totally inappropriate partitioning technique and a sole completely perfect algorithm. Because all of their performance are roughly comparable, and hence it is difficult to entirely discard one of them (apart few cases like the aforementioned *GA* and *PDC*).

In other words, the final consideration of this thesis should be "the clustering algorithms are not the magical box". Because, in order to define electricity grids' optimal zonal configurations, many of them may be suitable, whether in their Matlab or customized version, according to the nomenclature many times adopted inside this work, and fed with nodal *PTDFs* of most congestible lines or *LMP* hourly trends. It is all up to the specific real power network, which however cannot be defined a priori, and the zonal configurations' optimality assessment criteria that wants to be used. But this latter, as far as this thesis' author is concerned, is the real Achilles heel on which the *EU* should work in order to improve the optimal *BAs* definition within the electricity grids around the continent. Because once that the zonal configurations' optimality requirements will be clearly classified through unique tangible parameters, both objective and quantitative, the price zones assessment and thereby their optimal definition will become much more easier for all the European *TSOs*. But at the same time, until this lack will not be filled up, the clustering algorithms will not be able to define uniquely optimal zonal configurations, and hence they will not be discernible in totally good or bad ones.

From this latter point of view, the actual contribution of this thesis lies in having created a three-level methodology aimed at clearly stating the zonal configurations' optimality requirements, finding the most suitable clustering algorithms to comply with the just mentioned criteria (modified if necessary) and comparing the performance of these last through univocal zonal configurations' optimality assessment indicators (discernible into *CVIs* and *EEIs*).

Beyond this considerations, in order to hereafter improve this research of the most suitable clustering algorithm to deterministically define an optimal zonal configuration, some suggestions are afterwards presented.

- **To expand the time window.**
  Making an analysis similar to this thesis' one on a time window greater than a week indeed, maybe considering also an higher number of future scenarios, could bring to a twofold betterment. On the one side, the resulting zonal configurations would acquire a better temporal stability and thereby would improve from the zonal configurations' 1^{st} optimality requirement point of view. On the other side, the final considerations of a research of that type would unavoidably have a wider general sense.

- **To spread the view on clustering algorithms and distance metrics.**
  In fact, especially this latter field has been little explored inside this work, since it has only been used the Euclidean distance. As a result, it could be surely interesting to see how the usage of other distance metrics, like those more typical of load profiles clustering, may change and perhaps improve the zonal configurations definition.

Appendix A

Secondary references analysis

This appendix contains the detailed overview of this thesis' references which adopt clustering techniques for purposes different from *BAs* definition. This overview is organized with both tables and bulleted lists, in order to ease the consultation to the reader.

A.1 References summary

The following bulleted list contains a short description for each of the considered papers which deal with clustering algorithms but do not define any zonal configurations. It is organized with two subpoints for each reference, which respectively contain its rationale and its general description.

- **[6] Cao et al. (2018)**
  - **Paper's rationale:** To define a reduced model of the power network. Based on preserving system's congestions, and aimed at simulating its operation with less computational effort.
  - **Paper's summary:** This papers firstly states that the models used for analyzing and developing future energy systems must be simplified, due to their too high computational burden, and must include the modeling of power network's congestions, which are often ignored. Therefore, it is presented a new methodology for aggregating, hence simplifying, spatially highly resolved transmission grid information for energy system models, which is based on preserving system's congestions, and is aimed at simulating its operation with less computational effort. The aforementioned reduction process is mathematically done through a spectral clustering algorithm, fed with *LMPs* snapshots of significant power network's moments from the congestions point of view. In fact, these last are respectively evaluated:
    * at the hour of the year for which the maximum of the sum of the generated power from wind onshore and the load can be observed, representative of a particularly loaded moment for the power network;
    * at the hour of the year for which the maximum of the nodal price differences can be observed;

  - ∗ at the hour of the year for which the maximum of the relative grid transfer capacity usage can be observed.

- **[7] Cotilla-Sanchez et al. (2013)**
  - **Paper's rationale:** To present a hybrid clustering method to do electrically coherent partitionings of power networks. It adopts Electrical Distance (*ED*) as distance metric. Most of all, it is a general purpose algorithm, which can be tailored to most of specific applications thanks to its multi-attribute objective function.
  - **Paper's summary:** This papers observes that conventional partitioning algorithms, such as spectral and K-means clustering, are computationally efficient, but they are not easily adaptable to produce solutions aimed at optimizing objectives beyond those of maximizing between-clusters distances or minimizing within-clusters ones. Therefore, this document presents a hybrid clustering algorithm to cover this deficiency, made up of a preliminary K-means process and a subsequent genetic algorithm. The former is used to generate an initial set of candidate solutions. The latter is used to improve these solutions according to the fitness function, namely the user-defined multi-attribute objective function focused on pushing the final clustering result beyond the aforementioned two typical clustering objectives. This last multi-attribute objective function is composed of the weighted product of the partitioning quality measures initially defined by the user, which are here five indices dealing with *ED*, clusters sizes and nodes' connection. The distance metric used inside the here proposed methodology is the *ED*, also used within some of the previous clusters assessment criteria. This differs substantially from the more typical topological distance in power grids, by relating network topology to active-power sensitivities. For more information about its definition, please look at the following section A.4.

- **[10] Ferreira et al. (2011)**
  - **Paper's rationale:** To divide power network's nodes into clusters according to their *LMPs*, to take advantage of their economic signals in helping the *TSO* in defining future investments and network expansions.
  - **Paper's summary:** The *LMPs* are the clearest and most objective economic signals which can be used to price energy inside a power network. Thus, they represent the benchmark in this field, reason why they want to be preserved as much as possible when passing from *NP* to *ZP*. For this reason, in this paper power network's nodes are divided in groups by using a K-means clustering and a two-step one, both fed with *LMPs*. The goal is to find sets of nodes where, according to aforementioned benchmark economic signals, it would be profitable for system's performance improvement to increase generation or demand. These information become then useful to allow the *TSO* to choose the future expansion plan which increases the most the system's performance, e.g. improving its congestion management. The aforementioned methodology

is eventually applied to a real case study, namely the Californian database of 2009's hourly *LMPs* patterns, in order to assess its effectiveness.

- **[11] Ferreira et al. (2010)**
  - **Paper's rationale:** To divide power network's nodes into clusters according to their *LMPs*, to take advantage of their economic signals in helping the *TSO* in defining future investments and network expansions.
  - **Paper's summary:** Broadly speaking, this paper is quite similar to previous reference [10]. Indeed, they share all the authors and the period of publication is rough the same. Just slight differences are present, and thus are subsequently reported. On the one hand, here more attention is laid on pre-processing of input data, which are always the 2009's hourly *LMPs* patterns of the Californian power network. In this preliminary phase, all the physically incoherent trends are ousted from the input database. Namely *LMPs* with too many missing points in their course or with patently senseless values. On the other one, here it is also included a comment on the computational performance of both the used algorithms, namely, the two-step method and the K-means. This new comparison oddly reveals a faster performance of K-means clustering, even if two-step algorithm should be actually born to speed up the clustering of large databases like the one here used. Nevertheless, this comparison is not truthful because made between different software. Therefore, it must not be considered. Beyond these two differences, the remaining part of the paper roughly says the same things of previous reference [10]. Also the case study is the same.

