TESI DI LAUREA MAGISTRALE

SVILUPPO DI UN MODELLO DI AGGREGAZIONE PER LE RETI DI TELERISCALDAMENTO

RELATORI:

Ing. Vittorio Verda

Ing. Elisa Guelpa

CANDIDATO:

Nicola Barile

Sommario

ABSTRACT 5
1 INTRODUZIONE 6
1.1 TELERISCALDAMENTO E VANTAGGI AD ESSO CONNESSI 6
1.2 MODELLI DI SIMULAZIONE PER LE RETI DI TRASPORTO E DISTRIBUZIONE 8
1.2.1 NECESSITA' DEI MODELLI DI SIMULAZIONE 8
1.2.2 BREVE PANORAMICA DEI MODELLI DI SIMULAZIONE 9
1.2.3 NECESSITA' DI AGGREGAZIONE PER I MODELLI FISICI. 10
2 STUDIO MONODIMENSIONALE DELLE RETI DI TRASPORTO E DISTRIBUZIONE. 11
2.1 DESCRIZIONE TOPOLOGICA DI UNA RETE 11
2.2 MODELLO FLUIDODINAMICO 12
2.2.1 CONSERVAZIONE DELLA QUANTITA' DI MOTO 12
2.2.2 CONSERVAZIONE DELLA MASSA 13
2.2.3 SOLUZIONE DEL PROBLEMA FLUIDODINAMICO IN CONDIZIONI STAZIONARIE 14
2.2.4 SOLUZIONE DEL PROBLEMA FLUIDODINAMICO IN CONDIZIONI TRANSITORIE 19
2.3 MODELLO TERMICO 23
2.3.1 SOLUZIONE DEL PROBLEMA TERMICO IN CONDIZIONI STAZIONARIE 24
2.3.2 SOLUZIONE DEL PROBLEMA TERMICO IN CONDIZIONI TRANSITORIE 27
3 RIDUZIONE DEL COSTO COMPUTAZIONALE 29
3.1 CONFRONTO MODELLI DI AGGREGAZIONE 30
3.1.1 PASSAGGIO DA UNA STRUTTURA AD ALBERO AD UNA STRUTTURA LINEARE 31
3.1.2 RIMOZIONE DEI NODI 32
3.1.3 STRUTTURA DELLE RETI AGGREGATE 33
3.2 CENNI SUL MODELLO DI AGGREGAZIONE TEDESCO 35
3.2.1 LEGGI FONDAMENTALI 35
3.2.2 SEMPLIFICAZIONE DELLE DIVERSE STRUTTURE 37
3.3 MODELLO DI AGGREGAZIONE DANESE 41
3.3.1 SEMPLIFICAZIONE DELLE RAMIFICAZIONI 42
3.3.2 COMPATTAZIONE 49
3.3.3 RIASSUNTO PROCEDURA 52

4 APPLICAZIONE ALLA RETE BCT408 54
4.1 PROBLEMI INCONTRATI E SOLUZIONI ADOTTATE 56
4.1.1 VALORI DELLE DENSITÀ E DEL CALORE SPECIFICO 56
4.1.2 PRESENZA DI RAMI CON PORTATA SEMPRE NULLA 57
4.1.3 PRESENZA DI RAMI CON PORTATA NULLA NEL TRANSITORIO 59
4.2 SIMULAZIONI IN CONDIZIONI STAZIONARIE 61
4.2.1 SIMULAZIONE SULLA RETE ORIGINALE 62
4.2.3 SEMPLIFICAZIONE DELLA RETE ORIGINALE 66
4.2.4 SIMULAZIONI SULLE RETI SEMPLIFICATE E CONFRONTO DEI RISULTATI 67
4.3 SIMULAZIONI IN CONDIZIONI TRANSITORIE 1 71
4.3.1 SIMULAZIONE SULLA RETE ORIGINALE 72
4.3.2 SIMULAZIONI SULLE RETI EQUIVALENTI 74
4.4 SIMULAZIONI IN CONDIZIONI TRANSITORIE 2 79
4.4.1 SIMULAZIONE SULLA RETE ORIGINALE 80
4.4.2 SIMULAZIONI SULLE RETI EQUIVALENTI 83
5 CONCLUSIONI E SVILUPPI FUTURI 87

ABSTRACT

Il teleriscaldamento è una tecnologia che, se utilizzata in modo corretto su un'area urbana, permette di conseguire un importante risparmio di energia primaria.
Per massimizzare questo risparmio è necessario ottimizzare le condizioni di funzionamento degli impianti di generazione e delle reti di trasporto e di distribuzione del fluido termovettore. Si rende quindi necessario l'utilizzo di modelli di simulazione termo-fluidodinamici che calcolano la distribuzione delle portate e delle temperature nelle reti di teleriscaldamento.
I modelli di simulazione utilizzati si dividono in modelli fisici che necessitano di una descrizione topologica e fisica della rete e modelli statistici che necessitano di un gran numero di dati sperimentali .Tra i due, i più utilizzati sono i primi in quanto è relativamente semplice cambiare o aggiungere componenti alla rete esistente.
I modelli fisici, però, tendono ad essere costosi dal punto di vista computazionale; si è di conseguenza sviluppata la necessità di una semplificazione di questi modelli.
In questa tesi è presentato il modello di aggregazione danese delle reti di teleriscaldamento nel quale una rete complessa viene sostituita con un rete equivalente composta da pochi rami. Tale modello è stato sviluppato in Danimarca a seguito dell'avvio del progetto di ricerca " Equivalent models of district heating systems " nel 1996.

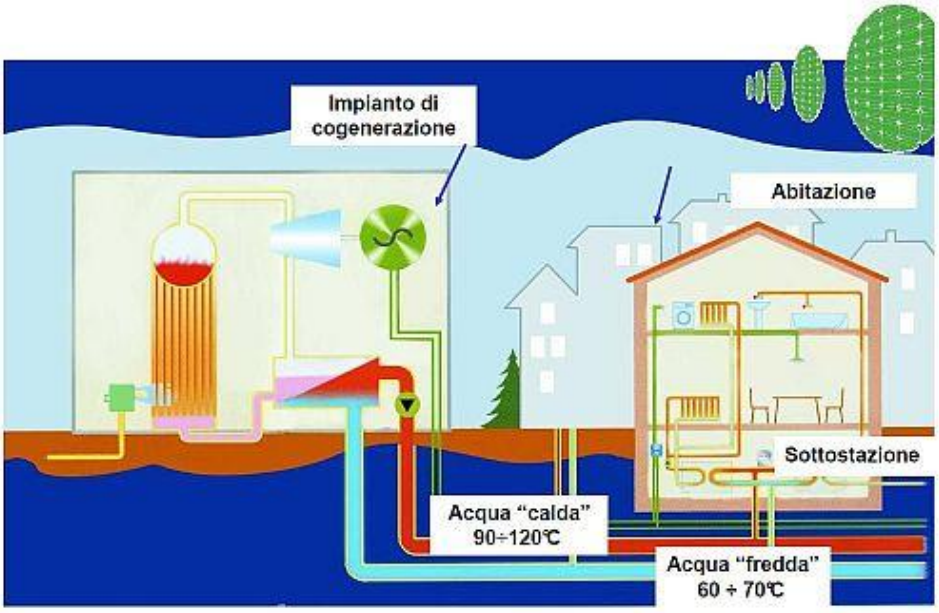
Poiché questo modello di semplificazione non riesce ad eliminare i loop, in prima istanza il modello di compattazione è stato applicato alla rete di distribuzione BCT 408 che non presenta circuiti chiusi. Nonostante il modello sia stato sviluppato per uno studio termo-fluidodinamico delle reti di trasporto e distribuzione per il teleriscaldamento in condizioni stazionarie, le reti equivalenti generate hanno riprodotto il comportamento della rete originale anche in condizioni transitorie.

Infine, mediante l'eliminazione del ramo con portata minore per ogni loop, il modello è stato applicato alla rete BCT414 presentante 4 circuiti chiusi, ottenendo tuttavia dei risultati positivi soltanto per il caso stazionario.

1 INTRODUZIONE

1.1 TELERISCALDAMENTO E VANTAGGI AD ESSO CONNESSI

Il teleriscaldamento è un sistema di riscaldamento a distanza composto da una centrale di generazione, che può essere termica, geotermica o cogenerativa e da una rete che porta l'acqua calda prodotta da questa alle abitazioni da riscaldare.
La rete può a sua volta essere suddivisa in rete di trasporto e rete di distribuzione.
La rete di trasporto è caratterizzata da diametri dell'ordine di 1-2 metri e collega la centrale termica con i baricentri : nodi che connettono la rete di trasporto con le reti distribuzione.
Le reti di distribuzione sono caratterizzate da diametri molto minori ed hanno il compito di portare il fluido termovettore dai baricentri alle sottostazioni situate all'interno degli edifici da servire.
Queste non sono altro che degli scambiatori di calore in cui l'acqua della rete di distribuzione cede potenza termica all'acqua del circuito secondario dell'edificio servito.
Il fluido caldo del secondario entra nelle abitazioni e quindi nei terminali al loro interno , cede il calore assorbito nella sottostazione ed in seguito fa ritorno nella sottostazione stessa.
Il fluido freddo del primario, invece, ritorna nella centrale di generazione dove viene riscaldato per poi ripetere il ciclo.

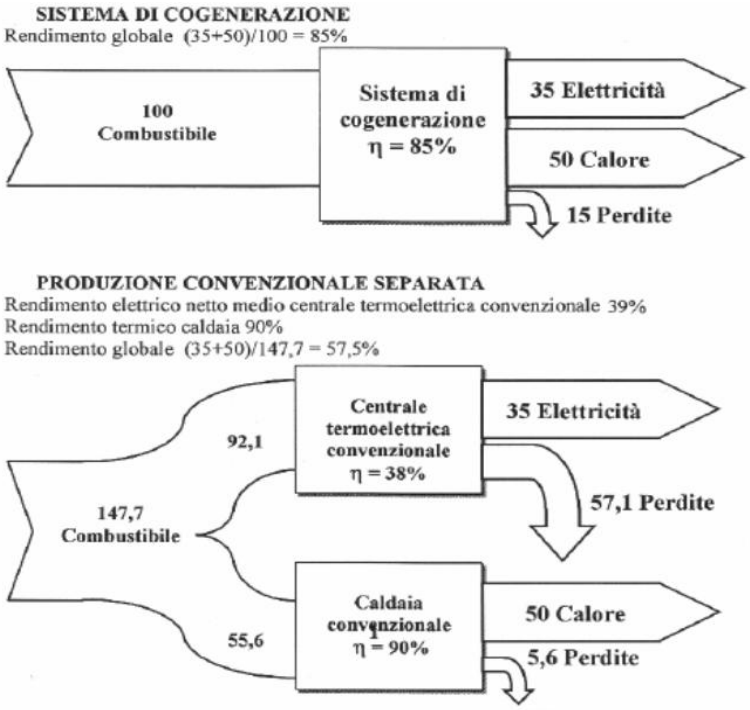


(Figura 1.1)

L'affidabilità di questo servizio è elevatissima ed è possibile applicare il sistema ad intere aree urbane rendendolo un vero e proprio servizio pubblico [1].

I vantaggi energetici ed ambientali legati al teleriscaldamento sono evidenti nel caso in cui l'energia termica viene prodotta da una centrale cogenerativa.
Queste centrali producono sia energia elettrica che energia termica ed hanno un' efficienza globale maggiore rispetto a quella delle centrali termoelettriche convenzionali. Le centrali a cogenerazione, infatti, riescono a convertire energia che altrimenti sarebbe dispersa (ad esempio l'energia termica contenuta nei fumi) in energia termica utilizzabile.

Consideriamo ad esempio la seguente immagine



(Figura 1.2)

In alto abbiamo una rappresentazione dei flussi energetici di una centrale cogenerativa mentre in basso troviamo una rappresentazione dei flussi energetici nel caso di produzione separata.
Nell'esempio in figura 1.2 si ha una richiesta di energia elettrica pari a 35 unità ed una richiesta di energia termica pari a 50 unità.
Considerando il caso della produzione separata con rendimento elettrico η_{el}=0.38 e rendimento termico η_{th}=0.9 , per produrre la quantità richiesta di energia termica E_{th} ed elettrica E_{el} bisogna utilizzare una quantità di energia primaria E_{pr} pari a:

$$E_{pr} = \frac{E_{th}}{\eta_{\text{th}}} + \frac{E_{el}}{\eta_{\text{el}}} = (55.6 + 92.1)\ unità = 144.7\ unità \qquad (1.1)$$

Se consideriamo invece una centrale a cogenerazione con un rendimento elettrico η_{el}=0.35 ed un rendimento termico η_{th}=0.5, per produrre la quantità richiesta di energia termica E_{th} ed elettrica E_{el} bisogna utilizzare una quantità di energia primaria E_{pr} pari a:

$$E_{pr} = \frac{E_{th}+E_{el}}{\eta_{\text{th}}+\eta_{\text{el}}} = 100\ unità \qquad (1.2)$$

Quindi mediante l' impianto di cogenerazione, a parità di energia prodotta si utilizza meno energia primaria e conseguentemente meno combustibile. Ciò naturalmente comporta vantaggi sia economici che ambientali. Bisogna inoltre considerare che i potenziali benefici ambientali non scaturiscono esclusivamente dalla riduzione della quantità di combustibile utilizzata: infatti, centralizzando la produzione di energia termica, diventa possibile utilizzare tecnologie che sfruttano fonti rinnovabili e sistemi volti all'abbattimento degli inquinanti .

1.2 MODELLI DI SIMULAZIONE PER LE RETI DI TRASPORTO E DISTRIBUZIONE

1.2.1 NECESSITA' DEI MODELLI DI SIMULAZIONE

Per aumentare l'efficienza del teleriscaldamento sono stati effettuati diversi studi sull'ottimizzazione delle condizioni di funzionamento a seconda dei carichi richiesti dalle utenze.
Diverse ricerche si concentrano sull'analisi della temperatura ottimale del fluido in uscita dalla centrale [2].
Durante il trasporto e la distribuzione si cerca di minimizzare le dissipazioni di calore attraverso la rete. Siccome il driver dello scambio termico è la differenza di temperatura tra i corpi coinvolti (in questo caso fluido termovettore e il terreno), è possibile diminuire le dispersioni abbassando la temperatura del fluido in uscita dalla centrale. In ogni caso,per far sì che nelle sottostazioni l'acqua del circuito primario riesca a riscaldare quella del secondario, la temperatura in ingresso nella rete primaria non può essere troppo bassa.
Inoltre bisogna considerare che, a parità di carico richiesto dalle utenze, al diminuire della temperatura in ingresso nella rete la portata massica aumenta e con essa anche la potenza da fornire alle pompe. Di conseguenza è necessario trovare una temperatura di ingresso nella rete primaria tale per cui la sommatoria tra le dispersioni termiche attraverso i condotti e la potenza da fornire alle pompe sia minima.

Altri studi si concentrano sulla massimizzazione della quantità di energia prodotta dagli impianti a cogenerazione o a fonti rinnovabili mediante la modifica della richiesta termica delle utenze[3].
Nell'area mediterranea, infatti, poichè durante la notte i sistemi di riscaldamento vengono spenti, nelle prime ore diurne si ha un picco di richiesta termica e di conseguenza è necessario utilizzare caldaie di integrazione. Per limitare l'uso di queste caldaie è possibile cercare di limare i picchi di richiesta termica durante la mattina utilizzando degli accumuli termici o modificando le richieste termiche delle utenze mediante appropriate politiche. In questo modo si sposterebbe una quota di domanda di energia da orari in cui gli impianti a cogenerazione e a fonti rinnovabili sono sfruttati a pieno, ad orari in cui la domanda è minore e gli impianti cogenerativi e a fonti rinnovabili non sfruttano la loro massima potenza.

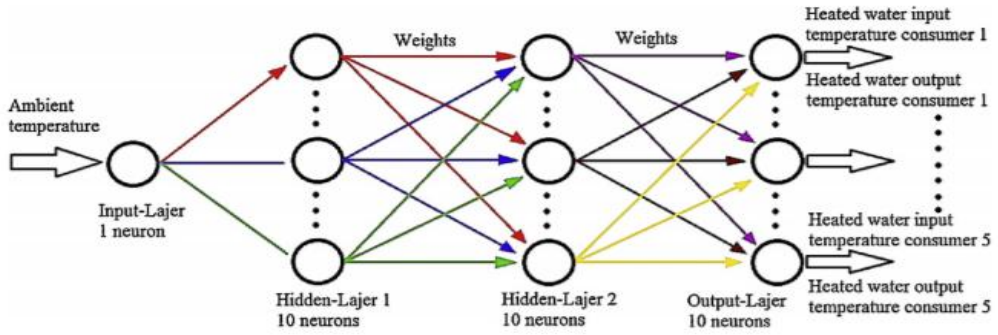
Infine vi sono delle ricerche volte ad ottimizzare la regolazione delle pompe in funzione delle portate. L'energia consumata dalle pompe non è infatti trascurabile: ad esempio nella rete di Torino, per garantire la pressione necessaria, bisogna fornire al fluido termovettore circa 6 MW di potenza meccanica, corrispondente al 2% dell'energia primaria in condizioni di carico massimo e al 6-8% nel caso di carico ridotto durante la notte [3].

In tutti questi studi appena elencati bisogna tener conto della complessità delle reti di trasporto e distribuzione in cui si hanno miscelazioni tra portate a temperature diverse, transitori termici, dispersioni attraverso i condotti della rete e non simultaneità tra la potenza prodotta dalle centrali di generazione e la potenza richiesta dalle utenze. Per tener conto di tutto ciò è necessario utilizzare dei modelli termo-fluidodinamici che descrivano il comportamento delle reti.

1.2.2 BREVE PANORAMICA DEI MODELLI DI SIMULAZIONE

I modelli di simulazione si dividono in due tipologie: modelli black box e modelli fisici

I modelli black box descrivono le reti di teleriscaldamento mediante delle funzioni di traferimento o delle ANN (Artificial Neural Network).
Un ANN è un sistema in cui un gran numero di elementi in relazione fra loro reagiscono ad uno o più input producendo uno o più output.
Questi elementi sono dei modelli matematici che imitano la struttura e le funzionalità di una rete biologica di neuroni.
Qui di seguito troviamo una figura che schematizza la struttura ANN per una rete di teleriscaldamento



(Figura 1.3)

La caratteristica principale di un ANN è la capacità di sviluppare mediante il " learning process " una relazione che collega gli input agli output. Il *learning process* non è altro che una procedura durante la quale vengono forniti dei valori di input e in base a questi vengono dati dei pesi alle funzioni di interconnessione tra i neuroni. Ciò produce degli output all'uscita del modello ANN (come ad esempio delle temperature) e questi vengono confrontati con i valori reali raccolti sul campo. Se le soluzioni fornite in output si discostano dal valore reale di una quantità inferiore alla tolleranza prefissata, la rete memorizza la distribuzione dei pesi delle relazioni di interconnessione tra i neuroni; altrimenti la procedura viene ripetuta. Infine Si ripete l'intero processo cambiando gli input e ricalibrando la distribuzione dei pesi tenendo in considerazione anche i risultati ottenuti con i vecchi input.
I modelli black box non necessitano di alcuna informazione sulla rete di distribuzione, ma per la calibrazione di questi bisogna avere una grande quantità di dati sperimentali.

I modelli fisici, invece, per trovare le temperature e le portate si basano sulla risoluzione della fisica del sistema in esame.
Questi modelli non necessitano di un gran numero di dati sperimentali, ma occorre conoscere le caratteristiche topologiche e fisiche della rete come ad esempio lunghezze, diametri e fattore di attrito dei vari condotti.
il lavoro svolto in questa tesi riguarda i modelli fisici in quanto i dati necessari per l'applicazione di questi sono spesso disponibili ed è relativamente semplice cambiare o aggiungere componenti al sistema esistente.

1.2.3 NECESSITA' DI AGGREGAZIONE PER I MODELLI FISICI.

Come esposto nel paragrafo precedente, i modelli fisici richiedono una descrizione approfondita della rete e di conseguenza hanno un costo computazionale più elevato, ovvero il calcolatore può impiegare molto tempo a trovare la soluzione del problema termodinamico e fluidodinamico.
Di conseguenza, soprattutto per le reti estese, è nata la necessità di semplificare la descrizione fisica senza perdere la relazione del modello con la rete reale [4].

In questa tesi si vuole quindi applicare uno dei modelli di aggregazione presenti : il modello danese.
Il lavoro riportato in questa tesi è quindi suddivisibile in quattro fasi:

- nella prima fase sono state effettuate le simulazioni relative ad una rete di distribuzione reale;
- nella seconda fase la rete originale è stata aggregata generando diverse reti equivalenti;
- nella terza fase sono state effettuate le simulazioni relative alle reti di distribuzione equivalenti;
- nella quarta fase si sono confrontati i risultati ottenuti dalle simulazioni sulla rete originale con quelli ottenuti dalle simulazioni sule reti equivalenti.

2 STUDIO MONODIMENSIONALE DELLE RETI DI TRASPORTO E DISTRIBUZIONE.

2.1 DESCRIZIONE TOPOLOGICA DI UNA RETE

In una rete di trasporto e distribuzione dell'acqua per il teleriscaldamento ci sono diversi componenti tra cui condotti, pompe,valvole e sottostazioni.
Per descrivere come i vari elementi di questo sistema sono collegati tra loro ci si avvale della teoria dei grafi che descrive la rete come un insieme di nodi e rami.

- Un nodo rappresenta il punto di connessione tra due elementi.
- Un ramo è una porzione della rete delimitata tra due nodi e spesso corrisponde ai condotti.

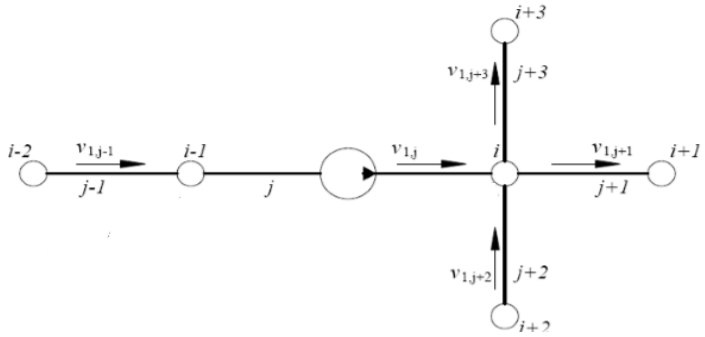
Quindi il problema della descrizione topologica della rete si riduce ad un problema di rappresentazione delle connessioni tra i nodi e i rami che viene risolto utilizzando una matrice detta "matrice di incidenza" che indicheremo con **A**.
La matrice di incidenza è costituita da:

- tante righe quanti sono i nodi della rete
- tante colonne quanti sono i rami della rete

Ogni elemento $a_{i,j}$ appartenente ad A è pari a

- +1 se il ramo j esce dal nodo i
- -1 se il ramo j entra nel nodo i
- 0 se il ramo j non è connesso al il nodo i

Quindi nota la rete e stabilito un verso arbitrario di scorrimento del fluido in ogni ramo, è possibile ricavare la matrice di incidenza; qui di seguito troviamo un esempio.