- **[16] Hong et al. (2002)**
  - **Paper's rationale:** To forecast power network *LMPs*, so as to give market participants helpful information to develop their bidding strategies. In this scenario, a fuzzy-c-means clustering algorithm is used on load levels, in order to classify them in three categories, before a Recurrent Neural Network (*RNN*) is trained on each load level.
  - **Paper's summary:** This papers tries to instruct a neural network focused on *LMPs* forecasting. This could give helpful information to market participants, in defining their bidding strategies. Therefore, firstly the transaction periods are divided into three clusters by a fuzzy-c-means clustering algorithm. These last are respectively peak load periods, medium-peak load ones and off-peak load ones. Then a *RNN* is trained on each of them, using historical data from PJM (Pennsylvania, New Jersey and Maryland) power network. Eventually, these neural networks are tested, revealing to be capable of efficiently forecasting *LMPs* values. It is worth remembering that, also a more traditional Neural Network (*NN*) is here studied for comparison. But this latter always shows worse performance. And this was predictable, since *RNN* is capable of modelling nonlinear and fast variations as well as complicated input/output relationships, just like the here considered *LMPs* forecasting is.

- **[20] Kiran et al. (2016)**
  - **Paper's rationale:** To produce the archetypes of the twenty-four *BAs*' zonal prices of both the two power exchanges which separately operate in India, so as to give respective market participants helpful information to develop their bidding strategies.
  - **Paper's summary:** This papers firstly admits the importance of predicting market clearing prices inside a deregulated environment. In fact, this information can help market participants to optimize their bidding strategies so as to maximize their profits. Nevertheless, this forecast is quite challenging due to energy price's unpredictability. And moreover, it is even harder in some cases, like the Indian system. In this latter indeed, the power network is split in two and separately operated by two independent power exchanges, respectively named the Indian Energy Exchange and the Power Exchange of India Limited. In this scenario, this paper uses clustering techniques to obtain the typical zonal prices profiles over a year of the twenty-four Indian *BAs*, respectively divided between the two aforementioned markets. These last come out as clusters' centroids of a K-means clustering algorithm, fed with Zonal Marginal Price (*ZMP*) patterns of the twenty-four Indian *BAs* along a year and speeded up by a preliminary Principal Component Analysis (*PCA*) process run before the K-means. The resulting seven archetypal patterns can be used by Indian players for strategic bidding purposes.

- **[23] Klos et al. (2015)**
  - **Paper's rationale:** To propose a clustering approach aimed at modifying a starting zonal configuration, in order to decrease its internal loop flows. These last are unscheduled power flows, which compromise system efficiency, stability and security, and hence they should be reduced as much as possible inside an optimal zonal configuration.
  - **Paper's summary:** This paper presents a clustering method aimed at minimizing the loop flows of a power network. These last are unscheduled flows, consisting of power flows transmitted through neighbouring zones but due to some intra-zonal transactions. They are not desired for many reasons, e.g. they introduce additional losses and they decrease transfer capacity of the affected lines, thus threatening even some reserve zones not to provide all the reserve capacity they were attributed with, by the *TSO*. Therefore, loop flows are unwanted since they decrease both system's stability and security. Anyway, there is also another type of unscheduled flow, which is called transit flow. This latter is less worrying, because it is an inter-zonal power flow which involves the two zones where it takes place respectively the injection and withdrawal of the exchanged amount of power. So that it is naturally managed by the *OP* which defines the zonal market clearing of the zonal-based market. This is why this paper proposes a clustering method to modify a starting zonal configuration in order to transform its loop flows, hardly manageable, into transit

ones, naturally managed by the zonal market clearing. For doing this, the aforementioned methodology has firstly to identify the power network's *BA* which is responsible for the most of system's loop flows, so as to straightaway operate on the zone which most affects system performance, namely the zone with the highest number of these unscheduled flows. Then, each node of this targeted zone is labeled with the impact that its power injection or withdrawal has on the zone's loop flows, which are so much present into this zone. Therefore, these aforementioned nodes are classified through something that recalls *PTDFs*. After that, these nodal features are put inside a hierarchical clustering algorithm, to split out this targeted zone into two *BAs*, in the way that the largest possible part of the previous loop flows is transformed into transition ones, more manageable. Thus increasing system performance as much as possible. It is worth remembering that, the here proposed methodology aimed at minimizing the loop flows, comes as an additional step in the problem of *BAs* redefinition. Indeed, its starting point is a zonal configuration which has to be previously defined in whatever way. Therefore, it has to be intended as a clustering algorithm for zonal configuration improvement rather than zonal configuration definition.

- **[24] Koivisto et al. (2012)**
  - **Paper's rationale:** To cluster Finnish load profiles, by using a K-means algorithm speeded up by a preliminary *PCA*, and to create a model of the main customer groups, by using a multiple repression analysis based on two largest clusters of the load profiles partitioning.
  - **Paper's summary:** In this paper, it is performed a clustering of Finnish load profiles using a classical K-means algorithm preceded by a *PCA*, aimed at speeding up the clustering process. After this first partitioning, a multiple regression analysis is carried out on the two largest clusters, to find the most important explanatory factors for the load modeling. These last reveal to be the temperature, the day length, mainly in a linear way or close to it through a piecewise linear approximation, and the day type, modeled by using dummy variables so as to reproduce sudden changes in the load profiles. This model of the main customer groups, since it is based on two largest clusters of the load profiles partitioning, reveals to be very important to assist the Distribution System Operator (*DSO*) during the long-term development of the power system.

- **[28] Sanchez-Garcia et al. (2014)**
  - **Paper's rationale:** To partition the power network in order to create a reduced model of it, which is both able to reduce the computational effort needed to treat it, and to preserve a user-defined feature of the actual power network. This latter can be the internal connectivity structure of the underlying network, when choosing lines admittances as lines' weight, or the existence of islands, when choosing average real power flows as lines' weight.

  - **Paper's summary:** This paper's aim is to partition the power network into smaller parts, so as to manage it in an easier way. Hence it does not deal with *BAs* redefinition, but with system treatment simplification. This concept goes back to the 1950s, when computers' memory was limited and so the systems' dimensions had to be contained to become more easily manageable. Therefore, here it is proposed a spectral clustering algorithm aimed at reducing the power network's model. Inside this latter, depending on the chosen lines' weight, different Laplacian matrices are obtained, and hence different features of the original power network are maintained in the final reduced model. Therefore, when using lines admittances, it is obtained a certain Laplacian matrix. The resulting final reduced model reveals the static internal connectivity structure of the underlying network. When using average real power flows, it changes the Laplacian matrix. The resulting final reduced model highlights the presence of islands, namely energetically autonomous zones.

- **[30] Shayesteh et al. (2014)**

  - **Paper's rationale:** To create a reduced model of power network, able to simulate its operation in an easier way meanwhile preserving its overall behavior. This is done through a spectral clustering algorithm fed with nodes' Available Transmission Capacity (ATC) coefficients and embedded with a K-means.
  - **Paper's summary:** This paper aims to define a methodology to create reduced models of existing power networks. This must be able to simulate in an easier way the operation of the respective power networks, meanwhile preserving their behaviors as much as possible. This analysis is important since simulations of prices, power flows and production costs are all crucial inputs to generation and transmission planning studies. Moreover, to calculate average system performance for many alternatives over long time periods, namely for different scenarios, it is necessary to simulate large numbers of hourly combinations of renewable production and loads across large regions. But all of these things require to simulate the power network's operation along a certain period, and this is usually impractical for full networks, due to the too high computational burden associated to their detailed models. Therefore, a power network reduction is needed in order to simplify the simulation of system's operation. This paper actually deals with this problem. In particular, it describes an innovative spectral clustering algorithm fed with nodes' ATC coefficients and embedded with a K-means as clustering step. This methodology is eventually applied on two case studies, to test its effectiveness. These last are the *IEEE* 118-bus test system and the Polish 3121-bus power system. There, the method's assessment is done by comparing total operation costs, total losses and nodal prices of the actual systems, with the ones obtained by the reduced models. The less there are differences between these two parameters families, the more the reduced models are truthful. This comparison's results reveal the adequacy of this power network reduction method.

A.2 Clustering features summary table

Table A.1 portrays an overview of the user-defined nodal parameters which are used inside each of the considered papers as clustering feature, to run the respective clustering algorithms.

	[6] Cao et al. (2018)	[7] Cotilla-Sanchez et al. (2013)	[10] Ferreira et al. (2011)	[11] Ferreira et al. (2010)	[16] Hong et al. (2002)	[20] Kiran et al. (2016)	[23] Klos et al. (2015)	[24] Koivisto et al. (2012)	[28] Sanchez-Garcia et al. (2014)	[30] Shayesteh et al. (2014)
LMP hourly patterns			X	X						
LMP snapshot	X									
ZMP hourly patterns						X				
Load levels					X					
Nodal influence on loop flows							X			
Load profiles								X		
Lines' average real power flows									X	
Lines' admittances									X	
ATC coefficients										X
System admittance matrix		X								

Table A.1: Clustering features summary table of references which deal with clustering algorithms for purposes different from *BAs* definition.

A.3 Clustering techniques summary table and descriptions

Table A.2 classifies the considered papers with respect to the adopted clustering algorithms. This latter is followed by a bulleted list, which contains two things for each clustering algorithm: its general description and the specific working processes which have been undertaken of it during its various applications, inside the papers included in this chapter.

	Fuzzy-c-means	Hierarchical Clustering	K-means	Spectral Clustering	Two-step	Hybrid Clustering Method
[6] Cao et al. (2018)				X		
[7] Cotilla-Sanchez et al. (2013)						X
[10] Ferreira et al. (2011)			X		X	
[11] Ferreira et al. (2010)			X		X	
[16] Hong et al. (2002)	X					
[20] Kiran et al. (2016)			X			
[23] Klos et al. (2015)		X				
[24] Koivisto et al. (2012)			X			
[28] Sanchez-Garcia et al. (2014)				X		
[30] Shayesteh et al. (2014)				X		

Table A.2: Clustering algorithms summary table of references which deal with clustering algorithms for purposes different from *BAs* definition.

- **Fuzzy-c-means:** This clustering algorithm has already been encountered inside papers dealing with *BAs* redefinition. Therefore, look at previous *Chapter 2* to find a general description of it.
  - **[16] Working process:** It has already been included inside chapter *Chapter 2*, in particular when dealing with the working process of reference [17, 27, 41]. Therefore, it is not repeated here.
- **Hierarchical Clustering:** This clustering algorithm has already been encountered inside papers dealing with *BAs* redefinition. Therefore, look at previous chapter *Chapter 2* to find a general description of it.

  - **[23] Working process:** The working process of classical hierarchical clustering algorithm has already been included inside chapter *Chapter 2*, in particular when dealing with the working process of references [4, 8, 18, 9, 25, 32, 36]. Therefore, it is not repeated here. The only difference of this paper resides in the external part of the aforementioned algorithm, which permits to use that clustering method for reducing power network's loop flows. This additional part is subsequently described. (a) Several Direct Current Optimal Power Flows (*DCOPFs*) are run in multiple scenarios, so as to identify transmission lines' power flows in different conditions of bids and offers. This allows to estimate the magnitude of each zone's average loop flows. (b) In fact, each of the aforementioned *DCOPFs* gives a different power network dispatching, related to its associated scenario. (c) Then for each of these dispatching the transmission lines' power flows are decomposed into their components, so as to classify the scenario's transitions among internal exchange, import/export, transit flow or loop flow. This is done through the Bialek's Proportional Sharing Principle, which constitutes the main assumption of this methodology, together with the lossless power flows analysis of *DCOPFs*. (d) The aforementioned decomposition gives to each transmission line the so-called "matrix of mutual power exchanges", which is a table containing the fractions of different transitions types that compose the line's power flux in that scenario. (e) At this point, through the average between these scenarios, the magnitude of each zone's average loop flows is found. In particular, it is also located the line with the highest average fraction of loop flows, which becomes the targeted line. The zone to which this latter belongs becomes the targeted zone, which has to be split due to its loop flows concentration. (f) Therefore, looking at this targeted-zone and taking the average of the targeted-line's matrices of mutual power exchanges (respectively associated to the various scenarios), it is stated the influence of each node of the targeted-zone on the targeted-line's loop flow. (g) Then these parameters, which represent something that recalls nodal *PTDFs*, are given as input to a classical hierarchical clustering algorithm, to divide in two the targeted-zone: one mainly importer and the other mainly exporter. In this way, the methodology comes to an end. Hence, the zone of the starting zonal configuration which contained the most of system's loop flows has correctly been split in two parts. In order to transform the zone's loop flows into transit ones, which are still unscheduled power flows but at the same time they are automatically handled by a zonal based market clearing.

- **K-means:** This clustering algorithm has already been encountered inside papers dealing with *BAs* redefinition. Therefore, look at *Chapter 2* to find a general description of it.

  - **[10, 11] Working process:** It has already been included inside *Chapter 2*, in particular when dealing with the working process of reference [25, 44]. Therefore, it is not repeated here. The only difference of this paper is that the

number of clusters, which unfortunately is a parameter that has to be user-defined beforehand even if it is always impossible to know its optimal value for the specific clustering application, is chosen by looking at clustering adequacy measures like Clustering Dispersion Indicator (*CDI*) or Mean Index Adequacy (*MIA*). The less these measures are, the better the partitioning is. Consequently, by measuring these indices for different number of zones it is seen that they significantly decrease up to thirty clusters in this case. After this value, they both remain mostly stable, which means that no relevant improvement of the partitioning is there ascertainable. Therefore, thirty clusters are then asked to the K-means clustering algorithm. Which is eventually elected as the best one, revealing better performance than the two-step one, and hence applied to the case study.