(Figura 2.1)

A =	j-1	j	j+1	j+2	j+3
i-2	+1	0	0	0	0
i-1	-1	+1	0	0	0
i	0	-1	+1	-1	+1
i+1	0	0	-1	0	0
i+2	0	0	0	+1	0
i+3	0	0	0	0	-1

Nel modello fluidodinamico ed in quello termico che saranno presentati nelle pagine seguenti, le proprietà di stato come pressione e temperatura sono definite nei nodi, mentre velocità e portate del fluido sono definite nei rami.

2.2 MODELLO FLUIDODINAMICO

Il modello fluidodinamico si basa su due equazioni: la conservazione della quantità di moto e la conservazione della massa

2.2.1 CONSERVAZIONE DELLA QUANTITA' DI MOTO

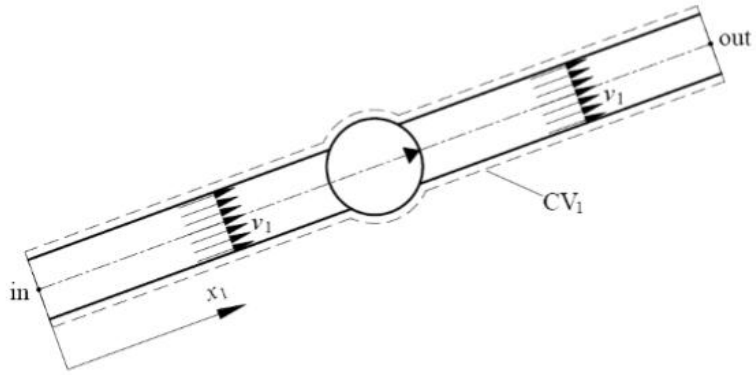
La conservazione della quantità di moto lungo la direzione x1 può essere espressa con

$$\rho * \frac{\partial v_1}{\partial t} + \rho * v_1 * \frac{\partial v_1}{\partial x_1} = -\frac{\partial p}{\partial x_1} - F_{FRICT} + F_1 \qquad (2.1)$$

Dove v indica la velocità in m/s, p la pressione in Pa, ρ la densità in kg/m^3, il termine F_{FRICT} tiene conto delle resistenze fluidodinamiche distribuite ed il termine F_1 considera l'accelerazione gravitazionale, le resistenze fluidodinamiche concentrate ed il contributo delle pompe

$$F_1 = \rho * g_{x1} - F_{LOCAL} + F_{PUMP} \qquad (2.2)$$

Consideriamo ora il seguente condotto



(Figura 2.2)

Applicando le equazioni 2.1 e 2.2 ed integrando sul volume di controllo CV_1 otteniamo

$$\rho * \frac{\partial v_1}{\partial t} * S * L + \rho * \left[\frac{{v_{1,out}}^2 - {v_{1,in}}^2}{2}\right] * S =$$

$$= (p_{in} - p_{out}) * S - \rho * g * (z_{out} - z_{in}) * S - \Delta P_{FRICT} * S - \Delta P_{LOCAL} * S + \Delta P_{PUMP} * S \qquad (2.3)$$

Dove S ed L sono rispettivamente la sezione interna e la lunghezza della tubazione.

Introduciamo ora la pressione totale P definita come

$$P = p + \rho * \frac{v^2}{2} + \rho * g * z \qquad (2.4)$$

Dalle equazioni 2.3 e 2.4 si ha:

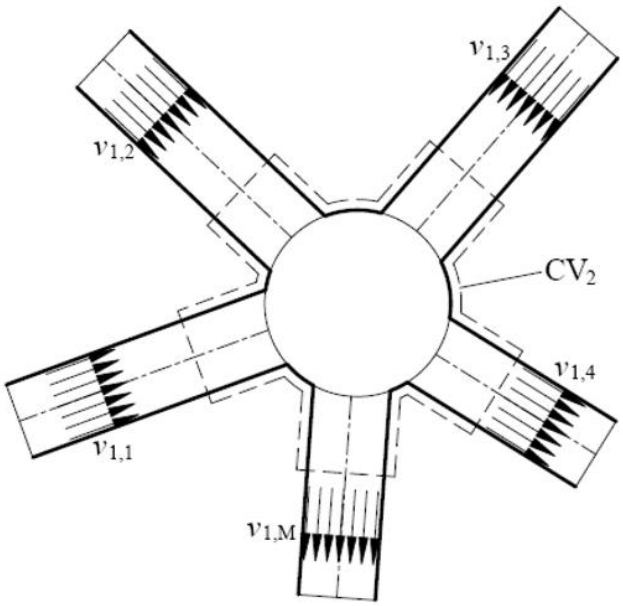
$$\rho * \frac{\partial v_1}{\partial t} * L + (P_{out} - P_{in}) = -\Delta P_{FRICT} - \Delta P_{LOCAL} + \Delta P_{PUMP} \qquad (2.5)$$

Dove

- $\Delta P_{FRICT} = \frac{1}{2} * f * \frac{L}{D_h} * \rho * {v_1}^2$ (2.6)
  Con f coefficiente di attrito determinato empiricamente e D_h diametro idraulico
- $\Delta P_{LOCAL} = \frac{1}{2} * \sum_k \beta_k * \rho * {v_1}^2$ (2.7)
- ΔP_{PUMP} è l'incremento di pressione fornito dalla pompa (2.8)

2.2.2 CONSERVAZIONE DELLA MASSA

Consideriamo un nodo connesso con più rami come nella figura sottostante



(Figura 2.3)

La conservazione della massa si esprime con

$$\frac{d}{dt}M + \sum_j \rho * v_{1,j} * S_j = 0 \qquad (2.9)$$

Dove il pedice j indica i rami ed M è la massa del fluido contenuta nel volume di controllo CV_2.
I termini della sommatoria sono positivi o negativi in base al verso di scorrimento del fluido assunto da noi in modo arbitrario: se il fluido percorrente il ramo entra nel nodo il segno è negativo, altrimenti è positivo.

Per tener conto di eventuali portate uscenti/entranti dalla/nella rete si aggiunge il termine G_{ext} [kg/s] che assume un valore positivo se la portata esce dalla rete e negativo se la portata entra nella stessa.

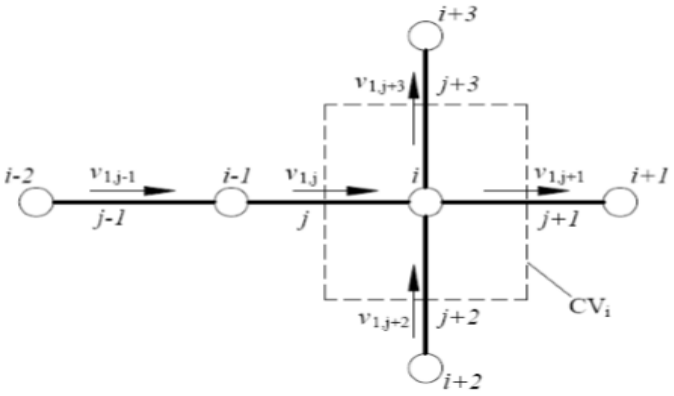
$$\frac{d}{dt}M + \sum_j \rho * v_{1,j} * S_j + G_{ext} = 0 \qquad (2.10)$$

2.2.3 SOLUZIONE DEL PROBLEMA FLUIDODINAMICO IN CONDIZIONI STAZIONARIE

Per la risoluzione del problema fluidodinamico come primo passo si applica l'equazione di conservazione della massa 2.10 a tutti i nodi e l'equazione di conservazione della quantità di moto 2.5 a tutti i rami, ottenendo cosi due sistemi di equazioni; successivamente si scrivono questi sistemi di equazioni in forma matriciale ottenendo due equazioni matriciali.
A causa della dipendenza delle conduttanze dalla portata, l'equazione di conservazione della quantità di moto non è lineare; pertanto, al fine di calcolare le portate e le pressioni, le due equazioni matriciali vengono combinate tra loro utilizzando l'algoritmo SIMPLE (Semi-Implicit Method for Pressure Linked Equations).

Conservazione della massa

Consideriamo un generico nodo i di una rete come quella in figura 2.4 ed un volume di controllo CV_i comprendente il nodo analizzato e la metà di tutti i rami ad esso connessi.



(Figura 2.4)

Applicando l'equazione 2.10 al volume di controllo ed annullando la derivata della massa nel tempo (stiamo studiando la rete in condizioni stazionarie) si ha

$$\sum_j \rho * v_{1,j} * S_j + G_{ext,i} = 0 \qquad (2.11)$$

Considerando che la portata nel ramo j può essere espressa con $G_j = \rho * v_{1,j} * S_j$ possiamo scrivere

$$\sum_j G_j + G_{ext,i} = 0 \qquad (2.12)$$

Scrivendo quindi l'equazione 2.12 per tutti i punti otteniamo un sistema di equazioni che può essere compattato nella seguente forma matriciale

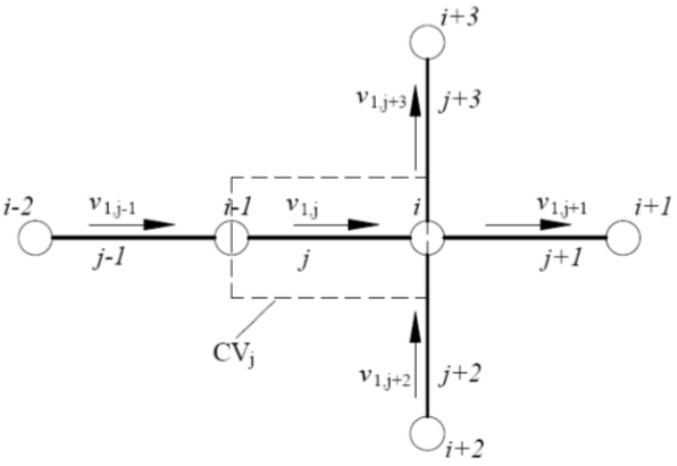
$$\boldsymbol{A} * \boldsymbol{G} + \boldsymbol{G_{ext}} = 0 \qquad (2.13)$$

Dove

- **A** è la matrice di incidenza di dimensioni n_N x n_R (numero nodi x numero rami)
- **G** è un vettore colonna lungo n_R in cui l'elemento j-esimo è la portata che scorre nel ramo j.
- $\mathbf{G_{ext}}$ è un vettore colonna lungo n_N in cui l'elemento i-esimo corrisponde alla portata uscente dal nodo i verso l'esterno (oppure entrante nel nodo i dall'esterno se negativa).
  Se non vi è alcuna portata estratta/immessa dal/nel nodo i allora l'elemento ha un valore nullo.

Conservazione della quantità di moto

Consideriamo un ramo generico j ed il relativo volume di controllo CV_j come in figura



(Figura 2.5)

Applicando le equazioni 2.5, 2.6 e 2.7 e annullando la derivata nel tempo (regime stazionario) si ha

$$(P_i - P_{i-1}) = -\frac{1}{2} * f_j * \frac{L_j}{D_{h,j}} * \rho * {v_{1,j}}^2 - \frac{1}{2} * \sum_k \beta_{k,j} * \rho * {v_{1,j}}^2 + \Delta P_{PUMP,j} \quad (2.14)$$

Esprimendo la portata massica con $G_j = \rho * v_{1,j} * S_j$ l'espressione 2.14 può essere scritta come

$$(P_{i-1} - P_i) = \frac{1}{2} * \frac{G_j^2}{\rho * S_j^2} * \left(f_j * \frac{L_j}{D_{h,j}} + \sum_k \beta_{k,j} \right) - \Delta P_{PUMP,j} \quad (2.15)$$

L'equazione 2.15 può essere espressa con

$$(P_{i-1} - P_i) = R_j * G_j - \Delta P_{PUMP,j} \quad (2.16)$$

Dove R_j è la resistenza idraulica

$$R_j = \frac{1}{2} * \frac{G_j}{\rho * S_j^2} * \left(f_j * \frac{L_j}{D_{h,j}} + \sum_k \beta_{k,j} \right) \quad (2.17)$$

Scrivendo l'equazione 2.16 per ogni ramo della rete otteniamo un sistema di equazioni che può essere compattato nella seguente forma matriciale

$$\boldsymbol{A^T * P = R * G - t} \quad (2.18)$$

Dove

- **R** è una matrice diagonale di dimensioni n_R x n_R in cui l'elemento R(j,j) è la resistenza idraulica del ramo j.
- **P** è un vettore colonna di dimensione n_N in cui l'elemento i-esimo è la pressione al nodo i.
- **t** è un vettore colonna di dimensione n_R in cui l'elemento j-esimo è la differenza di pressione a cavallo della pompa; se non vi sono pompe t(j)=0.

E' possibile riscrivere l'equazione 2.18 in modo da ottenere *formalmente* la portata nei rami

$$\boldsymbol{G = R^{-1} * A^T * P + R^{-1} * t \rightarrow G = Y * A^T * P + Y * t} \quad (2.19)$$

Dove $\mathbf{Y}=\mathbf{R}^{-1}$ è una matrice diagonale in cui l'elemento j-esimo della diagonale principale è la conduttanza fluidodinamica $1/R_j$

$$Y_j = \left(\frac{1}{2} * \frac{G_j}{\rho * S_j^2} * \left(f_j * \frac{L_j}{D_{h,j}} + \sum_k \beta_{k,j} \right) \right)^{-1} \quad (2.20)$$

E' importante notare che **R** dipende dalla portata nei vari rami e quindi non è possibile calcolare **Y** senza conoscere la portata; inoltre le pressioni non sono note. Di conseguenza l'equazione 2.19 esprime la portata solo formalmente.
Per calcolare quindi le pressioni nei vari nodi e le portate nei vari rami è necessario considerare sia l'equazione 2.19 che l'equazione 2.13.

Algoritmo SIMPLE

L'algoritmo SIMPLE è un " guess and correct method " [5] in cui si ipotizza un vettore di pressioni **P*** con il quale si calcolano le portate **G*** mediante il *fixed point method* (vedi appendice).
Con dei calcoli, che vedremo a breve, si trovano quindi i vettori delle correzioni $\mathbf{G_{corr}}$ e $\mathbf{P_{corr}}$ da sommare alle portate contenute in **G*** e alle pressioni contenute in **P*** e successivamente si trovano i residui delle portate.
Si aggiornano in seguito i valori delle pressioni ipotizzate e delle potate e si ripete la procedura fino a quando il massimo in valore assoluto del vettore dei residui delle portate risulta minore di una certa tolleranza prestabilita.

Ipotizzando quindi **P***, mediante il fixed point alghoritm calcoliamo **G*** :

$$\boldsymbol{G^* = Y^* * A^T * P^* + Y^* * t} \quad (2.21)$$

Introduciamo ora le correzioni alle portate e alle pressioni

$$\boldsymbol{G_{corr} = G - G^*} \quad (2.22)$$

$$\boldsymbol{P_{corr} = P - P^*} \quad (2.23)$$

Dalle equazioni 2.22, 2.21 e 2.19 si ha

$$\boldsymbol{G_{corr} = G - G^* = Y * A^T * P + Y * t - (Y^* * A^T * P^* + Y^* * t)} \quad (2.24)$$

Approssimando ora **Y** con $\mathbf{Y^*}$, otteniamo la relazione tra il vettore delle correzioni delle portate e quello delle correzioni delle pressioni

$$\boldsymbol{G_{corr} = Y^* * A^T * (P - P^*) = Y^* * A^T * P_{corr}} \quad (2.25)$$

Utilizzando la conservazione della massa e la definizione del vettore delle correzioni delle portate:

$$\begin{cases} \boldsymbol{A * G + G_{ext} = 0} \\ \boldsymbol{G = G^* + G_{corr}} \end{cases} \rightarrow \boldsymbol{A * (G^* + G_{corr}) + G_{ext} = 0;\ A * G_{corr} = -A * G^* - G_{ext}} \quad (2.26)$$

Dalle equazioni 2.25 e 2.26 otteniamo infine

$$\boldsymbol{A * Y^* * A^T * P_{corr} = -A * G^* - G_{ext}} \quad (2.27)$$

Introducendo ora i vettori **H** e **b** otteniamo l'equazione 2.30

$$\boldsymbol{H = A * Y^* * A^T} \quad (2.28)$$

$$\boldsymbol{b = -A * G^* - G_{ext}} \quad (2.29)$$

$$\boldsymbol{H * P_{corr} = b} \quad (2.30)$$

Mediante l'equazione 2.30 è possibile calcolare $\mathbf{P_{corr}}$ ed in seguito con l'equazione 2.25 si ottiene $\mathbf{G_{corr}}$.

In base alle caratteristiche della rete studiata, Il processo iterativo descritto finora può non essere convergente. Per evitare qualsiasi divergenza, durante l'aggiornamento delle pressioni e delle portate bisogna utilizzare i coefficienti α_p e α_G con valore compreso tra 0 e 1.

$$\boldsymbol{P_{new}} = \boldsymbol{P}^* + \alpha_p * \boldsymbol{P_{corr}} \qquad (2.31)$$

$$\boldsymbol{G_{new}} = \boldsymbol{G}^* + \alpha_G * \boldsymbol{G_{corr}} \qquad (2.32)$$

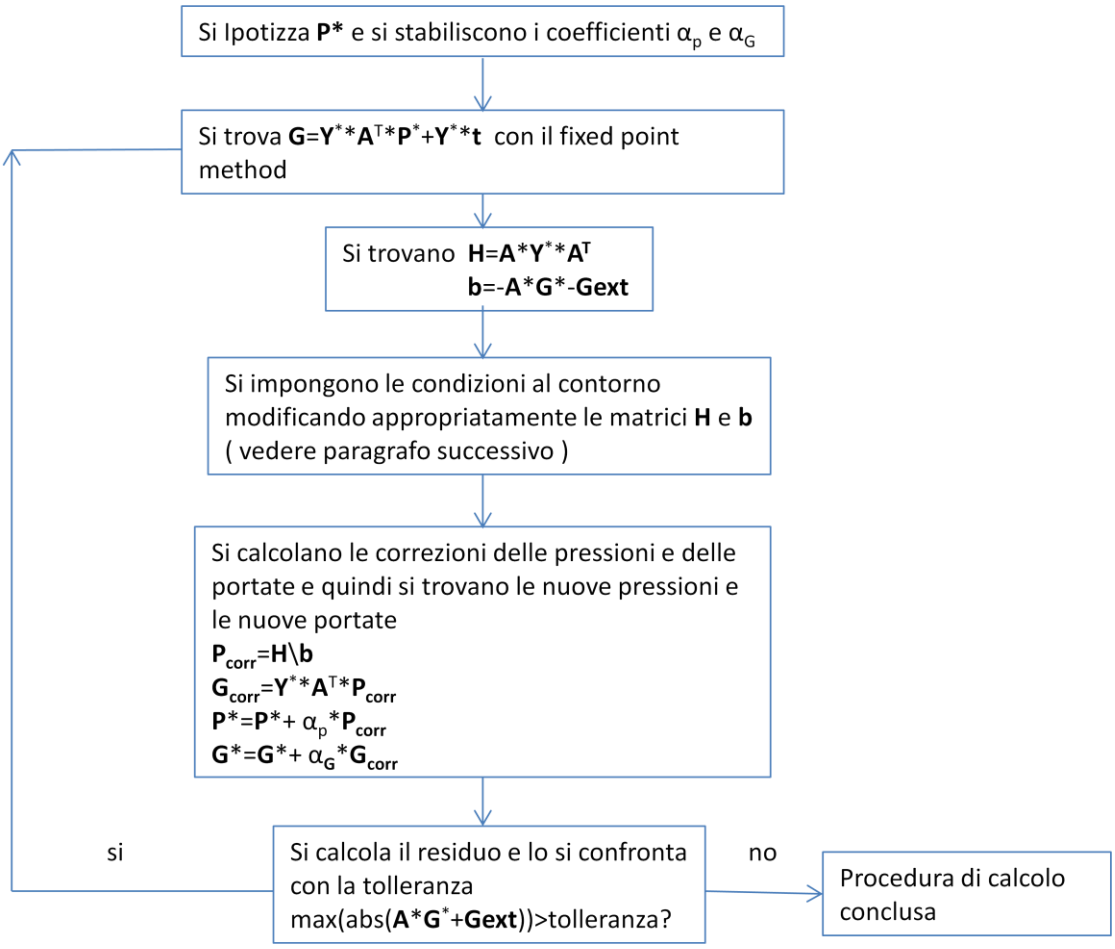
Per ridurre il più possibile il costo computazionale si possono cercare i valori ottimali per i coefficienti α_p e α_G. Sfortunatamente questi valori ottimali dipendono fortemente dalla rete studiata e non vi è un metodo che consente di trovarli senza effettuare un gran numero di simulazioni in cui si variano α_p e α_G.

Per capire quando fermare il ciclo dobbiamo calcolare il residuo per le portate definito come

$$\boldsymbol{R} = \boldsymbol{A} * \boldsymbol{G_{new}} + \boldsymbol{G_{ext}} \qquad (2.32)$$

Termineremo quindi di ciclare l'algoritmo quando il massimo del valore assoluto di **R** risulta minore di una tolleranza prestabilita da noi a seconda della precisione desiderata.

Qui di seguito è presente uno schema riassuntivo dell'algoritmo



(Figura 2.6)

Condizioni al contorno

Le condizioni al contorno del problema fluidodinamico riguardano le portate e le pressioni

Per specificare le portate in ingresso e in uscita dalla rete si utilizza il vettore colonna $\mathbf{G_{ext}}$ nel seguente modo:

- se il nodo i è un nodo di ingresso nella rete, l'elemento i-esimo del vettore $\mathbf{G_{ext}}$ deve essere pari alla portata in ingresso nel nodo i con segno negativo;
- se il nodo i è un nodo di uscita dalla rete, l'elemento i-esimo del vettore $\mathbf{G_{ext}}$ deve essere pari alla portata in uscita dal nodo i con segno positivo;
- se il nodo i non è né di ingresso, né di uscita dalla rete, l'elemento i-esimo del vettore $\mathbf{G_{ext}}$ deve essere nullo.

Per imporre invece una determinata pressione nel nodo i bisogna :

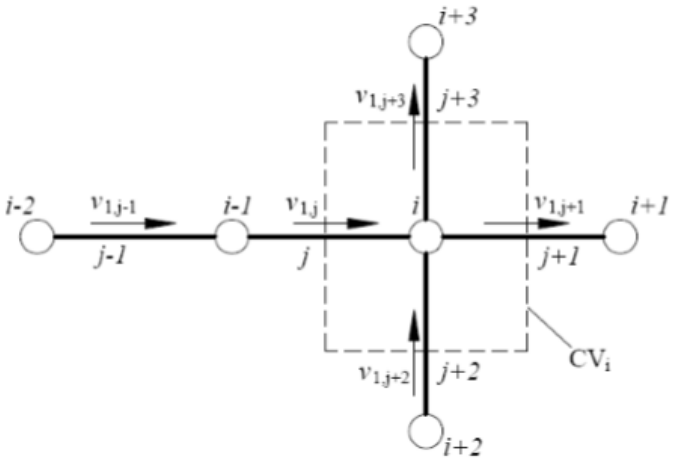
- inserire il valore da imporre nell'i-esimo elemento del vettore $\mathbf{P}^*$ ipotizzato prima del primo ciclo;
- modificare ad ogni ciclo le matrici **H** e **b** in modo che l'elemento i-esimo del vettore delle correzioni delle pressioni $\mathbf{P_{corr}}$ sia nullo.
  Per garantire ciò quindi:
  - la riga i-esima di H deve essere composta da elementi nulli ad eccezione dell'elemento i-esimo pari ad un valore qualsiasi diverso da zero;
  - L'elemento i-esimo del vettore b deve essere nullo.