  - **[20, 24] Working process:** It has already been included inside *Chapter 2*, in particular when dealing with the working process of reference [25, 44]. Therefore, it is not repeated here. The only difference of this paper is that there is also a preliminary *PCA* process run before the K-means, so as to speed it up. Consequently, here the input database made up of the *ZMP* patterns of the twenty-four Indian *BAs* along a year is firstly put inside a *PCA*. This latter is a statistical tool which has the objective to replace the set of original input variables with a smaller set of artificial ones, able to retain most of the properties of the former variables. Then, once the *PCA* has ended and hence has issued the Principal Components (*PCs*) of the input database (four in this case), these last are given as new input to the following clustering algorithm. Which is here a traditional K-means clustering algorithm. This latter reveals to be markedly accelerated by having to deal with a reduced number of input data, namely the *PCs* instead of the whole original database. In this description it is worth pointing out that the *PCA* does not change the number of points inside the clustered database. It only permits not to consider them as independent variable, but as linear combination of *PCs*. So as to speed up the clustering algorithm that wants to be used to partition the aforementioned database.

- **Spectral Clustering:** By clustering it is meant the identification of groups in a dataset, which are created by merging highly correlated points. The correlation between two points depends on both the feature which has been chosen for the clustering, and the distance measure which has been adopted within the algorithm. A good clustering result must have: highly connected intra-clusters' points and weakly connected inter-clusters' ones. The spectral clustering algorithm tries to create that partitioning by using the Laplacian matrix, and particularly its eigenvalues and eigenvectors. The general idea of spectral clustering process is following.

  - **[6] Working process:** (a) A dataset made up of N points is given to the algorithm as input. Its data are called "vertices", and are the observations to cluster. (b) It is chosen the correlation criterion to be used among data, and then it is used to fill the NxN matrix called Laplacian. Hence inside this latter, each cell contains the correlation between a couple of dataset's points. Which

is actually called "edge" between those observations. This is why, also choosing the correlation criterion is actually indicated as choosing "edges' weight". (c) At this point it is typically suggested to normalize the Laplacian matrix, so as to reduce the computational effort of the clustering algorithm, but it is not compulsory. (d) After that the Laplacian matrix is reduced by using its eigenvalues and eigenvectors. So that it passes from a *NxN* matrix to a *Nxk'* one, where *k'* is the number of Laplacian matrix's eigenvectors. (e) Now the number of partitioning's clusters has to be user-defined. A suggestion for this choice can come by looking at eigengaps equation included inside reference [28]. Because, the k-th index which maximizes that formula indicates the clusters number of the approximated optimal database partitioning. (f) And moreover, by using Cheeger inequality reported inside reference [28] too, you also understand how this latter database partitioning is actually close to the actual optimal database partitioning. (g) Once user-defined the number of clusters, indicated as k, the previous reduced and maybe normalized Laplacian matrix is taken again and only considered for its first k columns. Thereby it becomes a *Nxk* matrix, where each line refers to a dataset's observation and can be regarded as its coordinates vector. (h) This group of vectors, representing observations' coordinates, is eventually given as input to any one of the typical clustering algorithm. This latter is usually a K-means clustering algorithm, which adopts the Ward's minimum variance criterion as distance metric and hence uses the classical multi-dimensional Euclidean distance between observations' vectors to create database's clusters. The working process of this last clustering algorithm is not included here, since already proposed inside *Chapter 2*.

– **[28] Working process:** The same of previous reference [6], apart from the last step, namely the "h" one. In this paper the final clustering algorithm, which does the final clustering after having received as input the observations' coordinates contained inside the reduced and maybe normalized Laplacian matrix, is a hierarchical clustering algorithm instead of K-means. Its capacity of eventually giving out the summary dendrogram is particularly useful in this methodology, because it includes in a sole graphic the whole power network's behavior with respect to the feature chosen for the clustering.

– **[30] Working process:** (a) Historical data and forecasts are used to identify possible loads, renewable productions, and other power network's features. These information are clustered into scenarios groups. In this methodology many scenarios are used to create the equivalent and reduced model of the power network, to give more accuracy to the model. (b) For each scenario of the aforementioned scenarios groups, the ATC coefficients between all its couples of system buses are calculated. (c) Average ATC coefficients are computed for each group of scenarios, so that for each of them an average ATC matrix is produced. (d) This latter is used, for each group of scenarios, to partition the power network into zones, whose number has to be defined by the user through his experience on the system. This zonal split is done for each group of scenarios, and it is produced by a spectral clustering algorithm. This algorithm is fed

with the aforementioned nodes' *ATC* coefficients and embeds K-means as the clustering algorithm. (e) Once these zonal configurations are declared, one per group of scenarios, essential and non-essential buses are identified for each zone of these partitionings. The former ones are located along the zones' borders or are user-defined, in case of the user wants to focus on the behavior of a specific group of nodes of the system. The latter ones, namely non-essential nodes, are all the others. (f) All the generators and loads within non-essential buses are moved to respective nearest essential buses. Then, non-essential nodes are eliminated. (g) In this way, it is obtained a network reduction for each zonal configuration respectively associated to one group of scenarios. (h) These reductions are the initially desired power network's reduced models. Hence they can be used to simulate system's operation. In that situation, the more the simulation requirements are close to the features of one of the scenarios groups, the more the associated reduced model will be suitable for the simulation and then will produce accurate outcomes.

- **Two-step:** This clustering algorithm differs from typical ones, usually divided in connectivity-based and centroid-based methods. It came out after having observed that traditional clustering algorithms are usually effective and accurate on small or medium datasets, especially the connectivity-based ones. But then, they do not scale up efficiently to very large datasets. Therefore, the two-step clustering algorithm tries to compensate this lack.

  - **[10, 11] Working process:** (a) It is firstly applied a quick sequential clustering to the input dataset. Large or not it does not matter, since this first step is a rough process able to handle large datasets. In this way the input points are split into many subclusters, organized into a tree of features. The number of subclusters is user-defined, and affects the quality of the overall clustering itself. The more subclusters are used, the more precise this first partitioning becomes. But meanwhile, having more subclusters also causes a longer second clustering step. And this is not good, because the two-step clustering algorithm is actually born to speed up the clustering of very large databases. Therefore, when choosing the number of subclusters a trade-off is necessary between first step's precision and second one's speed, since both of them affect the performance of the overall two-step clustering algorithm. (b) Once the first step is ended and thus the features tree is done, the second step starts. This latter takes the subclusters as input dataset, and clusters them through one of the classical clustering algorithm, which is often a hierarchical and agglomerative clustering process, like in this paper. (c) Therefore, according to the input required by this second classical clustering algorithm, the number of clusters of the final partitioning has to be chosen also in this second step. At the latest, the clustering algorithm has ended and the summary dendrogram has been issued, as made possible by hierarchical algorithms. Inside this second step this choice is much more relevant, because it compromises the final partitioning

of the primary dataset. (d) Inside the second step, as normally happens inside traditional clustering algorithms, it can be adopted an arbitrary distance metric. In this paper it is chosen "the closest neighbor" one.

- **Hybrid Clustering:** This new clustering algorithm is actually just the combination of two well-known clustering algorithm, which are respectively the K-means clustering and the genetic one. Detailed description of both of them is individually provided in previous *Chapter 2*. Hence, it is not presented here anymore. From the combination's point of view, it just has to be specified that the output of the preliminary phase, thus the K-means algorithm in this case, is given as input to the following process, namely the genetic algorithm here used. This latter's result represents the final partitioning of the whole hybrid clustering.

  - **[7] Working process:** (a) Look at the power network nodes and merge the leaf-nodes, namely those with just one connection, with their immediately connected neighbors. This permits to shrink the dimension of the problem, which hence becomes lighter to be solved, without changing its nature, since anyway these nodes would have been ineluctably merged to their immediate upstream neighbors by the clustering algorithm. (b) Generate an initial population of both random and K-means clustering solutions. They can be for instance fifty percent each. On the one hand, the random clustering solutions must only have the prerequisite of having a Cluster Size Index (*CSI*) score greater than 0.9, to produce big enough zones, because too small ones are not acceptable for many reasons. On the other hand, the K-means clustering solutions have been produced through a classical K-means algorithm fed with the system admittance matrix, by which the *EDs* needed for the clustering can be computed, and provided by the user with choice of the number of clusters and their centroids. (c) Choose the quality measures of the multi-attribute objective function, and their respective weighting factors too. These last coefficients are directly proportional, with a range that goes from 0 to 1, to the relative importance that the user wants to give to each of the aforementioned assessment criteria, with respect to the specific application. For instance, in this paper it is suggested to set cluster count index exponent equal to unity, since the number of clusters is usually a very strict request by the user, and *CSI*'s exponent greater than 0.8, since too small clusters are not acceptable for many reasons. (d) Run the genetic algorithm, using the aforementioned initial population of step "b" as input and considering the previous multi-attribute objective function as fitness function. (e) According to the genetic algorithm's process, a new population is produced at each step, by starting from the initial one and iteratively using genetic operations like crossover or mutation. (f) Calculate the Pareto set of solutions from all the clustering results produced by the genetic algorithm. Choose the best one as power network final partitioning.

A.4 Distance metrics summary table

Table A.3 distinguishes the papers considered inside this appendix according to the similarity metric used inside their clustering algorithms. Afterwards, a quick description of these distance measures is provided inside a bulleted list, which contains also the reference to where the specific metric has been used.

	Multidimensional Euclidean Distance	**Monodimensional Euclidean Distance**	**Fuzzy Membership**	**The Closest Neighbor**	**Electrical Distance**
[6] Cao et al. (2018)	X				
[7] Cotilla-Sanchez et al. (2013)					X
[10] Ferreira et al. (2011)	X			X	
[11] Ferreira et al. (2010)	X			X	
[16] Hong et al. (2002)			X		
[20] Kiran et al. (2016)	X				
[23] Klos et al. (2015)		X			
[24] Koivisto et al. (2012)	X				
[28] Sanchez-Garcia et al. (2014)	X				
[30] Shayesteh et al. (2014)	X				

Table A.3: Distance metrics summary table of references which deal with clustering algorithms for purposes different from *BAs* definition.

- **Multidimensional Euclidean Distance:**
  - **Definition:** $E_{ij} = \sqrt{\sum_{v=1}^{Ndim}(x_{vi} - x_{vj})^2}$
  - **Applications:**
    - **[6, 28, 30]:** Multidimensional Euclidean distance between the coordinates vectors of the database's observations. Represented by the lines of the reduced and maybe normalized Laplacian matrix.

    - **[10, 11]:** Multidimensional Euclidean distance between hourly *LMPs* patterns along a year.
    - **[20]:** Multidimensional Euclidean distance between the *ZMPs* patterns along a year, described as linear combination of four *PCs* coming out from the preliminary *PCA*.
    - **[24]:** Multidimensional Euclidean distance between the hourly Finnish load profiles along a year.

- **Monodimensional Euclidean Distance:**
  - **Definition:** $E_{ij} =\mid x_i - x_j \mid$
  - **Applications:**
    - **[23]:** Mono dimensional Euclidean distance between the nodes' influences on the loop flows of their targeted zone.

- **Fuzzy Membership:**
  - **Definition:** For a detailed description please look at *Chapter 2*.
  - **Applications:**
    - **[16]:** Fuzzy membership between aggregated demands at different hours. It is used inside the fuzzy-c-means which makes the clustering of transition periods. Thus for instance, if the aggregated demand at hour "x" of the day "y" is among the highest observed during the time window of the input database. Then the system's *LMPs* set of that hour is put inside the cluster of peak-load periods. And so on with other aggregated demands at different hours.

- **Electrical Distance:**
  - **Definition:** There are mainly two methods to define this distance metric, which are respectively a sensitivity method and an impedance one. The former is based on the sensitivity study between voltage and reactive power. The latter is based on examining the relationship between the voltage drop due to injecting a unit of current at one bus and withdrawing it at the receiving bus. This second version, where the larger the voltage drop the larger the electrical distance, is the one actually used inside these papers. It can be evaluated by starting from the system admittance matrix. That is why, in the previous table references which adopt this distance metric use it as clustering input.
  - **Applications:**
    - **[7]:** The electrical distance, computed through the impedance method, is here used as distance metric inside the preliminary K-means clustering of the proposed hybrid method.
    - **[35]:** The first aforementioned definition of electrical distance is here used to define newly born reserve zones. The couples of nodes with lower *EDs* are merged into the same cluster.

- **The Closest Neighbor:**
  - **Definition:** the distance between one cluster and another one is considered equal to the shortest distance from any member of the former to any member of the latter.
  - **Applications:**
    - **[10, 11]:** It is used as distance metric during the hierarchical agglomerative clustering algorithm used inside the second step, of the two-step algorithm here used. In this way, inside each step the two clusters with the two closest members placed at the smallest distance are merged.

A.5 Clustering algorithms' strengths and weaknesses

The following bulleted list presents for each of the clustering algorithms, which have been used inside the papers considered in this chapter, its strengths and weaknesses. Each of these last is also endowed with a reference to where the specific comment can be observed. In case the clustering algorithm has already been used by references described into the previous *Chapter 2*, the following pages contain only the additional pros and cons, which can specifically be recognized within the articles here outlined.