2.2.4 SOLUZIONE DEL PROBLEMA FLUIDODINAMICO IN CONDIZIONI TRANSITORIE

Il procedimento è analogo a quello utilizzato per lo studio in stazionario, con la differenza che non si annullano le derivate nel tempo.

Conservazione della massa

Consideriamo il nodo generico i ed il relativo volume di controllo già definito per lo studio fluidodinamico in condizioni stazionarie.



(Figura 2.7)

Dall'equazione 2.10, esprimendo la massa contenuta nel volume di controllo con $M_i=\rho_i*V_i$ e ponendo $\rho_j*v_j*S_j=G_j$ otteniamo:

$$\frac{d\rho}{dt} * \Delta V_i + \sum_j G_j + G_{ext} = 0 \qquad (2.33)$$

Discretizzando nel tempo con il *backward euler method* (descritto nell'appendice)*:*

$$\frac{\rho^t-\rho^{t-\Delta t}}{\Delta t} * \Delta V_i + \sum_j G_j^t + G_{ext,i}^t = 0 \qquad (2.34)$$

Dove Δt è lo step temporale utilizzato e gli apici t e t- Δt indicano l'istante di tempo a cui la grandezza si riferisce.

Siccome il volume di controllo contiene la metà di tutti i rami collegati al nodo i, si ha

$$\Delta V_i = \sum_j \frac{S_j*L_j}{2} \qquad (2.35)$$

Quindi l'equazione 2.34 diventa

$$\frac{\rho^t-\rho^{t-\Delta t}}{\Delta t} * \left(\sum_j \frac{S_j*L_j}{2}\right) + \sum_j G_j^t + G_{ext,i}^t = 0 \quad \text{con j che indica i rami collegati ad i.} \qquad (2.36)$$

Scrivendo l'equazione 2.36 per tutti i nodi e compattando in forma matriciale l'insieme di equazioni cosi ottenuto si giunge alla seguente espressione matriciale:

$$\boldsymbol{A} * \boldsymbol{G}^t + \boldsymbol{G}_{\boldsymbol{ext}}^t + \boldsymbol{r}^t = 0 \qquad (2.37)$$

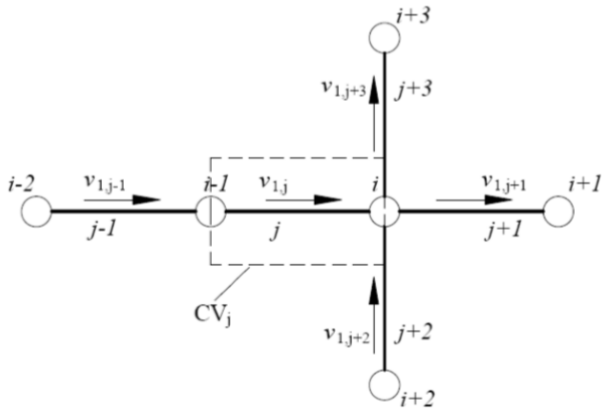
Dove:

- $\mathbf{G}^t$ e $\mathbf{G_{ext}}^t$ sono analoghi a **G** e $\mathbf{G_{ext}}$ del caso stazionario
- $\mathbf{r}^t$ è un vettore colonna lungo n_N n cui l'elemento i-esimo vale

$$\frac{\rho^t-\rho^{t-\Delta t}}{\Delta t} * \left(\sum_j \frac{S_j*L_j}{2}\right)_i \qquad (2.38)$$

Conservazione della quantità di moto

Consideriamo un ramo generico j ed il relativo volume di controllo CV_j mostrato in figura



(Figura 2.8)

Dalle equazioni 2.5, 2.6 e 2.7 si ha

$$\rho * \frac{\partial v_{1,j}}{\partial t} * L_j + (P_i - P_{i-1}) = -\frac{1}{2} * f_j * \frac{L_j}{D_{h,j}} * \rho * {v_{1,j}}^2 - \frac{1}{2} * \sum_k \beta_{k,j} * \rho * {v_{1,j}}^2 + \Delta P_{PUMP,j} \quad (2.39)$$

Utilizzando $\rho * v_{1,j} * S_j = G_j$ l'espressione 2.14 può essere scritta come

$$-\rho * \frac{d}{dt}\left(\frac{G_j}{\rho * S_j}\right) * L_j + (P_{i-1} - P_i) = \frac{1}{2} * \frac{G_j^2}{\rho * S_j^2} * \left(f_j * \frac{L_j}{D_{h,j}} + \sum_k \beta_{k,j}\right) - \Delta P_{PUMP,j} \quad (2.40)$$

Discretizzando nel tempo con il backward Euler method ed utilizzando l'ipotesi di densità costante:

$$-\rho^t * \frac{L_j * S_j}{\Delta t} * \left(\frac{G_j^t}{\rho^t} - \frac{G_j^{t-\Delta t}}{\rho^{t-\Delta t}}\right) + (P_{i-1} - P_i)^t == \frac{1}{2 * S_j^2} * \left(f_j * \frac{L_j}{D_{h,j}} + \sum_k \beta_{k,j}\right) * \left(\frac{G_j^2}{\rho}\right)^t - \Delta P_{PUMP,j}^t \quad (2.41)$$

L'equazione di conservazione del momento 2.41 può essere riformulata nel seguente modo

$$(P_{i-1} - P_i)^t = R_j^t * G_j^t + C_j^t * G_j^t - \Delta P_{PUMP,j}^t - s_j^t \quad (2.42)$$

Dove:

- R_j è espressa con l'equazione 2.17
- $C_j^t = \frac{L_j * S_j}{\Delta t}$ (2.43)
- $s_j^t = -\rho^t * \frac{L_j * S_j}{\Delta t} * \frac{G_j^{t-\Delta t}}{\rho^{t-\Delta t}}$ (2.44)

Scrivendo l'equazione 2.42 per tutti i rami j e compattando in forma matriciale l'insieme di equazioni così ottenuto si ha:

$$\boldsymbol{A}^T * \boldsymbol{P}^t = (\boldsymbol{R}^t + \boldsymbol{C}^t) * \boldsymbol{G}^t - \boldsymbol{t}^t - \boldsymbol{s}^t \quad (2.45)$$

Dove

- **R**,**P** e **T** sono state definite nel paragrafo relativo allo studio fluidodinamico in condizioni stazionarie
- **C** è una matrice diagonale di dimensione n_R x n_R contenente nella diagonale principale gli elementi Cj descritti dall'equazione 2.44

L'equazione 2.45 può essere riformulata in modo da avere un'espressione simile a quella che si ottiene nella conservazione della quantità di moto per il caso stazionario

$$\boldsymbol{G}^t = \overline{\boldsymbol{Y}}^t * \boldsymbol{A}^T * \boldsymbol{P}^t + \overline{\boldsymbol{Y}}^t * \boldsymbol{t}^t + \overline{\boldsymbol{Y}}^t * \boldsymbol{s}^t \quad (2.46)$$

Dove $\overline{\boldsymbol{Y}} = \boldsymbol{R}^{-1} + \boldsymbol{C}^{-1}$ è la seguente matrice diagonale

$$\overline{\mathbf{Y}} = \begin{bmatrix} (1/R_1 + 1/C_1) & 0 & \cdots & 0 \\ 0 & (1/R_2 + 1/C_2) & 0 & \vdots \\ \vdots & 0 & \ddots & \vdots \\ 0 & \cdots & 0 & (1/R_M + 1/C_M) \end{bmatrix}$$

Algoritmo SIMPLE

La logica è del tutto analoga a quella vista per il caso stazionario; nelle equazioni compaiono soltanto dei termini in più derivanti dalle derivate nel tempo.

Ipotizzato **P***, mediante il fixed point alghoritm si calcola $\mathbf{G}^{t}$***** :

$$\boldsymbol{G^{t^*}} = \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T * \boldsymbol{P}^{t^*} + \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{t}^t + \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{s}^t \quad (2.47)$$

Il calcolo di **Y*** è mostrato nell'appendice (vedere fixed point alghoritm)

Utilizzando ora le correzioni alle portate e alle pressioni

$$\boldsymbol{G_{corr}} = \boldsymbol{G}^t - \boldsymbol{G}^{t^*} \quad (2.48)$$

$$\boldsymbol{P_{corr}} = \boldsymbol{P}^t - \boldsymbol{P}^{t^*} \quad (2.49)$$

Dalle equazioni 2.48, 2.46 e 2.47 si ha

$$\boldsymbol{G_{corr}} = \boldsymbol{G}^t - \boldsymbol{G}^{t^*} = \overline{\boldsymbol{Y}}^t * \boldsymbol{A}^T * \boldsymbol{P}^t + \overline{\boldsymbol{Y}}^t * \boldsymbol{t}^t + \overline{\boldsymbol{Y}}^t * \boldsymbol{s}^t - \left(\overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T * \boldsymbol{P}^{t^*} + \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{t}^t + \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{s}^t\right)$$

(2.50)

Introducendo ora l'approssimazione $\overline{\boldsymbol{Y}}^t = \overline{\boldsymbol{Y}}^{t^*}$ otteniamo la relazione tra il vettore delle correzioni delle portate e quello delle correzioni delle pressioni

$$\boldsymbol{G_{corr}} = \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T * (\boldsymbol{P} - \boldsymbol{P}^*) = \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T * \boldsymbol{P_{corr}} \quad (2.51)$$

Dalla conservazione della massa e dalla definizione del vettore delle correzioni delle portate:

$$\begin{cases} \boldsymbol{A} * \boldsymbol{G}^t + \boldsymbol{G}^t_{\boldsymbol{ext}} + \boldsymbol{r}^t = 0 \\ \boldsymbol{G_{corr}} = \boldsymbol{G}^t - \boldsymbol{G}^{t^*} \end{cases} \rightarrow \boldsymbol{A} * \left(\boldsymbol{G}^{t^*} + \boldsymbol{G_{corr}}\right) + \boldsymbol{G}^t_{\boldsymbol{ext}} = 0;\ \boldsymbol{A} * \boldsymbol{G_{corr}} = -\boldsymbol{A} * \boldsymbol{G}^{t^*} - \boldsymbol{G}^t_{\boldsymbol{ext}} \quad (2.52)$$

Dalle equazioni 2.51 e 2.52 otteniamo infine

$$\boldsymbol{A} * \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T * \boldsymbol{P_{corr}} = -\boldsymbol{A} * \boldsymbol{G}^{t^*} - \boldsymbol{G}^t_{\boldsymbol{ext}} \quad (2.53)$$

Introducendo ora i vettori $\mathbf{H}^t$ e $\mathbf{b}^t$ otteniamo l'equazione 2.56

$$\boldsymbol{H}^t = \boldsymbol{A} * \overline{\boldsymbol{Y}}^{t^*} * \boldsymbol{A}^T \quad (2.54)$$

$$\boldsymbol{b}^t = -\boldsymbol{A} * \boldsymbol{G}^{t^*} - \boldsymbol{G}^t_{\boldsymbol{ext}} \quad (2.55)$$

$$\boldsymbol{H}^t * \boldsymbol{P_{corr}} = \boldsymbol{b}^t \quad (2.56)$$

Mediante l'equazione 2.56 calcoliamo $\mathbf{P_{corr}}$ ed in seguito con l'equazione 2.51 si ottiene $\mathbf{G_{corr}}$.

Anche nel caso transitorio, per evitare qualsiasi divergenza, nell' aggiornamento delle pressioni e delle portate bisogna utilizzare i coefficienti α_p e α_G con valore compreso tra 0 e 1.

$$\boldsymbol{P}^t_{\boldsymbol{new}} = \boldsymbol{P}^{t^*} + \alpha_p * \boldsymbol{P_{corr}} \quad (2.57)$$

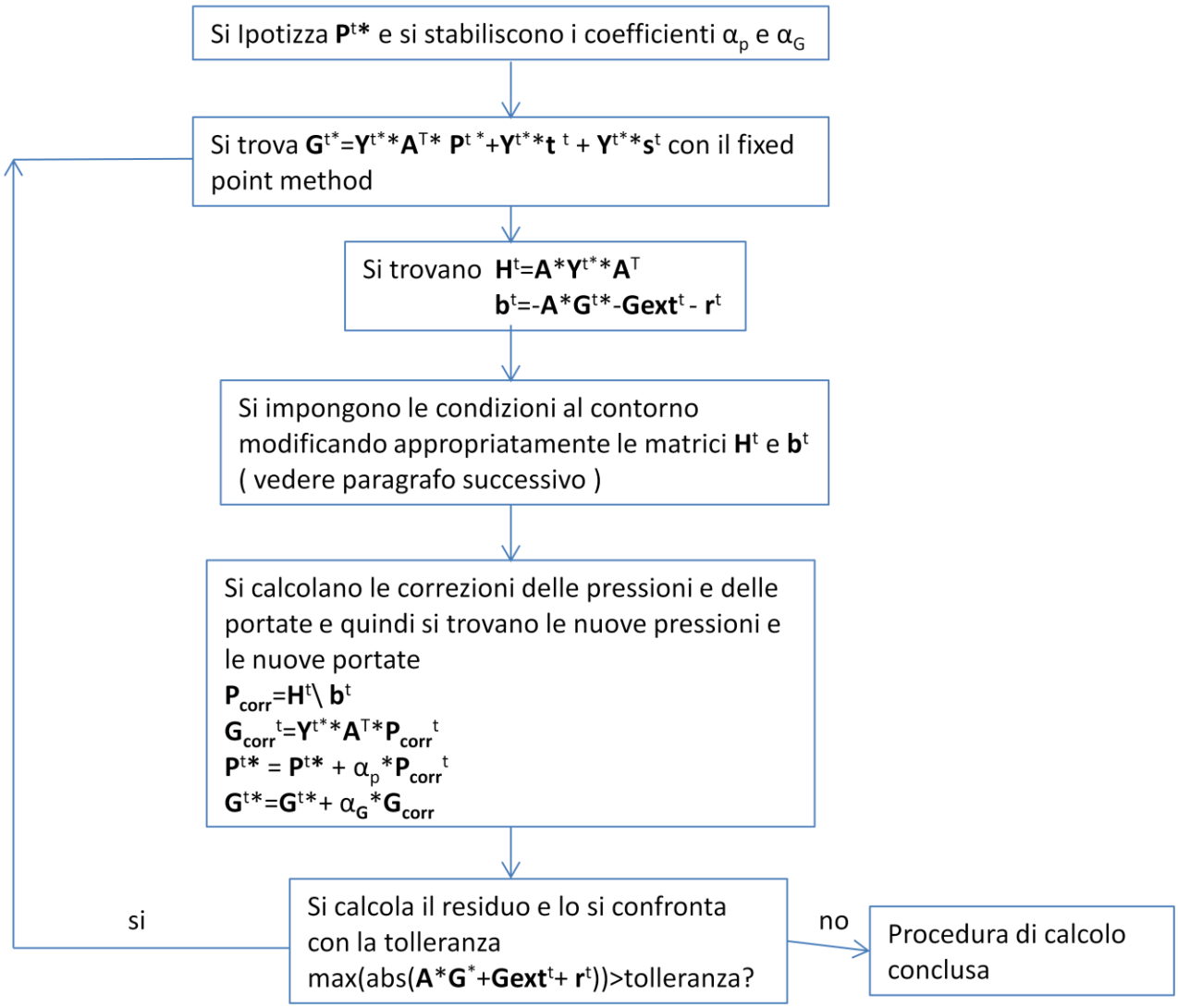
$$\boldsymbol{G}^t_{\boldsymbol{new}} = \boldsymbol{G}^{t^*} + \alpha_G * \boldsymbol{G_{corr}} \quad (2.58)$$

Per capire quando fermare il ciclo dobbiamo calcolare il vettore dei residui per le portate definito come:

$$R = \boldsymbol{A} * \boldsymbol{G}_{\boldsymbol{new}}^{t} + \boldsymbol{G}_{\boldsymbol{ext}}^{t} \qquad (2.59)$$

Termineremo quindi di ciclare l'algoritmo quando il massimo del valore assoluto di **R** risulta minore di una tolleranza prestabilita da noi a seconda della precisione desiderata.

Nella pagina seguente è presente uno schema riassuntivo dell'algoritmo



(Figura 2.9)

L'algoritmo schematizzato nella figura 2.9 deve essere ripetuto per ogni istante di tempo t considerato

Le condizioni al contorno del modello fluidodinamico per il caso transitorio si impongono in modo analogo a come mostrato per le condizioni al contorno del modello fluidodinamico in condizioni stazionarie.

2.3 MODELLO TERMICO

La risoluzione del problema termico consiste nel calcolo delle temperature in tutti i nodi, utilizzando l'equazione di conservazione dell'energia.
Ipotizzando di avere un fluido incomprimibile e di poter trascurare il riscaldamento dovuto alla frizione tra il fluido stesso e le tubazioni, l'equazione monodimensionale di conservazione dell'energia applicata ad un nodo è la seguente

$$\underbrace{\frac{\partial(\rho * c_p * T_i)}{\partial t} * \Delta V}_{1} + \underbrace{\sum_j c_p * G_j * T_j = k *}_{2} \underbrace{\left(\frac{\partial T}{\partial x}\right)_j * S_j}_{3} + \underbrace{Q_v}_{4} - \underbrace{Q_l}_{5} \qquad (2.60)$$

Dove

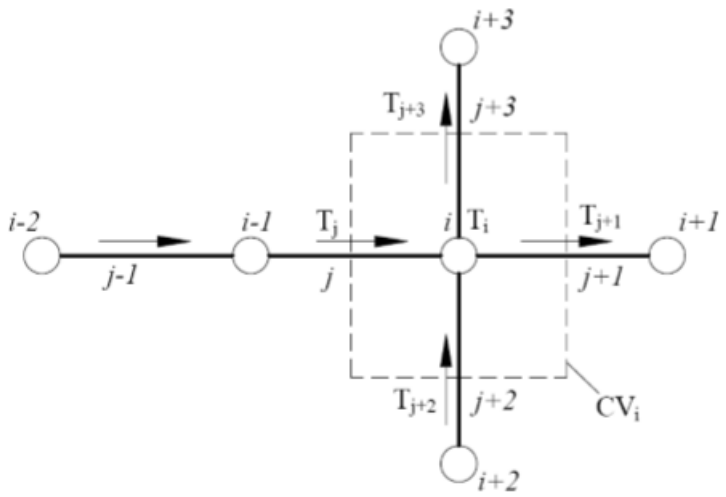
- Il termine 1 indica la variazione di energia nell'intervallo di tempo dt e nel volume di controllo ΔV
- Il termine 2 indica la variazione di energia dovuta alle portate entranti (segno negativo) ed uscenti (segno positivo) dal nodo
- Il termine 3 indica l'energia trasferita per conduzione e quindi, avendo a che fare con un fluido, è trascurabile.
- Il termine 4 si riferisce alla generazione interna di calore (per le reti di teleriscaldamento questo termine è sempre nullo)
- Il termine 5 si riferisce alle dissipazioni termiche verso l'esterno.

Sia per quanto riguarda il caso stazionario che per quello transitorio, le temperature sono state trovate applicando l'equazione 2.60 a tutti i nodi della rete; successivamente il sistema così ottenuto è stato scritto in forma matriciale ed infine è stata usata la funzione *mldivide* contenuta in matlab.

2.3.1 SOLUZIONE DEL PROBLEMA TERMICO IN CONDIZIONI STAZIONARIE

Nodi interni

Consideriamo un nodo generico i ed il relativo volume di controllo CV_i comprendente il nodo stesso e la metà di tutti i rami ed esso connessi.



(Figura 2.10)

Applicando l'equazione (2.60) e trascurando i termini 1 (lo studio in stazionario implica l'annullamento delle derivate nel tempo), 3 e 4 si ha:

$$\sum_j c_p * G_j * T_j = -Q_{l,i} \qquad (2.61)$$

La dissipazione verso l'esterno può essere espressa con:

$$Q_{l,i} = \sum_j \frac{L_j}{2} * \Omega_j * U_j * (T_i - T_G) \qquad (2.62)$$

Dove Ω_j è il perimetro della sezione della tubazione j perpendicolare al flusso d'acqua ,U_j è la trasmittanza termica e T_G è la temperatura dell'ambiente esterno (ovvero la temperatura del terreno).

Quindi unendo le equazioni (2.61) e (2.62) si ottiene la seguente espressione

$$\sum_j c_p * G_j * T_j = -\sum_j \frac{L_j}{2} * \Omega_j * U_j * (T_i - T_G) \qquad (2.63)$$

Con riferimento alla figura (2.10), applicando l'equazione (2.63) si ottiene

$$c_p * \left(G_j + G_{j+3}\right) * T_i - c_p * G_{j-1} * T_{i-1} - c_p * G_{j+2} * T_{i+2} = -\left(\sum_j \frac{L_j}{2} * \Omega_j * U_j\right) * (T_i - T_G) \qquad (2.64)$$

Nel passaggio dall'equazione 2.63 all'equazione 2.64 è stato applicato l' *upwind scheme* in cui la temperatura Tj viene approssimata alla temperatura del nodo all'ingresso del ramo j.

Raggruppando i termini in base alle temperature:

$$\left[c_p * \left(G_j + G_{j+3}\right) + \sum_j \frac{L_j}{2} * \Omega_j * U_j\right] * T_i + \left[-c_p * G_{j-1}\right] * T_{i-1} + \left[-c_p * G_{j+2}\right] * T_{i+2} = \left(\sum_j \frac{L_j}{2} * \Omega_j * U_j\right) * T_G \qquad (2.65)$$

Scrivendo l'equazione 2.65 per tutti i nodi i della rete otteniamo una serie di equazioni che può essere compattata nella seguente forma matriciale

$$\boldsymbol{K} * \boldsymbol{T} = \boldsymbol{f} \qquad (2.66)$$

Dove

- **K** è una matrice quadrata N x N con N numero di nodi della rete.
  Ogni riga i di questa matrice contiene i coefficienti di tutte le temperature, trovati applicando l'equazione di conservazione dell'energia al nodo i.
  Ogni colonna j di questa matrice contiene i coefficienti della temperatura T_j trovati applicando l'equazione di conservazione dell'energia a tutti i nodi.
  Quindi, con riferimento alla rete rappresentata in figura 2.10, nella riga i-esima della matrice k troveremo i seguenti coefficienti

  $$\begin{bmatrix} 0 & k_{i,i-1} & k_{i,i} & 0 & k_{i,i+2} & 0 \end{bmatrix} \qquad (2.67)$$

  Dove

  $$k_{i,i-1} = -c_p * G_{j-1} \qquad (2.68)$$

  $$k_{i,i} = c_p * \left(G_j + G_{j+3}\right) + \sum_j \frac{L_j}{2} * \Omega_j * U_j \qquad (2.69)$$

  $$k_{i,i+2} = -c_p * G_{j+2} \qquad (2.70)$$

- **T** è un vettore colonna lungo N contenente i valori delle temperature nei vari nodi; questo vettore non è altro che l'incognita del nostro problema.
  Con riferimento alla figura 2.10 **T** vale

  $$\begin{bmatrix} T_{i-2} \\ T_{i-1} \\ T_i \\ T_{i+1} \\ T_{i+2} \\ T_{i+3} \end{bmatrix}$$

- **f** è un vettore colonna contenente i termini noti, ovvero i termini in cui non compaiono le temperature dei nodi della rete.
  Il coefficiente f_i relativo alla riga i (e quindi al nodo i) è

  $$f_i = \left(\sum_j \frac{L_j}{2} * \Omega_j * U_j\right) * T_G \qquad (2.71)$$

Condizioni al contorno

Le condizioni al contorno vengono applicate ai nodi di ingresso e di uscita della rete considerata e vengono imposte modificando le matrici **K** ed **f**.