- **K-means:**
  - **Pros:**
    - **[10]:** Good clustering adequacy indices, namely *CDI* and *MIA*, on a large input dataset made up of *LMPs*, which could actually be the situation of *BAs* redefinition.
    - **[20]:** The insertion of a preliminary *PCA*, before the K-means clustering algorithm, effectively speeds up the overall partitioning process. Although it could not be necessary, due to its intrinsic speed even with large datasets.
- **Spectral Clustering:**
  - **Pros:**
    - **[6]:** It is able to handle very large dataset. Optimal thing for power networks analysis, actually associated with big databases.
  - **Cons:**
    - **[6]:** Obtaining the power network partitioning by using *LMPs* snapshots as input for the clustering algorithm risks to compromise the temporal stability of the final zonal configuration. This is an important feature for zonal configurations to be optimal, according to *CACM*'s guidelines in the field.
    - **[6]:** It requires the number of clusters as user-defined input. This is a drawback, since it is not possible to know it for the desired optimal zonal configuration in advance.

    - **[6]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.
    - **[6]:** When a K-means clustering algorithm is used in the second part of the spectral clustering, an important drawback of that centroid-based clustering algorithm is here inherited. This drawback is the strong dependence of the resulting clusters by the clusters centroids which are randomly or manually selected at the beginning of the clustering process. It is a con because: if these initial assignments are not well chosen, the algorithm only converges to a local optimum, and not to the global one, that would obviously be desired by the user. In other words, the outcome quality depends on a user's input. That is not acceptable in an optimization algorithm like this one for the find of an optimal *BAs* configuration. For these reasons, some measures would be needed to contain this drawback. But this means more complexity, namely a con for the clustering algorithm.

- **Two-step:**
  - **Pros:**
    - **[10]:** It is able to handle very large dataset. This is a positive aspect for power networks analysis, actually associated with big databases.
  - **Cons:**
    - **[10]:** It requires the number of clusters as user-defined input, and it is a drawback, since it is not possible to know it for the desired optimal zonal configuration in advance.
    - **[10]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. But this means more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations.
    - **[10]:** According to two clustering adequacy indices used inside this paper, namely *MIA* and *CDI*, this clustering approach does not reveal to be as efficient as a more traditional K-means algorithm. K-means is also used inside this reference, fed with the same group of data and eventually elected as the clustering approach adopted on the real case study, because of its better above mentioned clustering assessment criteria.

- **Hybrid Clustering:**
  - **Pros:**
    - **[7]:** Using the electrical distance as distance metric, it produces zonal configurations where the loop flows are minimized. This is more important in reserve zones definition rather than in BAs one. But anyway it is a pro of the resultant zonal configurations.
    - **[7]:** It is able to handle very large dataset. It includes two clustering algorithms, respectively a K-means clustering and a genetic algorithm, which are both able to do it. This is a positive aspect for power networks analysis, actually associated with big databases.
    - **[7]:** No check on the physical connection between nodes inside the same cluster is naturally enclosed. So that, even physically unfeasible zonal configuration can be created. Therefore, an additional control to prevent this situation has to be enclosed. This would usually be a con of the methodology, since it would mean more complexity, namely a drawback for the clustering algorithm, and the need to have a deeper knowledge of power network structure, in order to make nodal connections evaluations. Nevertheless, it is put on this side as point in favor of the algorithm. In fact, here the aforementioned additional check on nodes' physical connection, which moreover has to be included in many of the clustering algorithms used to redefine power network's BAs, is easily includable. In fact, inside the previously mentioned multi-attribute objective function, it is enough to include the Cluster Connectedness (CC) index with weighting factor equal 1. In this way, if one of the system nodes was not physically linked to any of the other nodes included in its own cluster, this parameter would become zero. Then, through multiplication with all the other assessment indices, the fitness function associated to this zonal configuration would become zero, thus rejecting this actually unfeasible system partitioning.
  - **Cons:**
    - **[7]:** The clustering method here described is not a finalized clustering algorithm, but a general partitioning approach which can be efficiently tailored on the specific application. This could seem to be a point in favour. Nevertheless, using the electrical distance as distance metric overall leads the clustering algorithm to define a final zonal configuration where buses within a zone are strongly connected, and buses between zones are weakly connected; from the electrical point of view. And this kind of partitioning reveals to be efficient when identifying closely-tied buses is advantageous. Like, for instance, when want to be defined new optimal reserve zones and hence ensuring deliverability between them is an important criterion. Indeed, this algorithm is perfectly able to minimize power network's loop flows, which would row against the aforementioned quote. But otherwise, the aforementioned zonal configurations are not suitable when the partitioning's aim depends on system's specific operating conditions and not

on network topology. Therefore, since defining an optimal *BAs* configuration is primarily focused on improving system's congestion management. This latter refers to a system's specific operating condition, namely the congestion of one or more of its transmission lines. This clustering algorithm is probably not well suitable for the *BAs* redefinition here analyzed, due to its poor efficiency in those applications where system dynamics are important, like the congestion management is.

* **[7]:** It requires the number of clusters as user-defined input, like the K-means and genetic algorithm included by it. This is a drawback, since it is not possible to know it for the desired optimal zonal configuration in advance.

Bibliography

[1] J.Bems, T.Kralik, J.Knapek and A.Kradeckaia, "Bidding zones reconfiguration - Current issues literature review, criterions and social welfare", *2016 2nd International Conference on Intelligent Green Building and Smart Grid (IGBSG)*, Prague, 2016, pp. 1-6, doi:10.1109/IGBSG.2016.7539427.

[2] E. Bjorndal, M.H. Bjorndal and V. Gribkovskaia, "A Nodal Pricing Model for the Nordic Electricity Market", *NHH Dept. of Business and Management Science Discussion Paper No. 2014/43*, pp. 1-33, `https://brage.bibsys.no/xmlui/bitstream/handle/11250/273333/4314.pdf?sequence=1`.

[3] C. Breuer, N. Seeger and A. Moser, "Determination of alternative bidding areas based on a full nodal pricing approach", *2013 IEEE Power & Energy Society General Meeting*, pp. 1-5, doi: 10.1109/PESMG.2013.6672466.

[4] B. Burstedde, "From nodal to zonal pricing: A bottom-up approach to the second-best", *2012 9th International Conference on the European Energy Market*, pp. 1-8, doi: 10.1109/EEM.2012.6254665.

[5] C. Breuer and A. Moser, "Optimized bidding area delimitations and their impact on electricity markets and congestion management", *11th International Conference on the European Energy Market (EEM14)*, Krakow, 2014, pp. 1-5, doi: 10.1109/EEM.2014.6861218.

[6] K. Cao, J. Metzdorf and S. Birbalta, "Incorporating Power Transmission Bottlenecks into Aggregated Energy System Models", *Sustainability*, vol. 10, no. 6, 2018, pp. 1-32, `https://www.mdpi.com/2071-1050/10/6/1916/pdf`.

[7] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, S. Blumsack and M. Patel, "Multi-Attribute Partitioning of Power Networks Based on Electrical Distance", *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4979-4987, Nov. 2013, doi: 10.1109/T-PWRS.2013.2263886.