Per quanto riguarda i nodi di ingresso nella rete si ha una condizione di temperatura imposta.
Per assegnare quindi una determinata temperatura T_i al nodo i bisogna effettuare i seguenti passaggi:

- annullare tutti gli elementi della matrice **K** che si trovano sulla riga i ad eccezione di $k_{i,i}$ che deve essere posto pari ad 1.
- Assegnare all'elemento di **f** sulla riga i il valore T_i

Relativamente ai nodi di uscita si ha una condizione di flusso imposto; ovvero si eguagliano le dissipazioni del fluido che attraversa il ramo in esame con il flusso termico che dalle tubazioni viene disperso nell'ambiente.
Il volume di controllo è diverso da quello considerato per i nodi interni alla rete: in questo caso, infatti, si prende in esame tutto il ramo che entra nel nodo di uscita al quale si vuole applicare la condizione al contorno.
Consideriamo ad esempio il nodo di uscita i+3 della rete rappresentata nella figura 2.10; dalla conservazione dell'energia si ha:

$$c_p * G_{j+3} * (T_i - T_{i+3}) = L_{j+3} * Ω_{j+3} * U_{j+3} * \left(T_{j+3} - T_G\right) \quad (2.72)$$

Dove la temperatura del ramo j+3 è la media tra la temperatura del nodo di ingresso e quella del nodo di uscita del ramo stesso

$$T_{j+3} = \frac{T_{i+3}+T_i}{2} \quad (2.73)$$

Sostituendo quindi l'equazione 2.73 nella 2.72 si ottiene

$$c_p * G_{j+3} * (T_i - T_{i+3}) = L_{j+3} * Ω_{j+3} * U_{j+3} * \left(\frac{T_{i+3}+T_i}{2} - T_G\right) \quad (2.74)$$

Raggruppando i termini in base alle temperature:

$$\left[-c_p * G_{j+3} - \frac{L_{j+3}*Ω_{j+3}*U_{j+3}}{2}\right] * T_{i+3} + \left[c_p * G_{j+3} - \frac{L_{j+3}*Ω_{j+3}*U_{j+3}}{2}\right] * T_i = -L_{j+3} * Ω_{j+3} * U_{j+3} * T_G \quad (2.75)$$

Di conseguenza la riga i+3 della matrice **K** risulta:

$$\begin{bmatrix}0 & 0 & k_{i+3,i} & 0 & 0 & k_{i+3,i+3}\end{bmatrix} \quad (2.76)$$

Dove

$$k_{i+3,i} = \left[c_p * G_{j+3} - \frac{L_{j+3}*Ω_{j+3}*U_{j+3}}{2}\right] \quad (2.77)$$

$$k_{i+3,i+3} = \left[-c_p * G_{j+3} - \frac{L_{j+3}*Ω_{j+3}*U_{j+3}}{2}\right] \quad (2.78)$$

L'elemento i+3 del vettore colonna **f** invece deve essere posto pari a:

$$f_{i+3} = -L_{j+3} * Ω_{j+3} * U_{j+3} * T_G \quad (2.79)$$

2.3.2 SOLUZIONE DEL PROBLEMA TERMICO IN CONDIZIONI TRANSITORIE

Nodi interni

Consideriamo il nodo generico i ed il volume di controllo indicati nella figura 2.10.
Dato che ora vogliamo esaminare il comportamento della rete nel transitorio, non possiamo annullare la derivata nel tempo e quindi solo i termini 3 e 4 dell'equazione 2.60 possono essere trascurati. Di conseguenza l'equazione di conservazione dell'energia assume la seguente espressione:

$$\frac{\partial(\rho*c_p*T_i)}{\partial t} * \Delta V + \sum_j c_{p,j} * G_j * T_j = -Q_l \quad (2.80)$$

Inserendo l'equazione 2.62 otteniamo perciò

$$\frac{\partial(\rho*c_p*T_i)}{\partial t} * \Delta V + \sum_j c_p * G_j * T_j = -\sum_j \frac{L_j}{2} * \Omega_j * U_j * (T_i - T_G) \quad (2.81)$$

Con $\Delta V = \sum_j \frac{S_j*L_j}{2}$ (2.82)

Discretizzando nel tempo usando il 'backward Euler method' abbiamo

$$\frac{(\rho*c_p*T_i)^t - (\rho*c_p*T_i)^{t-\Delta t}}{\Delta t} * \sum_j \frac{S_j*L_j}{2} + \sum_j c_p * G_j^t * T_j^t = -\sum_j \frac{L_j}{2} * \Omega_j * U_j * (T_i^t - T_G) \quad (2.83)$$

Dove l'apice t indica che la grandezza si riferisce all'istante di tempo t e l'apice t-Δt indica che la grandezza si riferisce all'istante di tempo precedente a t.
Applicando la 2.83 al volume di controllo della figura 2.10 otteniamo

$$\frac{\rho * c_p}{2 * \Delta t} * \left(\sum_j S_j * L_j\right) * \left(T_i^t - T_i^{t-\Delta t}\right) + c_p * G_{j+1}^t * T_i^t + c_p * G_{j+3}^t * T_i^t - c_p * G_j^t * T_{i-1}^t +$$

$$-c_p * G_{j+2}^t * T_{i+2}^t = -\sum_j \frac{L_j}{2} * \Omega_j * U_j * (T_i^t - T_G) \quad (2.84)$$

Raggruppando i termini in base alle temperature:

$$\left[\frac{\rho*c_p}{2*\Delta t} * \left(\sum_j S_j * L_j\right) + c_p * \left(G_{j+1}^t + G_{j+3}^t\right) + \sum_j \frac{L_j}{2} * \Omega_j * U_j\right] * T_i^t + \left[-c_p * G_j^t\right] * T_{i-1}^t +$$

$$\left[-c_p * G_{j+2}^t\right] * T_{i+2}^t = \left(\sum_j \frac{L_j}{2} * \Omega_j * U_j\right) * T_G + \frac{\rho*c_p}{2*\Delta t} * \left(\sum_j S_j * L_j\right) * T_i^{t-\Delta t} \quad (2.85)$$

Scrivendo l'equazione 2.85 per tutti i nodi i della rete otteniamo una serie di equazioni che può essere compattata nella forma matriciale **K*T=f** che abbiamo già visto nella soluzione del problema termico in condizioni stazionarie.
Quindi, con riferimento alla rete rappresentata in figura 2.10, nella riga i-esima della matrice k troveremo i seguenti coefficienti

$$\begin{bmatrix} 0 & k_{i,i-1} & k_{i,i} & 0 & k_{i,i+2} & 0 \end{bmatrix} \quad (2.86)$$

Dove

$$k_{i,i-1} = -c_p * G_j^t \quad (2.87)$$

$$k_{i,i} = \frac{\rho*c_p}{2*\Delta t} * \left(\sum_j S_j * L_j\right) + c_p * \left(G_{j+1}^t + G_{j+3}^t\right) + \sum_j \frac{L_j}{2} * \Omega_j * U_j \quad (2.88)$$

$$k_{i,i+2} = -c_p * G_{j+2} \quad (2.89)$$

Mentre l'elemento i-esimo del vettore colonna f è pari a

$$f_i = \left(\sum_j \frac{L_j}{2} * \Omega_j * U_j\right) * T_G + \frac{\rho*c_p}{2*\Delta t} * \left(\sum_j S_j * L_j\right) * T_i^{t-\Delta t} \quad (2.90)$$

Condizioni al contorno

Come abbiamo già visto in occasione dello studio in stazionario, anche nel transitorio, per imporre le condizioni al contorno, dobbiamo modificare i valori delle matrici **K** ed **f**.

Per quanto riguarda l'imposizione della temperatura al nodo di ingresso si segue il procedimento illustrato nello studio della rete in condizioni stazionarie.

Per quanto concerne la condizione da imporre ai nodi in uscita, si utilizza l'equazione di conservazione dell'energia, applicandola ad un volume di controllo che comprende esclusivamente il ramo entrante nel nodo considerato.
Di conseguenza il volume e il coefficiente di scambio termico vengono calcolati con

$$\Delta V = S_j * L_j \quad (2.91)$$

$$hL_j = L_j * Ω_j * U_j \ [W/K] \quad (2.92)$$

Come fatto per il caso stazionario, imponiamo la condizione al contorno per il nodo i+3 della rete rappresentata in figura 2.10
Quindi,considerando le equazioni 2.91, 2.92 e 2.60, la conservazione dell'energia per il ramo j+3 può essere espressa con:

$$\frac{\partial(\rho * c_p * \overline{T_{j+3}})}{\partial t} * S_{j+3} * L_{j+3} + c_p * G_{j+3} * (T_{i+3} - T_i) = -L_{j+3} * Ω_{j+3} * U_{j+3} * \left(\overline{T_{j+3}} - T_G\right) \quad (2.93)$$

Dove $\overline{T_{j+3}} = \frac{T_{i+3}+T_i}{2}$ è la temperatura media del ramo j+3.
Discretizzando nel tempo con il 'backward Euler method' si ottiene

$$\frac{\rho * c_p}{\Delta t} * S_{j+3} * L_{j+3} * \left(\frac{T_{i+3}^t + T_i^t}{2} - \frac{T_{i+3}^{t-\Delta t} + T_i^{t-\Delta t}}{2}\right) + c_p * G_{j+3} * (T_{i+3}^t - T_i^t) =$$

$$= -L_{j+3} * Ω_{j+3} * U_{j+3} * \left(\frac{T_{i+3}^t + T_i^t}{2} - T_G\right) \quad (2.94)$$

Raggruppando i termini in base alle temperature:

$$\left[\frac{\rho * c_p}{2 * \Delta t} * S_{j+3} * L_{j+3} + c_p * G_{j+3} + \frac{L_{j+3} * Ω_{j+3} * U_{j+3}}{2}\right] * T_{i+3}^t + \left[\frac{\rho * c_p}{2 * \Delta t} * S_{j+3} * L_{j+3} - c_p * G_{j+3} + \right.$$

$$Lj+3*Ωj+3*Uj+32*Tit = Lj+3*Ωj+3*Uj+3*TG + ρ*cpΔt*Sj+3*Lj+3*Ti+3t-Δt + Tit-Δt2 \quad (2.95)$$

Di conseguenza la riga i+3 della matrice **K** risulta:

$$\begin{bmatrix} 0 & 0 & k_{i+3,i} & 0 & 0 & k_{i+3,i+3} \end{bmatrix} \quad (2.96)$$

Dove

$$k_{i+3,i} = \frac{\rho * c_p}{2 * \Delta t} * S_{j+3} * L_{j+3} - c_p * G_{j+3} + \frac{L_{j+3} * Ω_{j+3} * U_{j+3}}{2} \quad (2.97)$$

$$k_{i+3,i+3} = \frac{\rho * c_p}{2 * \Delta t} * S_{j+3} * L_{j+3} + c_p * G_{j+3} + \frac{L_{j+3} * Ω_{j+3} * U_{j+3}}{2} \quad (2.98)$$

L'elemento i+3 del vettore colonna **f** invece deve essere posto pari a:

$$f_{i+3} = L_{j+3} * Ω_{j+3} * U_{j+3} * T_G + \frac{\rho * c_p}{\Delta t} * S_{j+3} * L_{j+3} * \frac{T_{i+3}^{t-\Delta t} + T_i^{t-\Delta t}}{2} \quad (2.99)$$

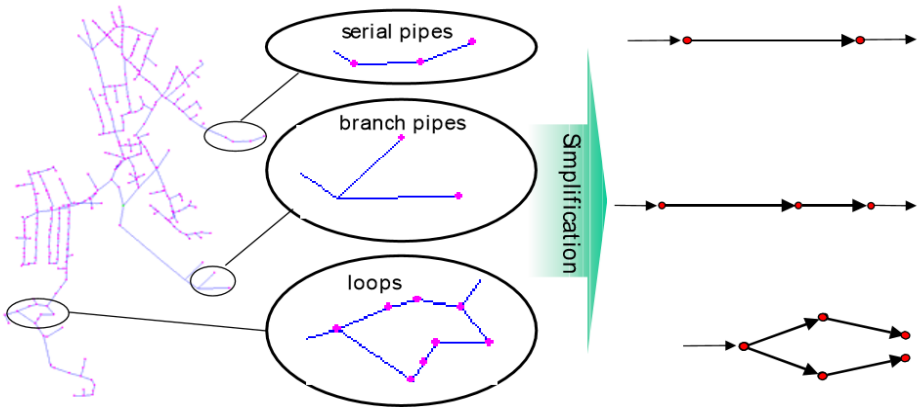
3 RIDUZIONE DEL COSTO COMPUTAZIONALE

Il costo computazionale non è altro che il tempo impiegato dal calcolatore per risolvere il problema.
Al giorno d'oggi vi sono reti di teleriscaldamento molto estese; di conseguenza, grazie anche alla presenza di due cicli iterativi (SIMPLE ALGHORITM) nella risoluzione del problema fluidodinamico, il costo computazionale può essere molto elevato.
E' necessario quindi elaborare dei modelli che permettano di simulare il comportamento delle reti di distribuzione e di trasporto in tempi contenuti.
Di recente, perciò , sono stati elaborate diverse soluzioni:

- vi sono i modelli POD RFB che si basano sul modello nodale precedentemente descritto.
  Al modello nodale viene fornito un vettore di input ed esso ci darà degli output.
  Effettuando diverse simulazioni con diversi input si giunge ad una relazione diretta tra il vettore di dati fornito al modello nodale e gli output. La relazione trovata ha però validità solo per la rete simulata;
- ci sono poi dei modelli di aggregazione fisica della rete in cui si utilizza un approccio completamente diverso: si va a modificare la morfologia e le caratteristiche della rete sostituendo , di fatto , la rete reale con una rete equivalente avente pochi rami.
  Questa rete equivalente deve naturalmente comportarsi come la rete originale in termini di dispersione di calore totale, portata totale, carico totale e tempo di ritardo totale del fronte di temperatura. Nel resto del lavoro ci concentreremo proprio su questi modelli.

I passaggi base di aggregazione della rete originale sono raffigurati nell'immagine sottostante:

- compattazione dei rami disposti in serie.
  Con quest'operazione si riduce quindi il numero dei nodi e dei rami di una rete;
- semplificazione delle ramificazioni.
  Si passa da una struttura ramificata ad una struttura in cui tutti i rami sono in serie;
- eliminazione dei circuiti chiusi (se il modello lo permette).
  Un circuito chiuso è caratterizzato dalla presenza di un nodo in cui confluiscono due rami.
  Nella semplificazione di questi circuiti chiusi si elimina il nodo di confluenza e la rete a valle di questo viene sostituita con due reti in parallelo.



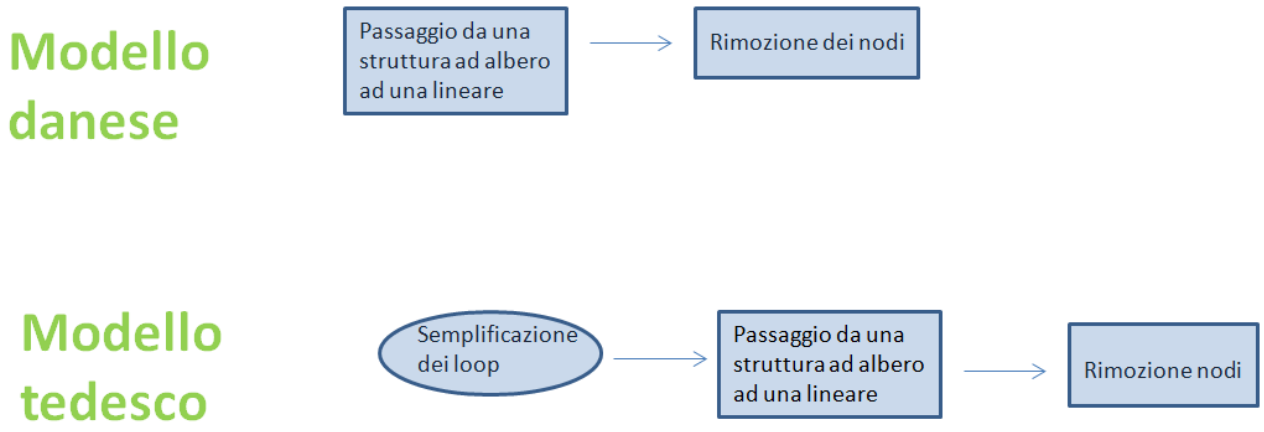
(Figura 3.1)

3.1 CONFRONTO MODELLI DI AGGREGAZIONE

Recentemente sono stati sviluppati due modelli di aggregazione: il modello tedesco e quello danese.
Entrambi i modelli hanno bisogno come input della distribuzione delle portate nella rete e quindi sono adatti ad uno studio in stazionario della stessa. Non avrebbe infatti senso costruire una rete aggregata equivalente per ogni variazione di carico di anche solo un' utenza; ciò infatti aumenterebbe il costo computazione mentre l'obiettivo di questi modelli è la diminuzione dello stesso.
Come però vedremo, per lo studio in transitorio, si possono comunque utilizzare le reti equivalenti ottenute dall'aggregazione dalla rete originale in condizioni di carico massimo.

Come mostrato nella figura sottostante, l'aggregazione può essere divisa in più blocchi in ognuno dei quali si compiono delle operazioni ben precise:

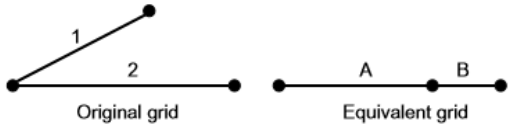
- semplificazione delle ramificazioni: si passa da una struttura ad albero ad una lineare;
- compattazione della rete : si eliminano i rami più corti ed i relativi nodi;
- semplificazione dei loop : si eliminano i circuiti chiusi. Quest' operazione è possibile solo nell'applicazione del modello tedesco.



(Figura 3.2)

3.1.1 PASSAGGIO DA UNA STRUTTURA AD ALBERO AD UNA STRUTTURA LINEARE

L'obiettivo di questa prima fase è schematizzato nella figura sottostante: vogliamo passare da una struttura ramificata ad una struttura lineare equivalente .



(Figura 3.3)

Qui di seguito troviamo una tabella che confronta le caratteristiche della rete equivalente rispetto a quella originale tra i due modelli

Comparison of aggregation methods of tree structures

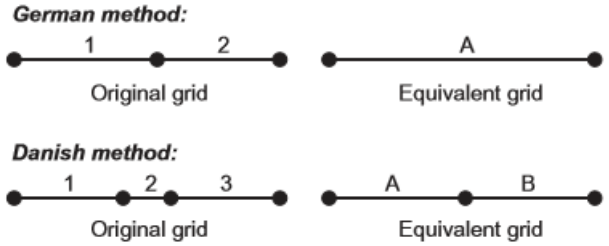
	German method conserves	Danish method conserves
Pipe length	LA+LB = L2	No
Pipe inner diameter	No	DB = D2
Water volume	Yes	Yes
Delay	Yes	Yes
Mass flow	Yes	Yes
Heat load	Yes	Yes
Heat loss from supply pipe	No[a]	Yes
Heat loss from return pipe	No[a]	Yes
Pressure drop	Yes	Not considered

3.1.2 RIMOZIONE DEI NODI

Durante lo step della rimozione dei nodi i due modelli sono molto differenti sia dal punto di vista topologico che da quello termico e fluidodinamico.

Differenza topologica

Il modello tedesco prende in considerazione due rami e li sostituisce con un solo ramo; il modello danese, invece, parte da una configurazione a 3 rami e giunge ad una struttura con due soli rami.



(Figura 3.4)

Differenze termiche

Nel modello tedesco le temperature di uscita dalle varie sottostazioni (lato primario) possono essere differenti; nel modello danese, invece, tutte le temperature di ritorno dai carichi (sempre lato primario) sono considerate uguali.

Diverse sono anche le imposizioni iniziali sulla rete equivalente:

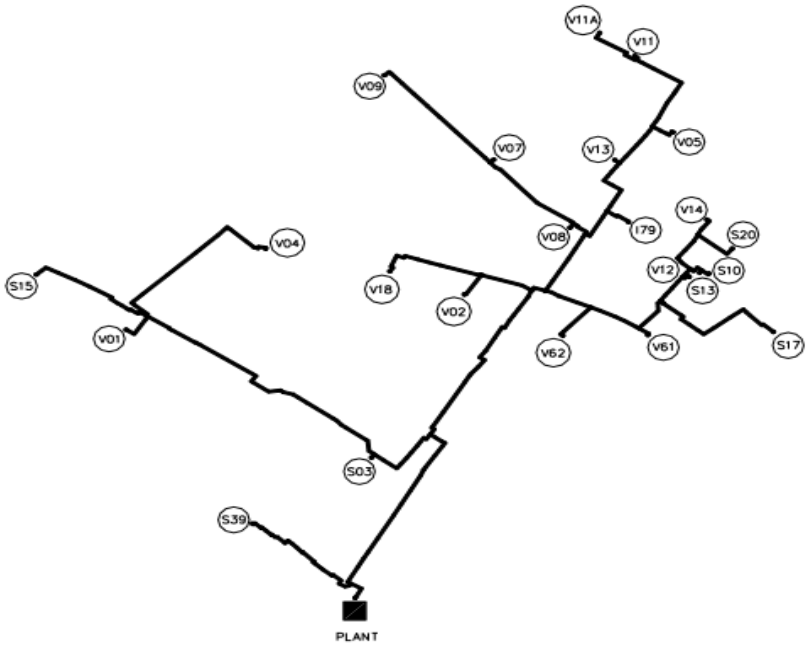
- nel modello tedesco i nodi che non sono stati eliminati preservano le temperature della rete originale e i coefficienti di dispersione termica sono calcolati in funzione di queste temperature. Inoltre conservando le temperature ai nodi, il volume e la portata totale, le dispersioni termiche risultano differenti;
- nel metodo danese,invece , si impone la conservazione delle dispersioni termiche e di conseguenza le temperature ai nodi non eliminati non corrispondono con quelle della rete originale.

Differenza fluidodinamica

Nel modello tedesco si modifica la rugosità dei condotti e i coefficienti di perdita concentrata in modo da mantenere le pressioni ai nodi non eliminati nella rete equivalente uguali a quelle che si hanno sulla rete originale.
Nel modello danese, al contrario, le pressioni non vengono considerate affatto.