[8] T. Felling and C. Weber, "Identifying price zones using nodal prices and supply & demand weighted nodes", *2016 IEEE International Energy Conference (ENERGYCON)*, Leuven, 2016, pp. 1-6, doi: 10.1109/ENERGYCON.2016.7514113.

[9] T. Felling and C. Weber, "Consistent and robust delimitation of price zones under uncertainty with an application to Central Western Europe", *Energy Economics*, vol.75, 2018, pp. 583-601, doi: 10.1016/j.eneco.2018.09.012.

[10] J. Ferreira, S. Ramos, Z. Vale and J. Soares, "A Data-Mining-Based Methodology for Transmission Expansion Planning", *IEEE Intelligent Systems*, vol. 26, no. 2, pp. 28-37, March-April 2011, doi: 10.1109/MIS.2011.4.

[11] J. Ferreira, S. Ramos, Z. Vale and J. Soares, "Zonal prices analysis supported by a data mining based methodology", *IEEE PES General Meeting*, 2010, pp. 1-8, doi: 10.1109/PES.2010.5590078.

[12] Gianfreda and L. Grossi, "Zonal price analysis of the Italian wholesale electricity market", *2009 6th International Conference on the European Energy Market*, pp. 1-6, doi: 10.1109/EEM.2009.5207198.

[13] Gianfreda, L. Grossi, "Forecasting Italian electricity zonal prices with exogenous variables", *Energy Economics*, vol. 34, no. 6, November 2012, pp. 2228-2239, doi: 10.1016/j.eneco.2012.06.024.

[14] V. Grimm, A. Martin, M. Weibelzahl and G. Zottl, "On the long run effects of market splitting: Why more price zones might decrease welfare", *Energy Policy*, vol. 94, pp. 453-467, 2016, doi: 10.1016/j.enpol.2015.11.010.

[15] V. Grimm, T. Kleinert, F. Liers, M. Schmidt and G. Zottl, "Optimal price zones of electricity markets: a mixed-integer multilevel model and global solution approaches", *Optimization Methods and Software*, 2017, pp. 1-31, doi: 10.1080/10556788.2017.1401069.

[16] Y.-Y. Hong and C.-Y. Hsiao, "Locational marginal price forecasting in deregulated electricity markets using artificial intelligence", *IEE Proceedings - Generation, Transmission and Distribution*, vol. 149, no. 5, September 2001, pp. 621-626, doi: 10.1049/ip-gtd:20020371.

[17] M. Imran and J. W. Bialek, "Effectiveness of zonal congestion management in the European electricity market", *2008 IEEE 2nd International Power and Energy Conference*, 2008, pp. 7-12, doi: 10.1109/PECON.2008.4762432.

[18] M. Jakubek, K. Wawrzyniak, M. Klos and M. Blachnik, "Are locational marginal prices a good heuristic to divide energy market into bidding zones?", *2015 12th International Conference on the European Energy Market (EEM)*, Lisbon, 2015, pp. 1-4, doi: 10.1109/EEM.2015.7216763.

[19] C.Q. Kang, Q.X. Chen, W.M. Lin, Y.R. Hong, Q. Xia, Z.X. Chen, Y. Wu, J.B. Xin, "Zonal marginal pricing approach based on sequential network partition and congestion contribution identification", *International Journal of Electrical Power & Energy Systems*, vol. 51, 2013, pp. 321-328, doi: 10.1016/j.ijepes.2013.02.033.

[20] D. Kiran, A. R. Abhyankar and B. K. Panigrahi, "Zonal price based clustering of bidding zones", *2016 IEEE 6th International Conference on Power Systems (ICPS)*, New Delhi, 2016, pp. 1-6, doi: 10.1109/ICPES.2016.7584201.

[21] D. Kiran, A. R. Abhyankar and B. K. Panigrahi, "Formation of Bidding Zones Based on Linear Bottleneck Games", *IEEE Systems Journal*, pp. 1-10, doi: 10.1109/JSYST.2017.2760886.

[22] M. Klos, K. Wawrzyniak, M. Jakubek and G. Orynczak, "The scheme of a novel methodology for zonal division based on power transfer distribution factors", *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, Dallas, TX, 2014, pp. 3598-3604, doi: 10.1109/IECON.2014.7049033.

[23] M. Klos, K. Wawrzyniak and M. Jakubek, "Decomposition of power flow used for optimizing zonal configurations of energy market", *2015 12th International Conference on the European Energy Market (EEM)*, Lisbon, 2015, pp. 1-5, doi: 10.1109/EEM.2015.7216779.

[24] M. Koivisto, P. Heine, I. Mellin and M. Lehtonen, "Clustering of Connection Points and Load Modeling in Distribution Systems", *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1255-1265, May 2013, doi: 10.1109/TPWRS.2012.2223240.

[25] N. Marinho, Y. Phulpin, D. Folliot and M. Hennebel, "Redispatch index for assessing bidding zone delineation", *IET Generation, Transmission & Distribution*, vol. 11, no. 17, pp. 4248-4255, 2017, doi: 10.1049/iet-gtd.2016.1334.

[26] G. Orynczak, M. Jakubek, K. Wawrzyniak and M. Klos, "Market coupling as the universal algorithm to assess zonal divisions", *11th International Conference on the European Energy Market (EEM14)*, Krakow, 2014, pp. 1-5, doi: 10.1109/EEM.2014.6861252.

[27] S. Jang, J. Kim, S. Lee and J. Park, "Zone Clustering LMP with Location information using an Improved Fuzzy C-Mean", *2007 International Conference on Intelligent Systems*, pp. 1-6, doi: 10.1109/ISAP.2007.4441605.

[28] R. J. Sanchez-Garcia et al., "Hierarchical Spectral Clustering of Power Grids", *IEEE Transactions on Power Systems*, vol. 29, no. 5, pp. 2229-2237, Sept. 2014, doi: 10.1109/TPWRS.2014.2306756.

[29] 29. M. Sarfati, M.R. Hesamzadeh and A. Canon, "Five indicators for assessing bidding area configurations in zonally-priced power markets", *2015 IEEE Power & Energy Society General Meeting*, Denver, CO, 2015, pp. 1-5, doi: 10.1109/PESGM.2015.7286517.

[30] E. Shayesteh, B.F. Hobbs, L. Soder and M. Amelin, "ATC-Based System Reduction for Planning Power Systems with Correlated Wind and Loads", *IEEE Transactions on Power Systems*, vol. 30, no. 1, pp. 429-438, Jan. 2015, doi: 10.1109/TPWRS.2014.2326615.

[31] R. Zhang, D. Wang and W.Y. Yun, "Power-Grid-Partitioning Model and its Tabu-Search-Embedded Algorithm for Zonal Pricing", *IFAC Proceedings Volumes*, vol. 41, no. 2, 2008, pp. 15927-15932, doi: 10.3182/20080706-5-KR-1001.0992.