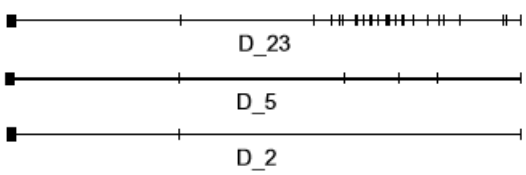
3.1.3 STRUTTURA DELLE RETI AGGREGATE

Nel modello di aggregazione danese prima si linearizza l'intera struttura e solo successivamente avviene l'eliminazione dei rami.
Utilizzando invece il modello di aggregazione tedesco , in base ai parametri che controllano il processo di aggregazione, la rete aggregata equivalente può essere lineare o contenere ramificazioni/loop.
A titolo di esempio, nelle immagini seguenti verranno mostrate delle aggregazioni della rete di Ishoej mediante i due modelli



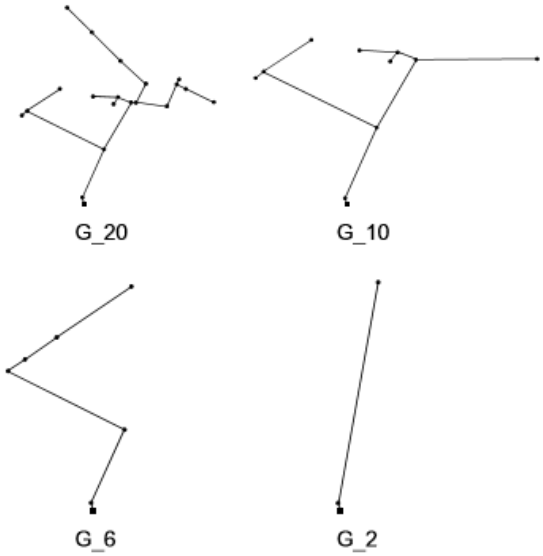
(rete originale con le sue 23 sottostazioni)

(Figura 3.5)



(Aggregazione mediante il modello danese)
-D indica il modello di aggregazione utilizzato.
-Il valore numerico indica il numero di rami di cui la rete aggregata è composta.

(Figura 3.6)



(Aggregazione mediante il modello tedesco)
-G indica il modello di aggregazione utilizzato.
-Il valore numerico indica il numero di rami di cui la rete aggregata è composta.

(Figura 3.7)

3.2 CENNI SUL MODELLO DI AGGREGAZIONE TEDESCO

Questo modello non verrà approfondito in quanto non è stato usato durante lo svolgimento della tesi. In questo paragrafo, pertanto, verranno solo fornite delle linee guida

3.2.1 LEGGI FONDAMENTALI

-La relazione tra la caduta di pressione P e la portata m' in un condotto è data dalla legge di Darcy-Weisbach.

$$\Delta P = \dot{m} * |\dot{m}| * r - \rho * g * \Delta h \quad (3.1)$$

Dove

- ρ è la densità dell'acqua in kg/m^3
- g è l'accelerazione gravitazionale in N/Kg
- La costante r vale

$$r = \frac{8}{\pi * \rho * d^4} * \left(\frac{l}{d} * \xi_R + \sum \xi_{zus}\right) \quad (3.2)$$

In cui

- ξ_{zus} indica le resistenze concentrate
- ξ_r è la frizione e deve essere calcolata in base alle caratteristiche del flusso
- d è il diametro interno del condotto

-La portata massica alle utenze è calcolabile con la seguente formula

$$\dot{m}_{VB} = \frac{\dot{Q}_{VB}}{c_{p,W} * (T_{VB}^E - T_{VB}^A)} \quad (3.3)$$

Dove

- il pedice VB indica l'utenza VB
- l'apice E indica che ci si sta riferendo alle condizioni di ingresso all'utenza
- l'apice A indica che ci si sta riferendo alle condizioni di uscita dall'utenza
- $C_{p,W}$ è il calore specifico dell'acqua
- Q' è il carico

-In caso di miscelazione si assume che questa sia perfetta, quindi si ha

$$T_K = \frac{\sum_{i=1}^{n} (\dot{m}_i^E * T_i^E)}{\sum_{i=1}^{n} \dot{m}_i^E} \quad (3.4)$$

In cui le sommatorie sono estese a tutti i rami in ingresso nel nodo di miscelazione

-Per un fluido che scorre in un condotto, si può trovare la temperatura uscente mediante la seguente equazione

$$T^A = T^U + (T^E - T^U) * exp\left(-\frac{\pi * k * l * d}{c_{p,w} * \dot{m}}\right) \quad (3.5)$$

Dove

- T^U è la temperatura ambiente
- T^A è la temperatura di uscita dalla tubazione
- T^E è la temperatura di ingresso nella tubazione
- l è la lunghezza della tubazione
- K è la trasmittanza termica

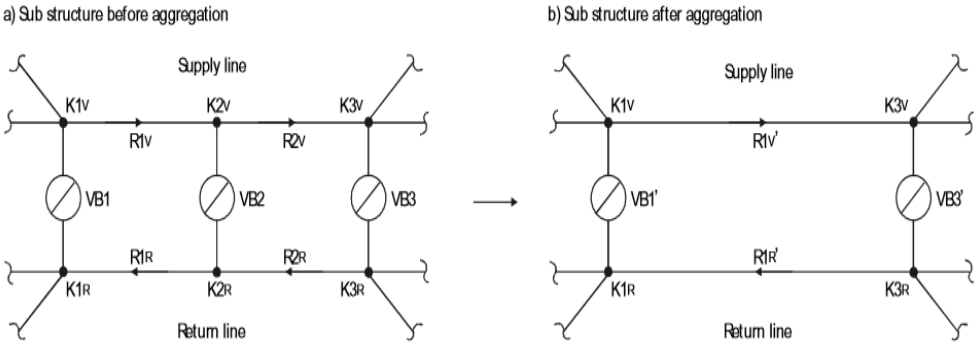
-Il tempo di percorrenza in secondi della tubazione τ (detto anche ritardo temporale) è calcolato in funzione del volume, della portata e della densità

$$\tau = \frac{V * \rho}{\dot{m}} \quad (3.6)$$

3.2.2 SEMPLIFICAZIONE DELLE DIVERSE STRUTTURE

a) <u>Compattazione di condotti in serie</u>

Come mostrato nella figura sottostante, i nodi ai quali è allacciata una sola utenza e che sono compresi tra due rami percorsi dal fluido nella stessa direzione, possono essere eliminati.



-R indica il ramo
-K indica il nodo
-VB indica le utenze
-L'apice V indica la mandata
-L'apice R indica il ritorno

(Figura 3.8)

Ad ogni compattazione, quindi, vengono eliminati due nodi (uno alla rete di mandata ed uno a quella di ritorno), due rami (uno alla rete di mandata ed uno a quella di ritorno) e un carico.
Il carico eliminato (VB2) viene distribuito tra le utenze adiacenti (VB1 e VB3).

Le condizioni da imporre durante quest'operazione sono le seguenti

1: Volume totale d'acqua costante
2: Ritardo temporale totale costante
3: Temperatura ai nodi non eliminati costante
4: Somma delle portate massiche alle utenze costante
5: Somma dei carichi termici costante

-Dall' imposizione 1 troviamo il diametro del ramo equivalente di mandata $d_{R1V'}$

$$\begin{cases} V_{R1V} + V_{R2V} = V_{R1V'} (imposizione\ 1) \\ V = L * \pi * \frac{d^2}{4} \end{cases} \rightarrow d_{R1V'} = \sqrt{\frac{d_{R1V}^2 * L_{R1V} + d_{R2V}^2 * L_{R2V}}{L_{R1V'}}} \qquad (3.7)$$

Dove la lunghezza del ramo di mandata $L_{R1V'}$ è determinata con metodi euristici

-Utilizzando ora l'imposizione 2 troviamo la portata che scorre nel ramo R1V'

$$\begin{cases} \tau_{R1} + \tau_{R2V} = \tau_{R1V'} (\ imposizione\ 2) \\ \tau = \frac{V * \rho}{\dot{m}} \end{cases} \rightarrow \dot{m}_{R1V'} = \frac{V_{R1V'} * \dot{m}_{R1V} * \dot{m}_{R2V}}{\dot{m}_{R2V} * V_{R1V} + \dot{m}_{R1V} * V_{R2V}} \qquad (3.8)$$

-Con l'imposizione 4 (conservazione della massa) si calcolano invece le portate massiche alle utenze equivalenti

$$\dot{m}_{VB1'} = \dot{m}_{in,K1V} - \dot{m}_{R1V'} \quad per\ il\ nodo\ K1V \qquad (3.9)$$

$$\dot{m}_{VB3'} = \dot{m}_{R1V'} - \dot{m}_{out,K3V} \quad per\ il\ nodo\ K3V \qquad (3.10)$$

-Applicando l'equazione della miscelazione perfetta al nodo K_{3R} si trova la temperatura all'uscita dall' utenza VB3'

$$\dot{m}_{VB3'} * T_{out,VB3'} + \dot{m}_{in,K3R} * T_{in,K3R} = \dot{m}_{R1R'} * T_{K3R} \rightarrow Trovo\ T_{out,VB3'} \quad (3.11)$$

-Si calcola ora il carico termico dell'utenza VB3'

$$\dot{Q}_{VB3'} = \dot{m}_{VB3'} * c_{p,w} * (T^{E}_{VB3'} - T^{A}_{VB3'}) \quad (3.12)$$

-Utilizzando l'imposizione 5 si calcola il carico termico dell'utenza VB1'

$$\dot{Q}_{VB1'} + \dot{Q}_{VB3'} = \dot{Q}_{VB1} + \dot{Q}_{VB2} + \dot{Q}_{VB3} \rightarrow Trovo\ \dot{Q}_{VB1'} \quad (3.13)$$

-Possiamo quindi trovare la temperatura in uscita dall'utenza VB1'

$$T^{A}_{VB1'} = T^{E}_{VB1'} - \frac{\dot{Q}_{VB1'}}{c_{p,w} * \dot{m}_{VB1'}} \quad (3.14)$$

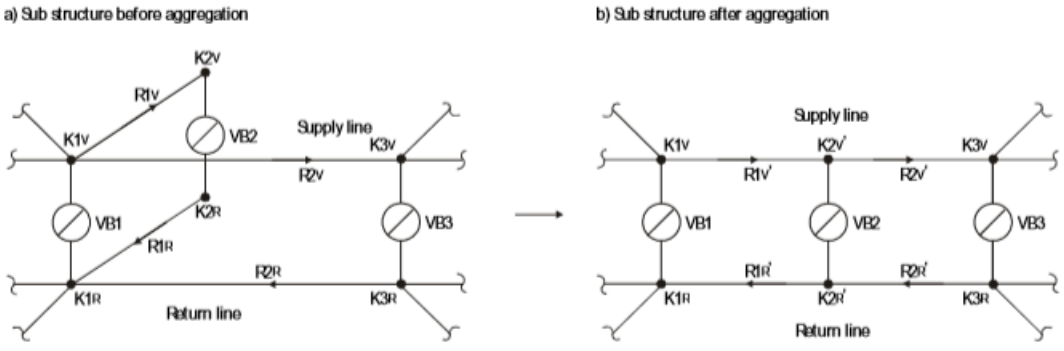
-Applicando la miscelazione perfetta (equazione (3.4)) al nodo K1R si trova la temperatura in ingresso al nodo K1R dal ramo R1R'. In seguito, utilizzando l'equazione 3.5 applicata al ramo R1R', si calcola il coefficiente di scambio del ramo R1R' (si deve esplicitare K)

-Applicando l'equazione 3.5 al ramo R1V' troviamo il coefficiente di scambio termico del ramo stesso (come per il ramo R1R').

-Infine, utilizzando l'equazione 3.1 si va a modificare il coefficiente r in entrambi i rami in modo da mantenere le pressioni ai nodi.

b) <u>Aggregazione di una ramificazione</u>

Consideriamo la seguente struttura in cui da un nodo (K1V) la portata fluisce in due condotti distinti (R1V e R2V).



(Figura 3.9)

La nomenclatura degli elementi è del tutto analoga a quella usata per la compattazione dei condotti in serie.

Affinché questa semplificazione sia possibile è necessario che al condotto R1V sia allacciata una sola utenza; se così non è, prima di eliminare la biforcazione ,bisogna procedere con la semplificazione della rete a valle della ramificazione .

I due rami in parallelo vengono quindi ricombinati in due rami in serie; pertanto non si ha alcuna riduzione degli elementi coinvolti.
-A causa dell'imposizione 2, la somma dei ritardi temporali dei rami R1V' e R2V' sarà pari al ritardo temporale maggiore tra quelli di R1V e R2V.
Ponendoci nel caso in cui $\tau_{R2V} > \tau_{R1V}$ allora, ipotizzando che $\tau_{R1V} = \tau_{R1V'}$:

$$\tau_{R1V'} = \tau_{R1V} \quad (3.15)$$

$$\tau_{R2V'} = \tau_{R2V} - \tau_{R1V} \quad (3.16)$$

-Le utenze non variano e quindi il carico termico e le portate alle utenze rimangono le stesse

-Per l'imposizione 4 il ramo R1V' deve contenere la portata necessaria ad entrambe le utenze più quella relativa alla rete a valle del nodo K3 (che però rimane invariata):

$$\dot{m}_{R1V'} = \dot{m}_{R1V} + \dot{m}_{R2V} \quad (3.17)$$

-Il ramo R2V' serve solo l'utenza VB3 oltre al resto della rete (che però rimane invariata), perciò

$$\dot{m}_{R2V'} = \dot{m}_{R2V} \quad (3.18)$$

-Imponendo $d_{R2V'} = d_{R2V}$ ed utilizzando l'equazione 3.16 con la 3.6 troviamo:

$$L_{R2V'} = L_{R2V} * (1 - \alpha * b) \quad (3.19)$$

Dove $\alpha = \dot{m}'_{R1V}/\dot{m}'_{R2V}$ e $b = V_{R1V}/V_{R2V}$

-Cerchiamo ora il diametro del ramo R1V' mediante l'imposizione 1

$$\begin{cases} V_{R1V} + V_{R2V} = V_{R1V'} + V_{R2V'} \\ V = L * \pi * \frac{d^2}{4} \end{cases} \rightarrow d_{R1V'} = \sqrt{\frac{{d_{R1V}}^2 * L_{R1V} + {d_{R2V}}^2 * L_{R2V} * \alpha * \beta}{L_{R1V'}}} \quad (3.20)$$

Dove

- $L_{R1V'}$ si trova con metodi euristici
- $V_{R2V'} = V_{R2V} * (1 - \alpha * \beta)$

-Utilizzando l'equazione 3.3 applicata all'utenza VB2 troviamo La temperatura in uscita dall'utenza stessa

-Dall'equazione di miscelazione perfetta applicata al nodo K2R' troviamo la temperatura di uscita dal ramo R2R'

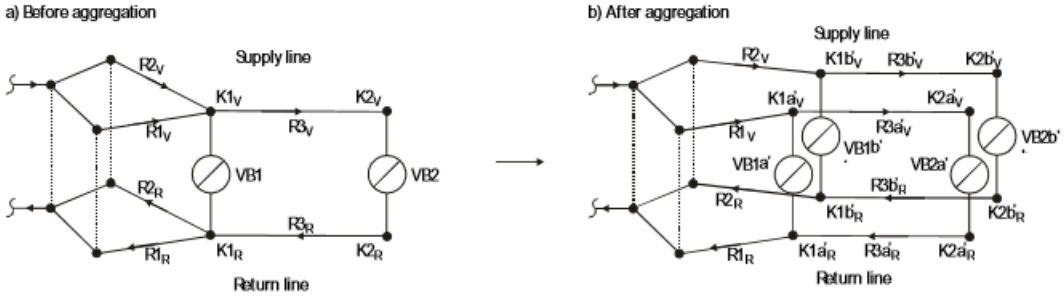
-Dall'equazione di miscelazione perfetta applicata al nodo K1R troviamo la temperatura di ingresso nel nodo K1R dal ramo R1R'

-La temperatura in uscita dal ramo R2V' è uguale a quella del nodo K3V (non ci sono miscelazioni)

-A questo punto conosciamo tutte le temperature di ingresso e di uscita dai rami R1V',R2V', R1R' e R2R' ; pertanto mediante l'equazione 3.5 troviamo le trasmittanze termiche k.

c) <u>Semplificazione dei circuiti chiusi (loop)</u>

Come mostrato nella figura sottostante, ogni elemento a valle dei loop viene duplicato e posizionato su due reti in parallelo.



(Figura 3.10)

Per evitare di creare troppi elementi e di complicare la rete, è opportuno effettuare questa semplificazione soltanto se a valle del loop si ha un solo ramo ed una sola utenza.
In caso contrario, prima di procedere con l'eliminazione del loop, è necessario semplificare la rete a valle del nodo K1V.
Una volta che il circuito chiuso è stato semplificato, ci si può ricondurre ad una struttura con un solo ramo mediante le due semplificazioni viste in precedenza.

3.3 MODELLO DI AGGREGAZIONE DANESE

In questo paragrafo viene presentato il metodo di aggregazione delle reti di teleriscaldamento utilizzato nel mio lavoro ed esposto nell'articolo scientifico riportato al riferimento [4] .
Data la volontà di concentrarsi sulle dispersioni termiche, il modello di aggregazione danese è stato preferito a quello tedesco.

Qui di seguito troviamo indicazioni su pedici, apici e variabili ausiliarie utilizzati in questa sezione.

APICI		PEDICI		VARIABILI AUSILIARIE	
s	mandata	0	nodo 0	α	m_2/m_1
r	ritorno	1	ramo o nodo 1	β	$1+\alpha$
w	acqua	2	ramo o nodo 2	φ	C^w/C^w+C^p
p	metallo tubazione	A	ramo o nodo A	ψ	φ_2/φ_1
		B	ramo o nodo B	γ	$V_1/V_2*\alpha$
		G	terreno	δ	$1-\gamma$
				θ_A^r	H_A^R/H_1^R
				θ_B^r	H_B^R/H_1^R
				θ_2^r	H_2^R/H_1^R
				θ_2^s	H_2^S/H_1^S

(Tabella 3.1)

Dove

- C è la capacità termica lineare in J/m/°C;
- H è il coefficiente di scambio termico in W/°C.
- V è il volume interno della tubazione

Come già accennato precedentemente vi sono due fasi di aggregazione:

- PRIMA FASE: SEMPLIFICAZIONE DELLE RAMIFICAZIONI
  Si passa da una struttura ramificata, che presenta dei nodi collegati con un numero di rami maggiore o uguale a tre, ad una struttura lineare in cui ogni nodo è collegato al più con due rami.
  Durante quest'operazione sia il numero dei rami che il numero dei nodi rimane costante.
- SECONDA FASE: COMPATTAZIONE
  Nella seconda fase si procede con l'aggregazone dei rami interni.

3.3.1 SEMPLIFICAZIONE DELLE RAMIFICAZIONI

L'obiettivo di questa prima fase è schematizzato nella figura sottostante: passare da una struttura ramificata (figura 3.11a) ad una struttura lineare equivalente (figura 3.11b).
Quest'ultima è composta da rami caratterizzati da diametri, lunghezze e coefficienti di scambio termico tali per cui la rete equivalente si comporta come quella originale ramificata.

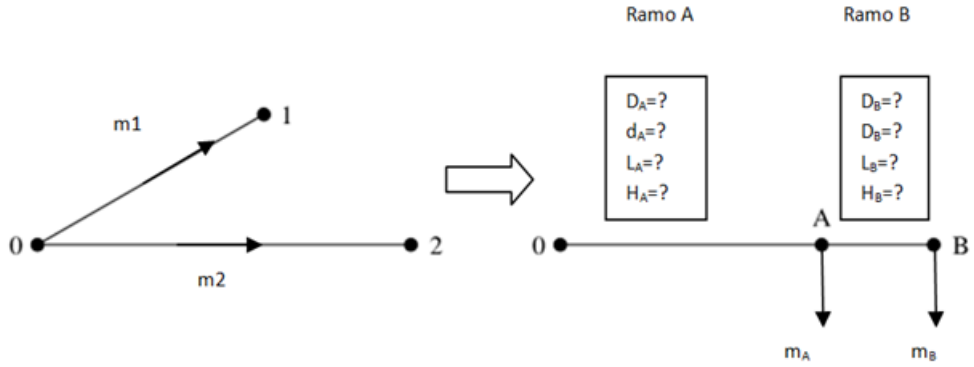


Figura 3.11a Figura 3.11b

Si richiede quindi che tra le due reti si conservi:

- portata massica
- volume d'acqua
- temperature dei nodi
- ritardo temporale del fronte di temperatura
- dissipazioni di calore

Come illustrato a breve, è possibile ottenere ciò soltanto con delle supposizioni alquanto forti sulle portate e sulle temperature di ritorno che non trovano riscontro nella realtà; tuttavia i risultati ottenuti dal modello equivalente si possono ritenere soddisfacenti.

<u>Lunghezze e diametri equivalenti</u>

Con riferimento ala figura 2.11, si assume che il ritardo temporale del fronte di temperatura del ramo 1 sia minore di quello del ramo 2 (se così non fosse basterebbe invertire il ramo 1 con il ramo 2) e per adempiere alle richieste sopra elencate si impone:

- il ritardo temporale nel ramo A è pari al ritardo temporale nel ramo 1
  $\tau_A = \tau_1$ (3.21)
- il ritardo temporale del ramo 2 è pari alla somma del ritardo temporale del ramo A con quello del ramo B. In questo modo il ritardo temporale totale si conserva
  $\begin{cases} \tau_2 = \tau_B + \tau_A \\ \tau_A = \tau_1 \end{cases} \rightarrow \tau_B = \tau_2 - \tau_1$ (3.22)
- il volume di acqua si conserva
  $V_A + V_B = V_1 + V_2$ (3.23)
- la portata massica si conserva (m denota la portata negli scambiatori, non quella nei rami)
  $m_A + m_B = m_1 + m_2$ (3.24)

- il carico termico totale, ovvero la potenza totale richiesta dalle utenze in [W], si conserva
  $P_A + P_B = P_1 + P_2$ (3.25)
- la potenza termica dissipata attraverso le tubazioni si conserva
  $Q_A + Q_B = Q_1 + Q_2$ (3.26)

Inoltre come punto di partenza scegliamo di metterci nelle seguenti condizioni:

- il ramo B è uguale alla parte del ramo 2 per cui $\tau > \tau_1$
  $d_B = d_2$ (diametro interno) (3.27)
  $D_B = D_2$ (diametro esterno) (3.28)
  $A_B = A_2$ (sezione interna della tubazione) (3.29)
  $\varphi_B = \varphi_2$ (3.30)
  $m_B = m_2$ (3.31)

  Dove φ è definito come $\frac{C^w + C^p}{C^p}$ con -C^w capacità termica lineare dell'acqua contenuta nella tubazione
  -C^p capacità termica lineare del metallo costituente la tubazione
- La velocità del fluido nel ramo A (ramo con portata maggiore nella rete lineare equivalente) è una media delle velocità del fluido nei rami 1 e 2 pesata sulle portate.