[32] K. Van den Bergh, C. Wijsen, E. Delarue and W. D'haeseleer, "The impact of bidding zone configurations on electricity market outcomes", *2016 IEEE International Energy Conference (ENERGYCON)*, Leuven, 2016, pp. 1-6, doi: 10.1109/ENERGYCON.2016.7514031.

[33] T. Vaskovskaya, P. G. Thakurta and J. Bialek, "Identifying congestion zones with weighted decomposition of locational marginal prices", *2017 IEEE Manchester PowerTech*, Manchester, 2017, pp. 1-6, doi: 10.1109/PTC.2017.7981143.

[34] V. Volodin and T.A. Vaskovskaya, "Clustering approach for determination of congestion zones on nodal electricity markets in long term periods", *2015 IEEE Eindhoven PowerTech*, Eindhoven, 2015, pp. 1-6, doi: 10.1109/PTC.2015.7232560.

[35] Wang and K. W. Hedman, "Reserve zone determination based on statistical clustering methods", *2012 North American Power Symposium (NAPS)*, Champaign, IL, 2012, pp. 1-6, doi: 10.1109/NAPS.2012.6336318.

[36] K. Wawrzyniak, G. Orynczak, M. Klos, A. Goska and M. Jakubek, "Division of the energy market into zones in variable weather conditions using Locational Marginal Prices", *IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society*, Vienna, 2013, pp. 2027-2032, doi: 10.1109/IECON.2013.6699443.

[37] R. Weron, "Electricity price forecasting: A review of the state-of-the-art with a look into the future", *International Journal of Forecasting*, vol. 30, no. 4, October?December 2014, pp. 1030-1081, doi: 10.1016/j.ijforecast.2014.08.008.

[38] J. Nowotarski, R. Weron, "Recent advances in electricity price forecasting: A review of probabilistic forecasting", *Renewable and Sustainable Energy Reviews*, vol. 81, part 1, January 2018, pp. 1548-1568, doi: 10.1016/j.rser.2017.05.234.

[39] H. Yang, "A new clustering method for partitioning price zone in power market environment", *Periodica Polytechnica Ser. El. Eng.*, vol. 48, no. 3-4, 2004, pp. 183-195.

[40] H. Yang, R. Zhou and J. Liu, "A RBFN hierarchical clustering based network partitioning method for zonal pricing", *Proc. 2nd International Conference on Electrical and Electronics Engineering*, 2005, pp. 282-285, doi: 10.1109/ICEEE.2005.1529627.

[41] H. Yang and R. Zhou, "Monte Carlo Simulation Based Price Zone Partitioning Considering Market Uncertainty", *2006 International Conference on Probabilistic Methods Applied to Power Systems*, pp. 1-5, doi: 10.1109/PMAPS.2006.360219.

[42] Y.T. Yoon, J.R. Arce, K.K. Collison and M.D. Ilic, "Implementation of Cluster-based Congestion Management Systems", *Energy Laboratory Publication MIT EL 00-001 WP, Massachusetts Institute of Technology, 2000*, pp. 1-21, `http://web.mit.edu/energylab/www/pubs/el00-001wp.pdf`.

[43] P. Wang, Y. Ding and Y. Xiao, "Technique to evaluate nodal reliability indices and nodal prices of restructured power systems", *IEE Proceedings - Generation, Transmission and Distribution*, vol. 152, no. 3, May 2005, pp. 390-396, doi: 10.1049/ip-gtd:20041250.

[44] N. Yao, J. Wu, K. Liu and J. Cai, "Dynamic locational marginal prices based zonal division in large-scale regional electricity markets", *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, Guilin, 2016, pp. 2443-2448, doi: 10.1109/WCICA.2016.7578346.

[45] G. Chicco, "Electricity Customer Grouping and Load Profiling", *Slides of the Electrical Engineering Doctoral Course "Electrical load management, forecasting and control"*, Turin, 2018.

[46] G. W. Milligan and M.C. Cooper, "An examination of procedures for determining the number of clusters in a data set", *Psychometrika*, vol. 50, 1985, pp. 159-179.

[47] R. Mojena, "Hierarchical grouping methods and stopping rules: an evaluation", *Computer Journal*, vol. 20, pp. 359-363, 1977.

[48] T. Zhang, R. Ramakrishnan and M. Livny, "BIRCH: An Efficient Data Clustering Method for Very Large Databases", 1996, pp. 1-12.

[49] SPSS company, "The SPSS TwoStep Cluster Component - A scalable component enabling more efficient customer segmentation", *White paper - technical report*, U.S.A., 2001, pp. 1-9.

[50] THEMA consulting group, "Loop flows - Final advice", *THEMA Report 2013-36 for the European Commission*, October 2013, pp. 1-62.

[51] P. Luickx and A. Marien, "Principles of Flowbased Market Coupling", *CREG Workshop*, Brussel, 16 June 2014, pp. 1-50.

[52] Agency for the Cooperation of Energy Regulators, "Framework guidelines on capacity allocation and congestion management for electricity (FG-2011-E-002)", pp. 1-15, `https://www.acer.europa.eu/en/Electricity/FG_and_network_codes/Electricity`.

[53] R. D. Christie, B. F. Wollenberg, I. Wangensteen, "Transmission Management in the Deregulated Environment", *Proceedings of the IEEE*, vol. 88, no. 2, February 2000, pp. 170-195, doi: 10.1109/5.823997

[54] J. Egerer, J. Weibezahn, H. Hermann, "Two price zones for the German electricity market ? Market implications and distributional effects", *Energy Economics*, vol. 59, pp. 365-381, `https://doi.org/10.1016/j.eneco.2016.08.002`.

[55] Nordic transmission system operators Statnett SF, Svenska kraftnat, Fingrid Oy, Energinet.dk and the Baltic transmission system operators Elering, Litgrid and Augstsprieguma tikls (AST), `https://www.nordpoolgroup.com`.

[56] A. Rogin, "Computational framework for evaluating the impact of power-to-gas technology on European transmission system with large penetration of renewable sources", *Master thesis*, Politecnico di Torino, 2018.

[57] E. Bompard, "Market power analysis in the electricity market", *Slides of the Electrical Engineering Master of Science Course "Economia e gestione competitiva dei sistemi elettrici"*, Politecnico di Torino, 2018.

[58] R. Almeida, "Local Economic Structure and Growth", *World Bank Research Department*, 21 September 2006, pp. 1-33.

[59] F. Cingano, F. Schivardi, "Identifying the sources of local productivity growth", *Journal of the European Economic Association*, vol. 2, no. 4, 2004, pp. 720-742, doi: 10.1162/1542476041423322.

[60] G. Chicco, "Overview and performance assessment of the clustering methods for electrical load pattern grouping", *Energy*, vol. 42, no. 1, June 2012, pp. 68-80, doi: 10.1016/j.energy.2011.12.031.

[61] G. Chicco, "Distribution system optimization", *Slides of the Electrical Engineering Master's degree Course "Electric power distribution and utilization"*, Turin, 2018.