$v_A = \frac{m_1 * v_1 + m_2 * v_2}{m_1 + m_2}$ (3.32)

Partendo dalle assunzioni elencate fin'ora, vogliamo ricavare le restanti caratteristiche della rete lineare equivalente (con pedici A e B) in funzione delle caratteristiche della rete originale (con pedici 1 e 2).

Dalle equazioni 3.24 e 3.31 :

$m_A = m_1$ (3.33)

Ipotizzando che l'acqua e la tubazione abbiano la stessa temperatura, il ritardo temporale del fronte di temperatura può essere espresso con

$\tau = \frac{V * \rho^w}{m * \varphi}$ (3.34)

Dalle equazioni 3.31) e 3.34 segue la seguente espressione

$\frac{\rho^w * V_A}{(m_A + m_B) * \varphi_A} = \frac{\rho^w * V_1}{m_1 * \varphi_1}$ (3.35)

Inserendo le equazioni 3.31 e 3.33 ed utilizzando definizione di α in tabella 3.1:

$V_A = V_1 * (1 + \alpha) * \frac{\varphi_A}{\varphi_1}$ (3.36)

Allo stesso modo combinando le equazioni 3.32 e 3.34 abbiamo

$\frac{\rho^w * V_B}{m_B * \varphi_B} = \frac{\rho^w * V_2}{m_2 * \varphi_2} - \frac{\rho^w * V_1}{m_1 * \varphi_1}$ (3.37)

Usando le equazioni 3.30 e 3.31 e le definizioni delle variabili ausisliarie α e ψ (tabella 3.1) otteniamo

$V_B = V_2 - V_1 * \alpha * \psi$ (3.38)

Utilizzando ora la variabile ausiliare γ (vedere tabella 3.1) si ha

$V_B = V_2 * (1 - \gamma * \psi)$ (3.39)

Quindi considerando anche l'equazione 3.27 troviamo L_B

$$L_B = L_2 * (1 - \gamma * \psi) \quad (3.40)$$

Sostituendo l'espressione di V_B dell'equazione 3.38 nell'equazione 3.23 si ottiene

$$V_A = V_1 * (1 + \alpha * \psi) \quad (3.41)$$

Mettendo a sistema l'equazione 3.36 con la 3.41 in modo da eliminare V_A otteniamo

$$(1 + \alpha) * \frac{\varphi_A}{\varphi_1} = 1 + \alpha * \psi \quad (3.42)$$

Esplicitando φ_A si ha

$$\varphi_A = \frac{\varphi_1 + \alpha * \varphi_2}{1 + \alpha} \quad (3.43)$$

La velocità del fronte di temperatura può essere espressa con

$$v = \frac{\varphi * m}{\rho^w * A} \quad (3.44)$$

Combinando l'equazione 3.32) con la 3.44 e ricordando che la portata massica nel ramo A vale $m_A + m_B$ si giunge alla seguente espressione

$$\frac{\varphi_A * (m_A + m_B)}{A_A} = \frac{m_1 * \frac{\varphi_1 * m_1}{A_1} + m_2 * \frac{\varphi_2 * m_2}{A_2}}{m_1 + m_2} \quad (3.45)$$

Sostituendo le espressioni per φ_A e per $m_A + m_B$ usando le equazioni 3.24 e 3.43

$$\frac{\varphi_1 + \alpha * \varphi_2}{1 + \alpha} * \frac{(m_1 + m_2)}{A_A} = \frac{m_1 * \frac{\varphi_1 * m_1}{A_1} + m_2 * \frac{\varphi_2 * m_2}{A_2}}{m_1 + m_2} \quad (3.46)$$

Introducendo la variabile $\beta = 1 + \alpha$ ed esplicitando A_A

$$A_A = \frac{\beta * (1 + \alpha * \psi) * A_1 * A_2}{A_2 + A_1 * \alpha^2 * \psi} \quad (3.47)$$

Dall'equazione 3.41 possiamo scrivere

$$L_A * A_A = V_1 * (1 + \alpha * \psi) \quad (3.48)$$

Esplicitando L_A ed esprimendo A_A con l'equazione 3.47

$$L_A = \frac{V_1 * (1 + \alpha * \psi)}{\frac{\beta * (1 + \alpha * \psi) * A_1 * A_2}{A_2 + A_1 * \alpha^2 * \psi}} = L_1 * \frac{1 + \alpha^2 * \psi * \frac{A_1}{A_2}}{\beta} \quad (3.49)$$

Il diametro interno del ramo A può essere trovato quindi dalla sezione interna con

$$d_A = 2 * \sqrt{\frac{A_A}{\pi}} = 2 * \sqrt{\frac{\beta * (1 + \alpha * \psi) * A_1 * A_2}{(A_2 + A_1 * \alpha^2 * \psi) * \pi}} \quad (3.50)$$

Infine il diametro esterno D_A della tubazione viene trovato usando la definizione di φ_A

$$\varphi_A = \frac{C_A^W}{C_A^W + C_A^P} \quad (3.51)$$

Dove C_A^W e C_A^P sono rispettivamente la capacità termica lineare dell'acqua contenuta nella tubazione e la capacità termica lineare della tubazione stessa

$$C_A^W = \frac{1}{4} * \pi * d_A^2 * \rho^w * c^w \quad (3.52)$$

$$C_A^P = \frac{1}{4} * \pi * (D_A^2 - d_A^2) * \rho^p * c^p \quad (3.53)$$

Inserendo le espressioni di queste capacità termiche nell'equazione 3.51 ed esplicitando D_A si ha

$$D_A = d_A * \sqrt{1 + \frac{\rho^w * c^w}{\rho^p * c^p} * \frac{1 - \varphi_A}{\varphi_A}} \quad (3.54)$$

Abbiamo quindi trovato le dimensioni e il ritardo temporale del fronte di temperatura dei rami A e B della rete lineare equivalente.

Coefficienti di scambio termico equivalenti

Fino ad ora non abbiamo tenuto conto dell' isolamento delle tubazioni; pertanto in questo paragrafo verranno calcolati i coefficienti di dispersione termica della rete lineare equivalente in base all' isolamento della rete reale ramificata.
Come premessa è necessario chiarire che, siccome la variazione di temperatura a cavallo della tubazione in acciaio è minima, per calcolare le dispersioni è stata utilizzata la temperatura corrispondente al diametro interno.

Come primo passo si esprimono le temperature T_A e T_B della rete equivalente in funzione delle temperature T_1 e T_2 della rete originale.
Con riferimento alla figura 3.11, imponendo nei due sistemi la stessa dissipazione di calore si ha

$$(T^s - T_G)(H_A^s + H_B^s) + (T_{AB} - T_G) * H_A^r + (T_B^r - T_G) * H_B^r = (T^s - T_G)(H_1^s + H_2^s) + (T_1^r - T_G) * H_1^r + (T_2^r - T_G) * H_2^r \quad (3.55)$$

Dove T_A^r è la temperatura di ritorno dall'utenza A prima della miscelazione con l'acqua proveniente dal ramo B e T_{AB} è la temperatura nel nodo A una volta avvenuta la miscelazione.
Richiedendo che le reti di ritorno abbiano la stessa temperatura al nodo 0 ed utilizzando le variabili ausiliarie α e β (pari a 1+α) si ha

$$T_{AB} = \frac{m_A * T_A^r + m_B * T_B^r}{m_A + m_B} = \frac{m_1 * T_1^r + m_2 * T_2^r}{m_1 + m_2} = \frac{T_1^r + \alpha * T_2^r}{\beta} \quad (3.56)$$

Per quanto riguarda la mandata, scegliamo di imporre una dissipazione termica uguale e quindi si ottiene la seguente uguaglianza

$$H_A^s + H_B^s = H_1^s + H_2^s \quad (3.57)$$

Inserendo quest'equazione nella 3.55

$$(T_{AB} - T_G) * H_A^r + (T_B^r - T_G) * H_B^r = (T_1^r - T_G) * H_1^r + (T_2^r - T_G) * H_2^r \quad (3.58)$$

Inoltre considerando l'equazione 3.56 si ha

$$T_B^r * H_B^r = T_1^r * (H_1^r - H_A^r/\beta) + T_2^r * (H_2^r - H_A^r * \alpha/\beta) + T_G * (H_A^r + H_B^r - H_1^r - H_2^r) \quad (3.59)$$

A questo punto imponiamo che il carico termico totale della rete lineare equivalente sia uguale al carico termico totale della rete originale

$$c_p^w * (T^s - T_A^r) * m_A + c_p^w * (T^s - T_B^r) * m_B = c_p^w * (T^s - T_1^r) * m_1 + c_p^w * (T^s - T_2^r) * m_2 \quad (3.60)$$

Semplificando il calore specifico dell'acqua c_p^w e considerando che $m_A=m_1$ e $m_B=m_2$ abbiamo

$$T_A^r = T_1^r + (T_2^r - T_B^r) * \frac{m_1}{m_2} = T_1^r + (T_2^r - T_B^r) * \alpha \quad (3.61)$$

Combinando le equazioni 3.59 e 3.61 ed eliminando T_B^r otteniamo

$$T_A^r * H_B^r = T_1^r * \left(H_A^r * \frac{\alpha}{\beta} + H_B^r - H_1^r * \alpha\right) + T_2^r * \left(H_A^r * \frac{\alpha}{\beta} + H_B^r - H_2^r\right) * \alpha + T_G * (H_1^r + H_2^r - H_A^r - H_B^r) * \alpha \quad (3.62)$$

Utilizzando le variabili ausiliarie θ_A^r e θ_B^r le equazioni 3.62 e 3.59 possono essere espresse come

$$T_A^r = T_1^r * \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \alpha}{\theta_B^r} + T_2^r * \frac{\left(\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \theta_2^r\right) * \alpha}{\theta_B^r} + T_G * \frac{(1 + \theta_2^r - \theta_A^r - \theta_B^r) * \alpha}{\theta_B^r} \quad (3.63)$$

$$T_B^r = T_1^r * \frac{1-\theta_A^r/\beta}{\theta_B^r} + T_2^r * \frac{\theta_2^r - \theta_A^r * \frac{\alpha}{\beta}}{\theta_B^r} + T_G * \frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} \quad (3.64)$$

Si è quindi arrivati ad equazioni lineari in cui T_A^r e T_B^r sono espresse in funzione delle temperature della rete originale T_1^r e T_2^r.

Le equazioni 3.63 e 3.64 possono essere espresse nel seguente modo

$$T_A^r = C_{A1} * T_1^r + C_{A2} * T_2^r + C_{AG} * T_G \quad (3.65)$$

$$T_B^r = C_{B1} * T_1^r + C_{B2} * T_2^r + C_{BG} * T_G \quad (3.66)$$

Dove

$$C_{A1} = \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \alpha}{\theta_B^r} = -\alpha * C_{B1} + 1 \quad (3.67)$$

$$C_{A2} = \frac{\left(\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \theta_2^r\right) * \alpha}{\theta_B^r} = -\alpha * C_{B2} + \alpha \quad (3.68)$$

$$C_{AG} = \frac{(1 + \theta_2^r - \theta_A^r - \theta_B^r) * \alpha}{\theta_B^r} = -\alpha * C_{BG} \quad (3.69)$$

$$C_{B1} = \frac{1-\theta_A^r/\beta}{\theta_B^r} \quad (3.70)$$

$$C_{B2} = \frac{\theta_2^r - \theta_A^r * \frac{\alpha}{\beta}}{\theta_B^r} \quad (3.71)$$

$$C_{BG} = \frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} \quad (3.72)$$

Ora cerchiamo di esprimere i carichi termici della rete lineare equivalente (P_A e P_B) in funzione dei carichi termici della rete originale (P_1 e P_2)

$$P_1 = m_1 * c^w * (T^s - T_1^r) \quad (3.73)$$

$$\rightarrow T_1^r = T^s - \frac{P_1}{m_1 * c^w} \quad (3.74)$$

$$P_2 = m_2 * c^w * (T^s - T_2^r) \quad (3.75)$$

$$\rightarrow T_2^r = T^s - \frac{P_2}{m_2 * c^w} \quad (3.76)$$

$$P_B = m_B * c^w * (T^s - T_B^r) \quad (3.77)$$

Ora consideriamo l'equazione 3.77 e sostituiamo T_B^r con l'espressione 3.66. Sostituendo poi T_1^r e T_2^r con le espressioni 3.74 e 3.76 e sviluppando le espressioni dei coefficienti C si arriva a

$$P_B = P_1 * \frac{\alpha * (1 - \theta_A^r/\beta)}{\theta_B^r} + P_2 * \frac{\theta_2^r - \theta_A^r * \alpha/\beta}{\theta_B^r} + P_G * \frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} \quad (3.78)$$

Dove $P_G = c^w * m_2 * (T^s - T_G)$ è la potenza che sarebbe fornita al nodo 2 se la temperatura in uscita dall'utenza collegata al nodo 2 fosse T_G invece che T_2^r.

Imponendo che nel passaggio dalla rete originale a quella equivalente il carico totale si conservi ($P_A + P_B = P_1 + P_2$) ed utilizzando l'equazione 3.78 otteniamo:

$$P_A = P_1 * \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \alpha}{\theta_B^r} + P_2 * \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \theta_2^r}{\theta_B^r} - P_G * \frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} \quad (3.79)$$

Come abbiamo fatto per le temperature, anche per i carichi possiamo scrivere le potenze trovate nel seguente modo

$$P_A = K_{A1} * P_1 + K_{A2} * P_2 + K_{AG} * P_G \quad (3.80)$$

$$P_B = K_{B1} * P_1 + K_{B2} * P_2 + K_{BG} * P_G \quad (3.81)$$

Dove

$$K_{A1} = \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \alpha}{\theta_B^r} = C_{A1} = 1 - \alpha * C_{B1} \quad (3.82)$$

$$K_{A2} = \frac{\theta_A^r * \frac{\alpha}{\beta} + \theta_B^r - \theta_2^r}{\theta_B^r} = C_{A2}/\alpha = 1 - C_{B2} \quad (3.83)$$

$$K_{AG} = -\frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} = C_{BG} \quad (3.84)$$

$$K_{B1} = \frac{\alpha * (1 - \theta_A^r/\beta)}{\theta_B^r} = \alpha * C_{B1} \quad (3.85)$$

$$K_{B2} = \frac{\theta_2^r - \theta_A^r * \alpha/\beta}{\theta_B^r} = C_{B2} \quad (3.86)$$

$$K_{BG} = \frac{\theta_A^r + \theta_B^r - \theta_2^r - 1}{\theta_B^r} = C_{BG} \quad (3.87)$$

Ora il problema consiste nell'assegnazione dei valori corretti ai vari θ o H.
Prima di tutto bisogna notare che i nodi 1 e 2 in generale rappresentano delle sotto-ramificazioni della rete. Per evitare quindi di dover dividere ogni sotto-ramificazione nella rete originale in due soto-ramidicazioni equivalenti, si assegna al nodo A la sotto-ramificazione connessa al nodo 1 ed al nodo B la sotto-ramificazione connessa al nodo 2.
Imponiamo quindi $P_A=P_1$, $P_B=P_2$, $T_A^r=T_1^r$ e $T_B^r=T_2^r$ modificando i coefficienti C e K come segue

$$C_{A1} = C_{A1} = K_{A1} = K_{B2} = 1 \quad (3.88)$$

$$C_{A2} = C_{B1} = K_{A2} = K_{B1} = 0 \quad (3.89)$$

Guardando le equazioni 3.67-3.72 e 3.84-3.87 si ha che le equazioni 3.88 e 3.89 sono verificate se e solo se

$$\theta_A^r = 1 + \alpha \leftrightarrow H_A^r = (1 + \alpha) * H_1^r \quad (3.90)$$

$$\theta_B^r = \theta_2^r - \alpha \leftrightarrow H_B^r = H_2^r - \alpha * H_1^r \quad (3.91)$$

Queste due equazioni implicano che

$$\theta_A^r + \theta_B^r = \theta_2^r + 1 \leftrightarrow H_A^r + H_B^r = H_1^r + H_2^r \quad (3.92)$$

Combinando ora l'equazione 3.92 con le equazioni 3.84 e 3.87 si ha

$$C_{AG} = C_{BG} = K_{AG} = K_{BG} = 0 \quad (3.93)$$

Passiamo ora all'analisi della mandata.
Considerando l'equazione 3.57 scegliamo di suddividere i coefficienti di scambio termico H tra A e B allo stesso modo di come è stato fatto per il ritorno; in questo modo otteniamo le stesse formule per la mandata e per il ritorno.

$$H_A^S = (1 + \alpha) * H_1^S \quad (3.94)$$

$$H_B^S = H_2^S - \alpha * H_1^S \quad (3.95)$$

Grazie a questa scelta ,quindi, se $H_1^S=H_1^r$ e $H_2^S=H_2^r$ allora $H_A^S=H_A^r$ e $H_B^S=H_B^r$.
Poichè i coefficienti di scambio termico non possono essere negativi, le equazioni 3.94 e 3.95 sono valide solo se

$$\theta_2^S \geq \alpha \leftrightarrow \frac{H_2^S}{H_1^S} \geq \frac{m_1}{m_2} \quad (3.96)$$

Allo stesso modo per la rete di ritorno le equazioni 3.90 e 3.91 sono valide solo se

$$\theta_2^r \geq \alpha \leftrightarrow \frac{H_2^r}{H_1^r} \geq \frac{m_1}{m_2} \quad (3.97)$$

Se invece $\theta_2^s<\alpha$ oppure $\theta_2^r<\alpha$ dobbiamo imporre un' ulteriore assunzione: il carico e la temperatura di ritorno nei nodi A e B sono uguali ai carichi e alle temperature di ritorno dei nodi 1 e 2

$$P_A = P_1 \qquad P_B = P_2 \quad (3.98)$$

$$T_A^r = T_1^r \qquad T_B^r = T_2^r \quad (3.99)$$

$$h_B^r = h_2^r \qquad h_B^s = h_2^s \quad (3.100)$$

Cominciando dalla rete di ritorno, dalle equazioni 3.40 e 3.100 troviamo

$$H_B^r = H_2^r * (1 - \gamma * \psi) \leftrightarrow \theta_B^r = \theta_2^r * (1 - \gamma * \psi) \quad (3.101)$$

La conservazione delle dissipazioni termiche tra la rete ramificata originale e quella lineare equivalente si può esprimere con

$$(T_{AB} - T_G) * H_A^r + (T_B^r - T_G) * H_B^r = (T_1^r - T_G) * H_1^r + (T_2^r - T_G) * H_2^r \quad (3.102)$$

Per semplificare la formula inseriamo nell'equazione 3.102 le seguenti temperature relative ottenendo l'equazione 3.107

$$\Delta T_1 = T_1^r - T_G \quad (3.103)$$

$$\Delta T_2 = T_2^r - T_G \quad (3.104)$$

$$\Delta T_B = T_B^r - T_G = T_2^r - T_G = \Delta T_2 \quad (3.105)$$

$$\Delta T_{AB} = T_{AB} - T_G = (\Delta T_1 + \alpha * \Delta T_2)/\beta \quad (3.106)$$

$$\frac{\Delta T_1 + \alpha * \Delta T_2}{\beta} * H_A^r + \Delta T_2 * H_B^r = \Delta T_1 * H_1^r + \Delta T_2 * H_2^r \quad (3.107)$$

Sostituendo l'espressione di H_B^r da 3.101 nell'equazione 3.107 ed esplicitando H_A^r

$$H_A^r = \frac{(\Delta T_1 * H_1^r + \Delta T_2 * H_2^r * \gamma * \psi) * \beta}{\Delta T_1 + \alpha * \Delta T_2} \quad (3.108)$$

Introducendo la variabile Ώ definita come

$$Ώ = \frac{\Delta T_2}{\Delta T_1} = \frac{T_2^r - T_G}{T_1^r - T_G} \quad (3.109)$$

H_A^r può essere scritta come

$$H_A^r = \frac{(H_1^r + H_2^r * \psi * \gamma * Ώ) * (1 + \alpha)}{1 + \alpha * Ώ} \leftrightarrow \theta_A^r = \frac{(1 + \theta_2^r * \psi * \gamma * Ώ) * (1 + \alpha)}{1 + \alpha * Ώ} \quad (3.110)$$

Ora passiamo alla rete di mandata, dalle equazioni 3.40 e 3.100 troviamo

$$H_B^s = H_2^s * (1 - \gamma * \psi) \quad (3.111)$$

Inoltre, tenendo conto dell'equazione 3.57 abbiamo

$$H_A^s = H_1^s + \gamma * \psi * H_2^s \quad (3.112)$$

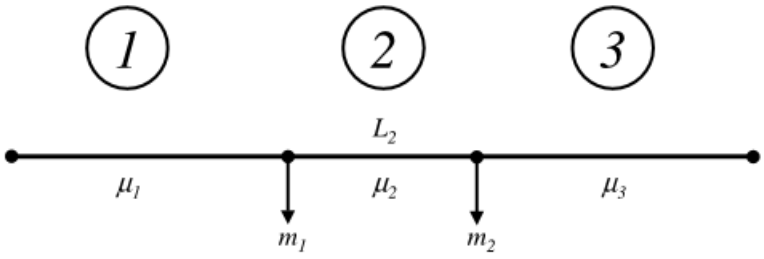
Effettuando l'ulteriore assunzione in cui $T_1^r=T_2^r$ otteniamo

$$H_A^s = H_1^s + \gamma * \psi * H_2^s \qquad H_B^s = H_2^s * (1 - \gamma * \psi) \quad (3.113)$$

$$H_A^r = H_1^r + \gamma * \psi * H_2^r \qquad H_B^r = H_2^r * (1 - \gamma * \psi) \quad (3.114)$$

3.3.2 COMPATTAZIONE

A questo punto abbiamo ottenuto una rete equivalente lineare in cui molti rami sono particolarmente corti rispetto a quelli più lunghi.
Seguendo il metodo che mostreremo, questi rami (e quindi anche i relativi nodi) possono essere inglobati dai rami (e nodi) adiacenti in modo da semplificare ulteriormente la rete.
Applicando più volte questa semplificazione è possibile arrivare ad una rete formata da due soli rami.
Prendiamo in considerazione la seguente configurazione con un ramo (il ramo 2) più corto rispetto a quelli adiacenti (rami 1 e 3).



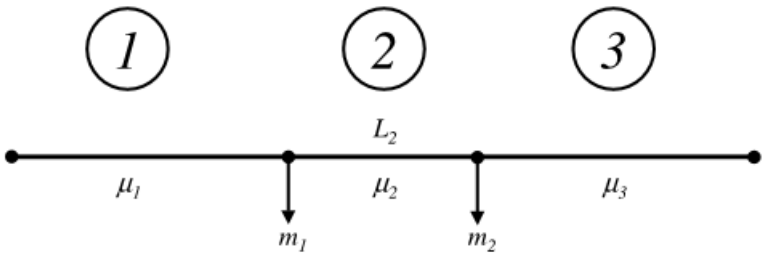
μ indica la portata nel ramo
m indica la portata alle utenze

(Figura 3.12)

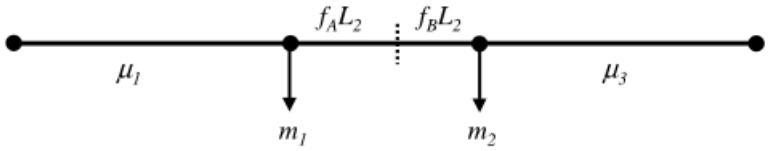
Come mostrato nella figura sottostante vogliamo dividere il ramo 2 in due parti e aggiungere queste al ramo 1 e al ramo 3 ottenendo quindi una configurazione aggregata equivalente formata da due rami:

-il ramo A composto dal ramo 1 e una parte del ramo 2
-il ramo B composto dal ramo 3 e l'altra parte del ramo 2

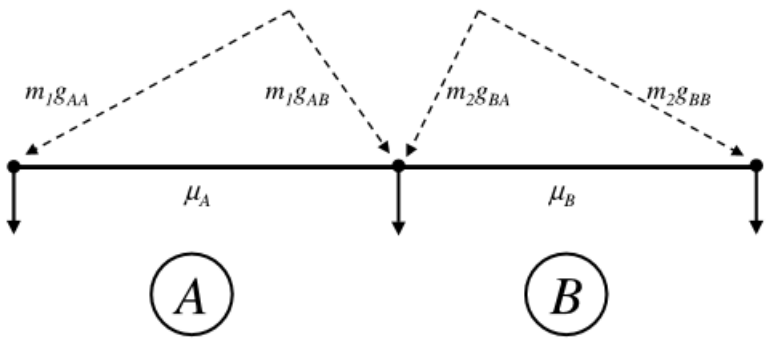
m1 è la portata all'utenza 1
m2 è la portata all'utenza 2



(Figura 3.13a)



(Figura 3.13b)



(Figura 3.13c)

Guardando la figura 3.13a risulta evidente che

$$m_1 = \mu_1 - \mu_2 \quad (3.115)$$

$$m_2 = \mu_2 - \mu_3 \quad (3.116)$$

La lunghezza del ramo 2 viene distribuita ai rami adiacenti in base ai coefficienti f_A e f_B rappresentantanti la frazione di massa estratta in un nodo rispetto a quella estratta totale.

$$f_A = \frac{m_2}{m_1+m_2} \quad (3.117)$$

$$f_B = \frac{m_1}{m_1+m_2} = 1 - f_A \quad (3.118)$$

In questo modo se l'utenza 1 richiede una portata molto maggiore rispetto a quella richiesta dall'utenza 2 il ramo 2 viene aggregato quasi totalmente al ramo B e la lunghezza del ramo A sarà simile a quella del ramo 1.

Anche le portate alle utenze 1 e 2 vengono suddivise (vedi figura 3.13b e 3.13c).
I coefficienti g_{AA} e g_{AB} indicano la suddivisione della portata m1 tra i nodi del ramo A mentre g_{BA} e g_{BB} indicano la suddivisione della portata m2 tra i nodi del ramo B.

$$g_{AA} = \frac{f_A*\tau_2}{\tau_1+f_A*\tau_2} \qquad g_{AB} = \frac{\tau_1}{\tau_1+f_A*\tau_2} \quad (3.119)$$

$$g_{BB} = \frac{f_B*\tau_2}{\tau_3+f_B*\tau_2} \qquad g_{BA} = \frac{\tau_3}{\tau_3+f_B*\tau_2} \quad (3.120)$$

Con

$$g_{AA} + g_{AB} = g_{BB} + g_{BA} = 1 \quad (3.121)$$

In questo modo la maggior parte delle portate ai carichi viene assegnata ai nodi che sulla rete equivalente lineare sono più vicini.
Le portate nei rami A e B possono essere espresse con

$$\mu_A = \mu_1 - g_{AA} * m_1 \quad (3.122)$$

$$\mu_B = \mu_2 - g_{BA} * m_2 \quad (3.123)$$

Per trovare le caratteristiche dei rami A e B rispettivamente dai dati dei rami 1-2 e 2-3 imponiamo che la lunghezza totale, le dispersioni termiche totali e il ritardo del fronte di temperatura si preservino tra le due reti.
La lunghezza del ramo A si trova con:

$$L_A = L_1 + f_A * L_2 \quad (3.124)$$

In generale il ritardo del fronte di temperatura è espresso dalla seguente equazione

$$\tau = \frac{\rho^w*V}{\mu*\varphi} \quad (3.125)$$

Quindi relativamente al ramo A

$$\tau_A = \tau_1 + f_A * \tau_2 \leftrightarrow \frac{V_A}{\mu_A*\varphi_A} = \frac{V_1}{\mu_1*\varphi_1} + \frac{f_A*V_2}{\mu_2*\varphi_2} \leftrightarrow V_A = V_1 * \frac{\mu_A}{\mu_1} * \frac{\varphi_A}{\varphi_1} + f_A * V_2 * \frac{\mu_A}{\mu_2} * \frac{\varphi_A}{\varphi_2} \quad (3.126)$$

Eccetto φ_A,tutti i termini dell'equazione precedente sono conosciuti; in generale φ si esprime con

$$\varphi = \frac{C^w}{C^w+C^p} = \frac{1}{1+\frac{C^p}{C^w}} = \frac{1}{1+X*V^p/V} \quad (3.127)$$

Dove $X=\rho^p*C^p/(\rho^w*C^w)$.
Applicando l' equazione 3.127 al ramo A si ottiene:

$$\varphi_A = \frac{1}{1+X*\frac{V_1^p+f_A*V_2^p}{V_1+f_A*V_2}} \quad (3.128)$$

Conoscendo quindi φ_A abbiamo tutti i dati per trovare anche V_A attraverso l'equazione 3.126.
Applichiamo ora l'equazione 3.127 al ramo A

$$\varphi_A = \frac{1}{1+X*V_A^p/V_A} \quad (3.129)$$

Esplicitiamo il volume del metallo della tubazione del ramo A

$$V_A^p = V_A * \frac{(1-\varphi_A)}{\varphi_A * X} \quad (3.130)$$

Conoscendo V_A e V_A^p possiamo quindi calcolare il diametro interno e quello esterno

$$d_A = 2 * \sqrt{\frac{V_A}{\pi * L_A}} \qquad D_A = 2 * \sqrt{\frac{V_A + V_A^p}{\pi * L_A}} \quad (3.131)$$

Infine calcoliamo il coefficiente di scambio termico per metro di lunghezza

$$h_A = \frac{h_1 * L_1 + f_A * h_2 * L_2}{L_A} \quad (3.132)$$

Per trovare le grandezze riferite al ramo B si può seguire la procedura utilizzata per il ramo A

3.3.3 RIASSUNTO PROCEDURA

Qui di seguito possiamo trovare dei prospetti riassuntivi relativi ai calcoli da effettuare una volta dati tutti i parametri relativi alla rete originale

a) <u>Semplificazione delle ramificazioni</u>

-Lunghezze e diametri equivalenti

$$m_A = m_1$$
$$V_A = V_1 * (1 + \alpha * \psi)$$
$$L_A = \frac{L_1 * \left(1 + \alpha^2 * \psi * \frac{A_1}{A_2}\right)}{\beta}$$
$$d_A = 2 * \sqrt{\frac{\beta * (1 + \alpha * \psi) * A_1 * A_2}{(A_2 + A_1 * \alpha^2 * \psi) * \pi}}$$
$$\varphi_A = \frac{\varphi_1 + \alpha * \varphi_2}{1 + \alpha}$$
$$D_A = d_A * \sqrt{1 + \frac{\rho^w * c^w}{\rho^p * c^p} * \frac{1 - \varphi_A}{\varphi_A}}$$
$$d_B = d_2$$
$$D_B = D_2$$
$$L_B = L_2 * (1 - \gamma * \psi)$$
$$V_B = V_2 * (1 - \gamma * \psi)$$
$$\varphi_B = \varphi_2$$

-Coefficienti di scambio termico

Per la rete di mandata:

- Se $\theta_2^s >= \alpha$, assumendo α costante:
  $H_A^s = (1 + \alpha) * H_1^s$
  $H_B^s = H_2^s - \alpha * H_1^s$
- Se $\theta_2^s < \alpha$, assumendo α costante:
  $H_A^s = H_1^s + \gamma * \psi * H_2^s$
  $H_B^s = H_2^s * (1 - \gamma * \psi)$

Per la rete di ritorno invece:

- Se $\theta_2^r >= \alpha$, assumendo α costante:
  $H_A^r = (1 + \alpha) * H_1^r$
  $H_B^r = H_2^r - \alpha * H_1^r$
- Se $\theta_2^r < \alpha$, assumendo α costante e $T_1^r = T_2^r$:
  $H_A^r = H_1^r + \gamma * \psi * H_2^r$
  $H_B^r = H_2^r * (1 - \gamma * \psi)$
- Se $\theta_2^r < \alpha$, assumendo $Ω = (T_2^r - T_G)/(T_1^r - T_G)$ costante:
  $H_A^r = \frac{(H_1^r + H_2^r * \psi * \gamma * Ω) * (1 + \alpha)}{1 + \alpha * Ω}$
  $H_B^r = H_2^r * (1 - \gamma * \psi)$

Compattazione

Prima di tutto bisogna calcolare:

$m_1 = \mu_1 - \mu_2$ $\qquad$ $m_2 = \mu_2 - \mu_3$

$f_A = \frac{m_2}{m_1+m_2}$ $\qquad$ $f_B = \frac{m_1}{m_1+m_2} = 1 - f_A$

$g_{AA} = \frac{f_A * \tau_2}{\tau_1 + f_A * \tau_2}$ $\qquad$ $g_{BA} = \frac{\tau_3}{\tau_3 + f_B * \tau_2}$

$\mu_A = \mu_1 - g_{AA} * m_1$ $\qquad$ $\mu_B = \mu_2 - g_{BA} * m_2$

A questo punto possiamo calcolare tutte le caratteristiche dei rami A e B

$L_A = L_1 + f_A * L_2$ $\qquad$ $L_B = L_3 + f_B * L_2$

$\varphi_A = \frac{1}{1+X*\frac{V_1^p + f_A * V_2^p}{V_1 + f_A * V_2}}$ $\qquad$ $\varphi_B = \frac{1}{1+X*\frac{V_3^p + f_B * V_2^p}{V_3 + f_B * V_2}}$

$V_A = V_1 * \frac{\mu_A}{\mu_1} * \frac{\varphi_A}{\varphi_1} + f_A * V_2 * \frac{\mu_A}{\mu_2} * \frac{\varphi_A}{\varphi_2}$ $\qquad$ $V_B = V_3 * \frac{\mu_B}{\mu_3} * \frac{\varphi_B}{\varphi_3} + f_B * V_2 * \frac{\mu_B}{\mu_2} * \frac{\varphi_B}{\varphi_2}$

$V_A^p = V_A * \frac{(1-\varphi_A)}{\varphi_A * X}$ $\qquad$ $V_B^p = V_B * \frac{(1-\varphi_B)}{\varphi_B * X}$

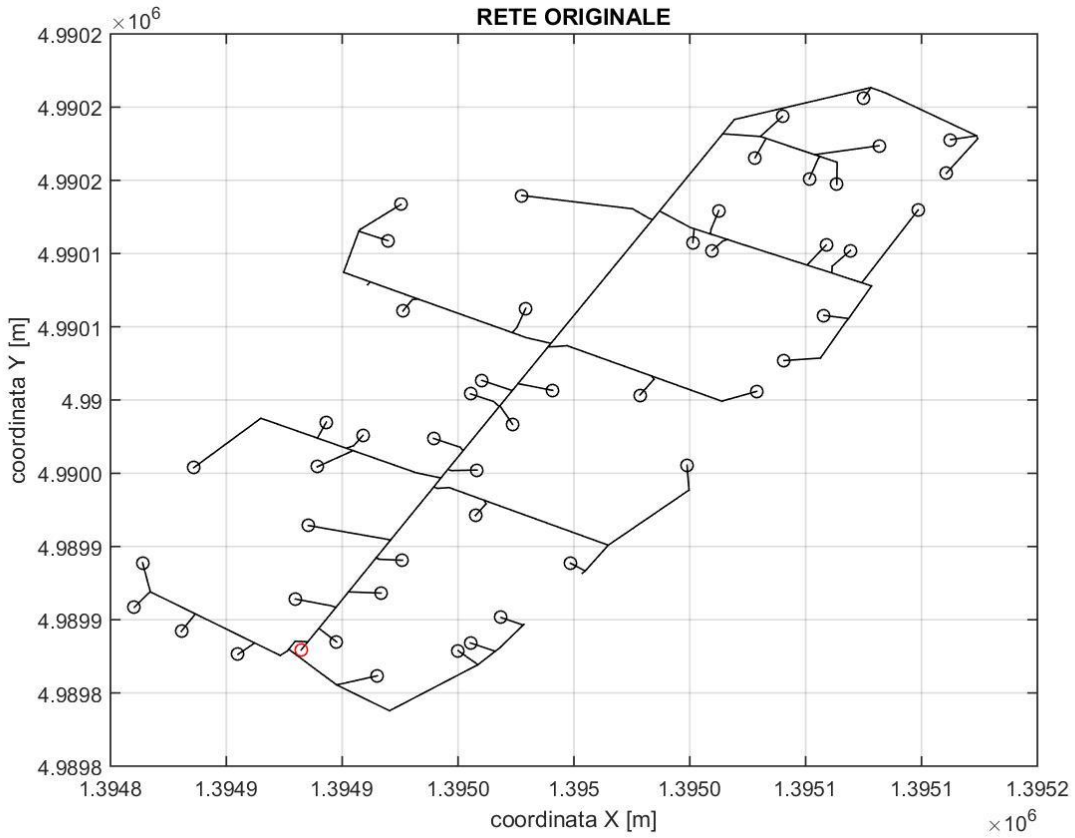
$d_A = 2 * \sqrt{\frac{V_A}{\pi * L_A}}$ $\qquad$ $d_B = 2 * \sqrt{\frac{V_B}{\pi * L_B}}$

$D_A = 2 * \sqrt{\frac{V_A + V_A^p}{\pi * L_A}}$ $\qquad$ $D_B = 2 * \sqrt{\frac{V_B + V_B^p}{\pi * L_B}}$

$h_A = \frac{h_1 * L_1 + f_A * h_2 * L_2}{L_A}$ $\qquad$ $h_B = \frac{h_3 * L_3 + f_B * h_2 * L_2}{L_B}$

4 APPLICAZIONE ALLA RETE BCT408

Il modello esposto nel capitolo precedente è stato testato sulla rete di distribuzione BCT408 raffigurata nella seguente mappa.



(Figura 4.1)

Il pallino rosso corrisponde al baricentro mentre i pallini neri indicano le utenze.
Sono state quindi effettuate delle simulazioni sia sulla rete originale che sulle reti semplificate equivalenti ed in seguito sono stati confrontati i risultati ottenuti.
Come possiamo osservare dalla mappa della rete non vi sono loop, di conseguenza il problema fluidodinamico può essere risolto utilizzando soltanto l'equazione di conservazione della massa 2.13 o 2.37 a seconda delle condizioni stazionarie o transitorie.

I dati raccolti dall' IREN per lo svolgimento delle simulazioni relative alla rete originale e per la costruzione del modello compatto sono qui sotto elencati:

- profilo dei carichi delle varie utenze durante una giornata;
- portate in uscita dalla rete di distribuzione verso le sottostazioni;
- nodo corrispondente al baricentro e nodi corrispondenti alle utenze;
- coordinate dei nodi della rete;
- diametri e lunghezze dei condotti.

Per quanto concerne il coefficiente di scambio termico lineare, al fine di mettere in evidenza le dispersioni altrimenti trascurabili, è stato utilizzato un valore di molto superiore rispetto a quelo reale pari a 0.06 W/m/K.

Sono inoltre stati ipotizzati i seguenti dati necessari per le simulazioni:

- temperatura del terreno T_G=10 °C=283.15 K
- temperatura del fluido in ingresso nel baricentro $T_{BC,in}$=120 °C=393.15 K

Nella semplificazione della rete originale, come abbiamo visto dal capitolo 3, le dimensioni dei condotti della rete equivalente dipendono dalle portate le quali però variano nel tempo in base all'andamento dei carichi richiesti dalle varie utenze.
Siccome l'obiettivo della tesi è la costruzione di un modello di compattazione che riduca il costo computazionale, non è opportuno applicare tale modello ad ogni step temporale considerato. In questo modo, infatti, i tempi di calcolo aumenterebbero invece di diminuire. Di conseguenza si è scelto di semplificare la rete utilizzando le portate relative ad un unico istante di tempo: quello in cui si ha il carico totale massimo.

Questa scelta ci ha permesso di :

- tener conto del contributo di tutte le utenze che, nel giorno a cui si riferiscono i dati, richiedono calore;
- minimizzare l'errore in condizioni di carico totale massimo.
  Risulta infatti naturale supporre che il modello commetta un errore minore nell'istante di tempo in cui le portate, che effettivamente scorrono nei condotti della rete originale, corrispondono a quelle considerate per la semplificazione della stessa.

4.1 PROBLEMI INCONTRATI E SOLUZIONI ADOTTATE

4.1.1 VALORI DELLE DENSITÀ E DEL CALORE SPECIFICO

La densità ed il calore specifico dipendono dalla temperatura.
Entrambe le grandezze sono presenti nell'equazione di risoluzione del problema termico in condizioni transitorie, per cui nella risoluzione dell'equazione 2.84 bisognerebbe utilizzare un metodo iterativo.
Considerando che questo aumenterebbe di molto il costo computazionale della risoluzione del problema termico e che la temperatura dell'acqua durante il funzionamento dell'impianto non varia molto, è stata presa la decisione di considerare due valori di densità costanti ed un valore di calore specifico anch'esso costante:

- $\rho_{w,r} = \rho_w(T=60\ °C) = 983.3\ kg/m^3$ per la rete di ritorno;
- $\rho_{w,m} = \rho_w(T=120\ °C) = 951.2\ kg/m^3$ per la rete di mandata;
- $c_{p,w} = 4186\ J/Kg/K$ sia per la mandata che per il ritorno.

4.1.2 PRESENZA DI RAMI CON PORTATA SEMPRE NULLA

Come è possibile osservare nella figura 4.1, vi sono dei rami in uscita dalla rete che non portano a delle utenze; inoltre vi sono delle utenze che non richiedono mai potenza nella giornata di cui ci sono stati forniti i dati.
Di conseguenza ci sono dei condotti in cui la portata è nulla anche in condizioni di carico massimo; ciò genera dei problemi sia per il calcolo delle temperature nella rete originale in condizioni stazionarie che per il processo di compattazione.

Per quanto riguarda il calcolo delle temperature, il problema si ha nell'applicazione della condizione al contorno ad un nodo corrispondente ad un'utenza spenta.
Consideriamo ad esempio la figura 2.10 e poniamoci in una situazione in cui la portata G_{j+3} è nulla in quanto l'utenza i+3 non richiede calore.
Applicando infatti l'equazione 2.72 si ha:

$$0 = \left(T_{j+3} - T_G\right) \qquad (4.1)$$

L'equazione 4.1 in sostanza ci dice che il ramo j+3 è in equilibrio termico con il terreno.
Se ora eguagliamo la temperatura del ramo j+3 alla media delle temperature dei nodi ai suoi capi otteniamo

$$0 = \left(\frac{T_{i+3}+T_i}{2} - T_G\right) \qquad (4.2)$$

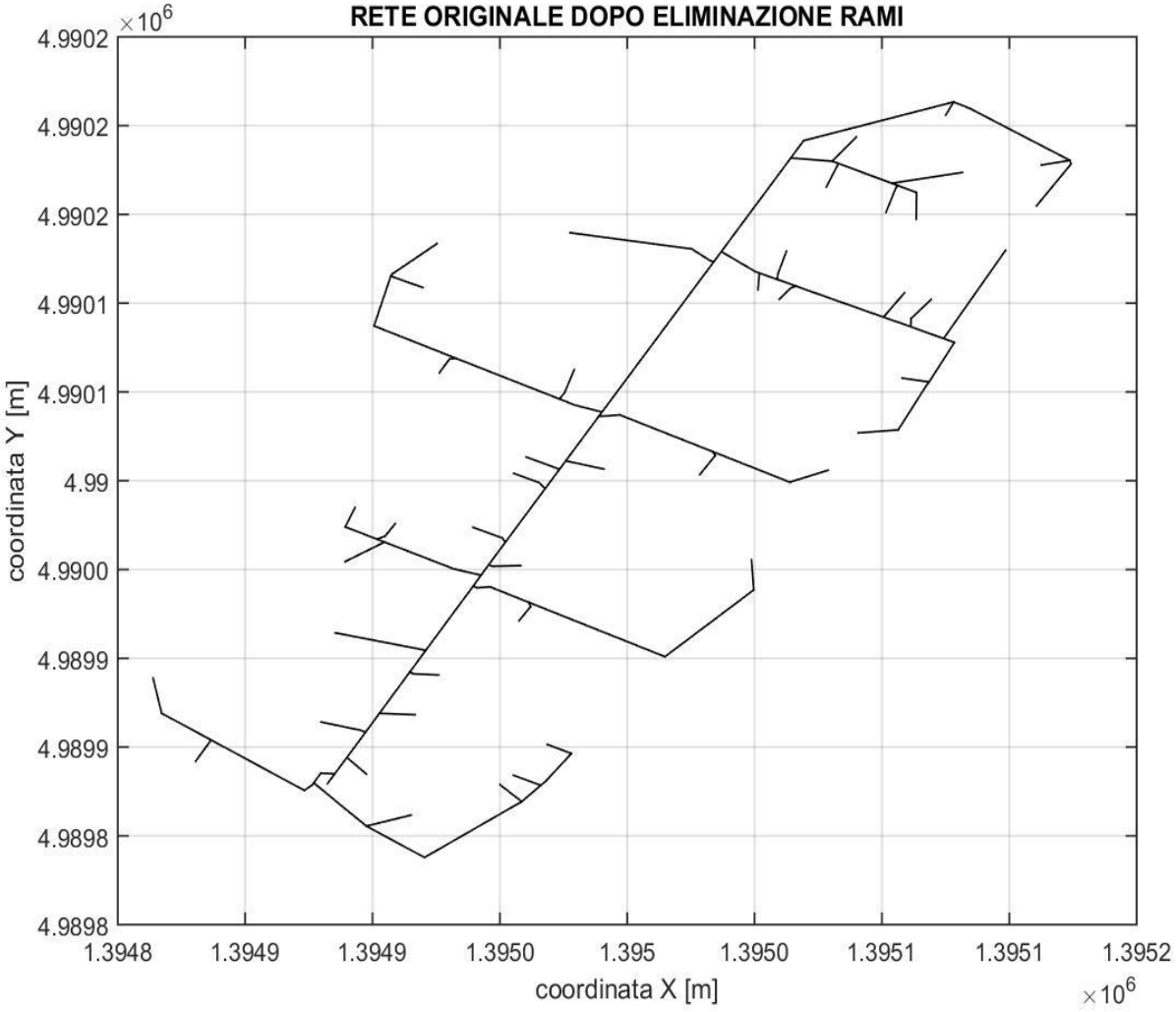
Dove T_i corrisponde ad una temperatura elevata in quanto l'utenza i+1 richiede calore e quindi nei rami j+1 e j si ha una portata non nulla.
Di conseguenza ,affinché l'equazione 4.2 sia verificata, T_{i+3} deve essere molto più piccola della temperatura del terreno TG; si ottiene dunque un risultato che non ha senso in quanto va contro il secondo principio della termodinamica.
Per risolvere questo problema si approssima la temperatura del ramo j+3 alla temperatura del nodo i+3; dall'equazione 4.1 si ottiene quindi:

$$0 = (T_{i+3} - T_G) \rightarrow T_{i+3} = T_G \qquad (4.3)$$

L'annullamento della portata G_{j+3} crea problemi anche nel processo di compattazione in quanto il ritardo temporale del fronte di temperatura nel ramo j è dato dalla seguente espressione, avente la portata al denominatore

$$\tau_j = \frac{V_j * \rho}{\varphi_j * G_j} \qquad (4.4)$$

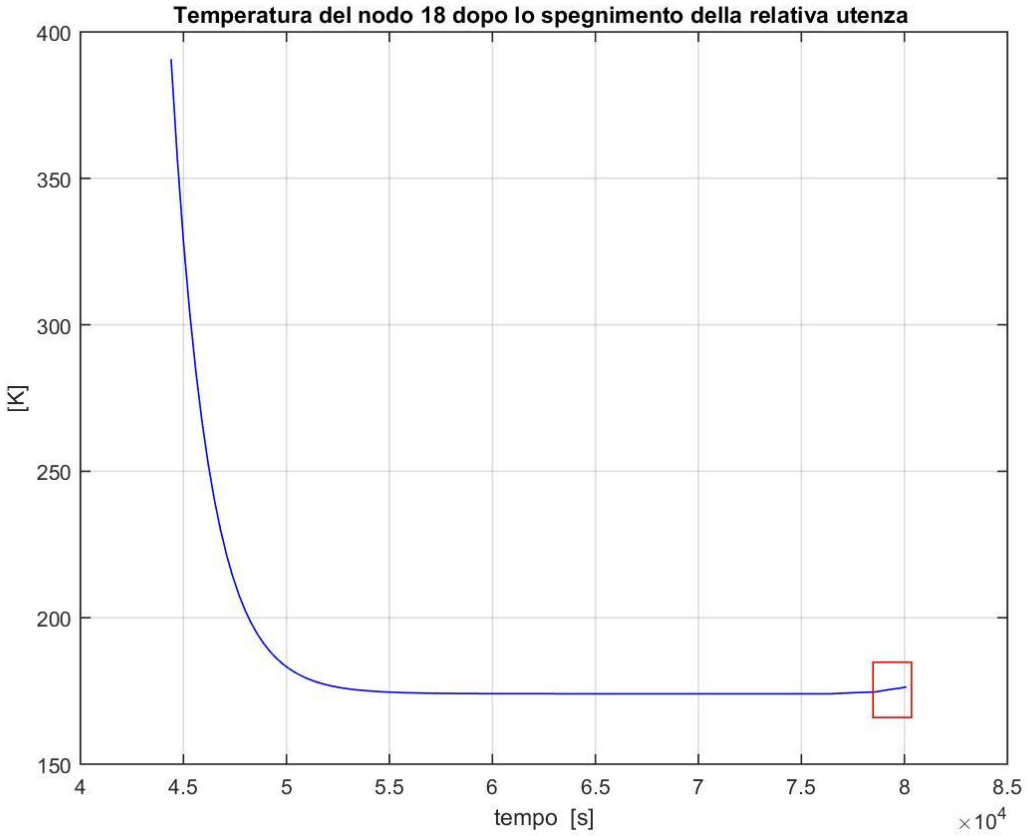
Di conseguenza il calcolatore restituisce il risultato di NaN (not a number) per τ.
Questo risultato non può essere ignorato in quanto τ è un parametro fondamentale per la compattazione della rete.
Per risolvere questo problema si è quindi deciso di utilizzare come rete di input al modello di compattazione quella ottenuta dopo l'eliminazione di tutti i rami con portata sempre nulla.


RETE ORIGINALE DOPO ELIMINAZIONE RAMI
×10^6
4.9902
4.9902
4.9902
4.9901
4.9901
4.99
4.9900
4.9899
4.9899
4.9898
4.9898
coordinata Y [m]
1.3948
1.3949
1.3949
1.3950
1.395
1.3950
1.3951
1.3951
1.3952
coordinata X [m]
×10^6

(Figura 4.2)

4.1.3 PRESENZA DI RAMI CON PORTATA NULLA NEL TRANSITORIO

I profili di carico delle varie utenze sono piuttosto variegati, pertanto ci sono delle sottostazioni il cui rifornimento di acqua calda viene interrotto in un istante di tempo t molto precedente rispetto a quello relativo ad altre sottostazioni.
Applicando quindi l'equazione 2.94 nell'imposizione della BC ai nodi corrispondenti a queste utenze che smettono di richiedere potenza , si incorre in un problema simile a quello descritto nel paragrafo precedente: la portata nel ramo che serve l'utenza è nulla e quindi ,nel caso in cui il suddetto nodo di ingresso sia attraversato dall'acqua, la curva che descrive l'andamento di temperatura del nodo nel tempo decresce in modo molto veloce e tende ad un valore molto inferiore rispetto alla temperatura del terreno TG supposta pari a 298.15 K.
Qui di seguito troviamo un grafico che descrive l'andamento della temperatura del nodo 18 che si ottiene utilizzando l'equazione 2.94



(Figura 4.3)
Come possiamo osservare la temperatura decresce fino ad arrivare ad un valore molto inferiore rispetto alla temperatura del terreno.
Nella parte riquadrata in rosso vi è un moderato aumento di temperatura dovuto allo spegnimento di tutte le utenze a valle del nodo di ingresso al ramo collegato all'utenza 18.
Siccome l'utenza allacciata al nodo 18 è spenta da molto tempo, è lecito supporre una condizione di stazionarietà; indicando inoltre il nodo 18 con il pedice i+3 e il nodo di ingresso al ramo collegato al nodo 18 con il pedice i, possiamo utilizzare l'equazione 4.2. Da questa si evince che se

la temperatura T_i diminuisce, affinché la media tra T_{i+3} e la stessa T_i sia pari alla temperatura del terreno T_G(costante), T_{i+3} deve aumentare.
L'andamento della temperatura presente nel grafico della figura 4.3 non è accettabile in quanto il terreno, assorbendo il calore uscente dalle tubazioni, per il secondo principio della termodinamica, deve avere una temperatura minore di quella dei condotti.
Di conseguenza bisogna modificare l'equazione relativa alla BC per tutti i nodi ai quali si allacciano le utenze spente. In particolare, come già fato per il caso stazionario, è stata cambiata la definizione della temperatura del ramo collegato all'utenza: con riferimento alla condizione al contorno per il nodo i+3 della figura 2.10 si ha

$$\overline{T_{j+3}} = T_{i+3} \qquad (4.5)$$

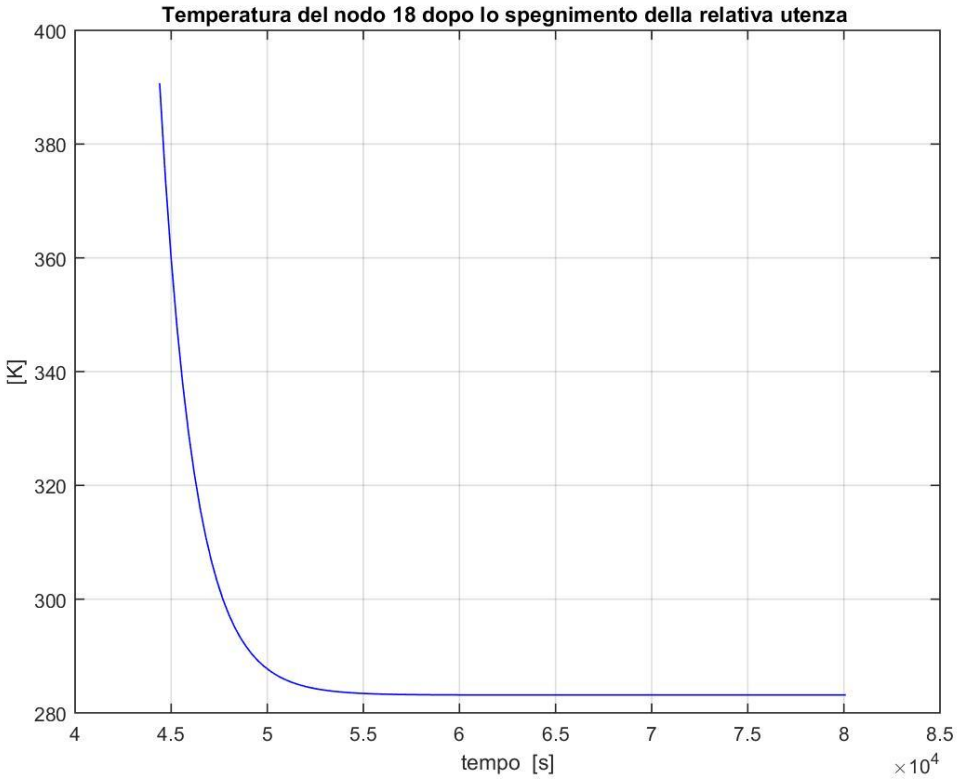
Considerando quindi le equazioni 2.93 e 4.5 e discretizzando nel tempo con il 'backward Euler method' si ottiene

$$\frac{\rho * c_p}{\Delta t} * S_{j+3} * L_{j+3} * \left(T_{i+3}^{t} - T_{i+3}^{t-\Delta t}\right) + c_p * G_{j+3} * (T_{i+3}^{t} - T_i^{t}) =$$
$$= -L_{j+3} * \Omega_{j+3} * U_{j+3} * (T_{i+3}^{t} - T_G) \qquad (4.6)$$

Per poter inserire l'equazione nel sistema matriciale **K*T=f** dobbiamo raggruppare i termini in base alle temperature

$$\left[\frac{\rho * c_p}{\Delta t} * S_{j+3} * L_{j+3} + c_p * G_{j+3} + L_{j+3} * \Omega_{j+3} * U_{j+3}\right] * T_{i+3}^{t} + \left[-c_p * G_{j+3}\right] * T_i^{t} =$$
$$= \frac{\rho * c_p}{\Delta t} * S_{j+3} * L_{j+3} * T_{i+3}^{t-\Delta t} + L_{j+3} * \Omega_{j+3} * U_{j+3} * T_G \qquad (4.7)$$

Con questa modifica delle condizioni al contorno, al nodo 18, dopo lo spegnimento dell'utenza ad esso allacciata, si ottiene il seguente andamento temporale di temperatura



(figura 4.4)

4.2 SIMULAZIONI IN CONDIZIONI STAZIONARIE

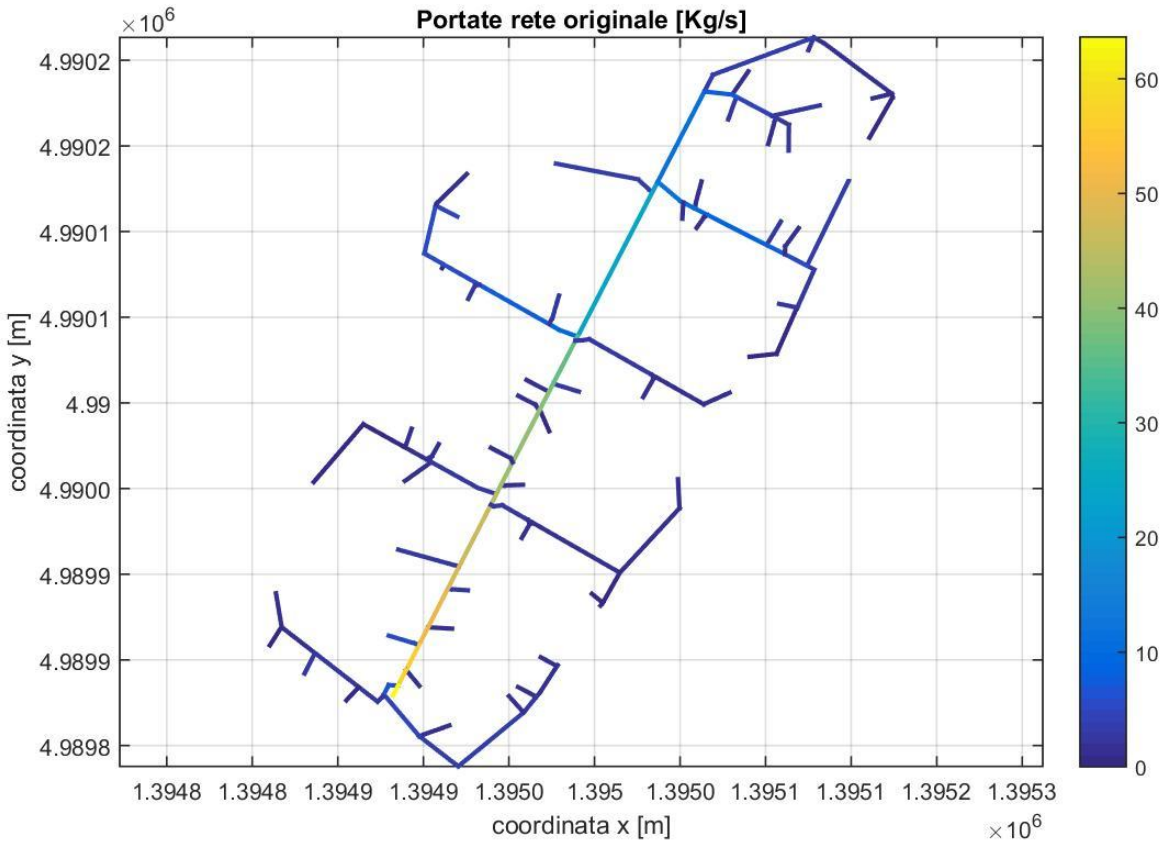
Come già spiegato e motivato precedentemente, per l'analisi in condizioni stazionarie sono stati considerati i carichi e le portate esterne nell' istante di tempo in cui la richiesta totale di potenza è massima.
Il lavoro svolto si è quindi sviluppato in 4 step:

- risoluzione del problema termico e fluido-dinamico sulla rete originale di mandata e di ritorno (figura 4.1) per il calcolo delle portate, delle temperature, della potenza che bisogna fornire alla rete e delle dispersioni durante la distribuzione;
- semplificazione della rete originale;
- risoluzione del problema termico e fluido-dinamico sulle reti semplificate per il calcolo delle portate, delle temperature, della potenza che bisogna fornire alla rete e delle dispersioni termiche attraverso la rete stessa;
- confronto dei risultati ottenuti dalla simulazione sulla rete originale con quelli ottenuti dalle simulazioni sulle reti equivalenti.

4.2.1 SIMULAZIONE SULLA RETE ORIGINALE

Calcolo delle portate e delle temperature

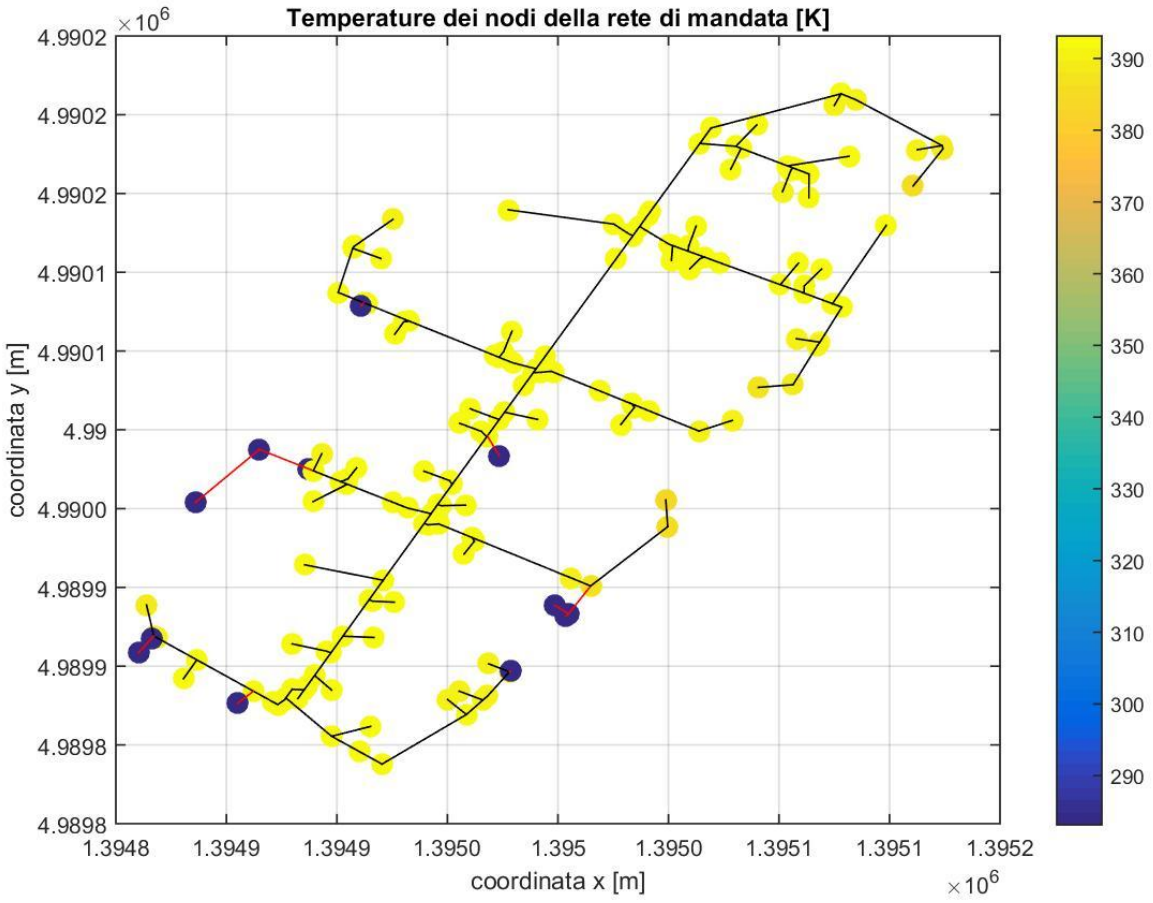
Le portate sono state calcolate utilizzando l'equazione matriciale 2.13.
Il risultato ottenuto è illustrato nella figura sottostante.



(figura 4.5)

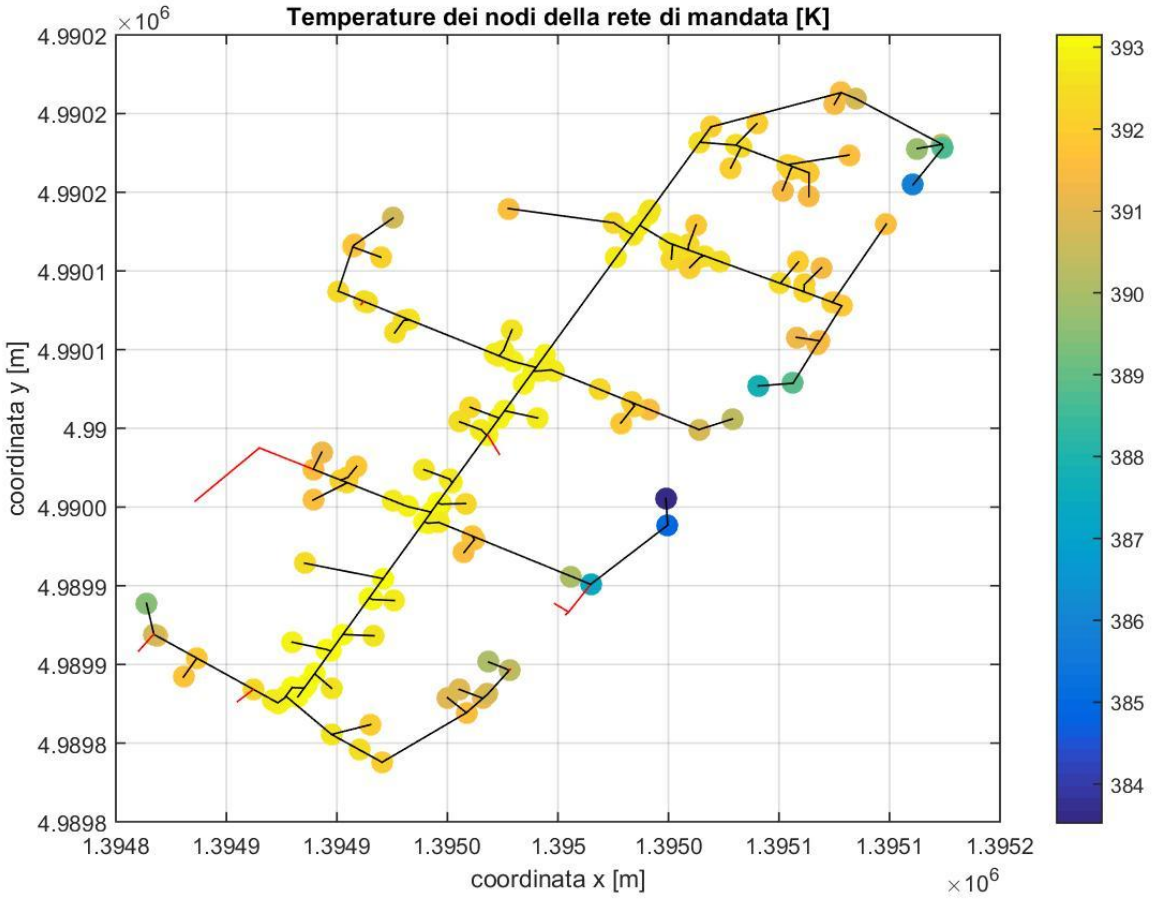
Come possiamo vedere la portata è massima nei pressi del baricentro (posizionato in corrispondenza del cerchietto rosso della figura 4.1) e diminuisce man mano che ci si avvicina alle utenze.

Le temperature sono state calcolate utilizzando l'equazione matriciale 2.66.
Per quanto concerne la rete di mandata, come condizione al contorno, al nodo corrispondente al baricentro è stata imposta una temperatura di 120 °C.
Le temperature ottenute sono rappresentate nell'immagine sottostante in cui i rami con portata nulla sono indicati con un linea rossa, gli altri con una linea nera.



(figura 4.6)

Come possiamo osservare, la temperatura dei nodi a cui non arriva alcuna portata è molto inferiore rispetto a quella degli altri nodi. Ne consegue che, siccome la gamma dei colori deve coprire tutti i valori di temperatura compresi tra quello massimo (393.15 K in corrispondenza del baricentro) e quello minimo (pari alla temperatura del terreno T_G=293.15 K), le temperature relative ai nodi in cui arriva una portata non nulla sono comprese in un intervallo cromatico molto piccolo e quindi non è possibile apprezzarne le differenze.
Nella figura della pagina seguente troviamo la rappresentazione delle temperature dei soli nodi in cui la portata non è nulla.



(figura 4.7)

La temperatura è quindi massima in corrispondenza del baricentro e diminuisce man mano che ci si allontana da esso.

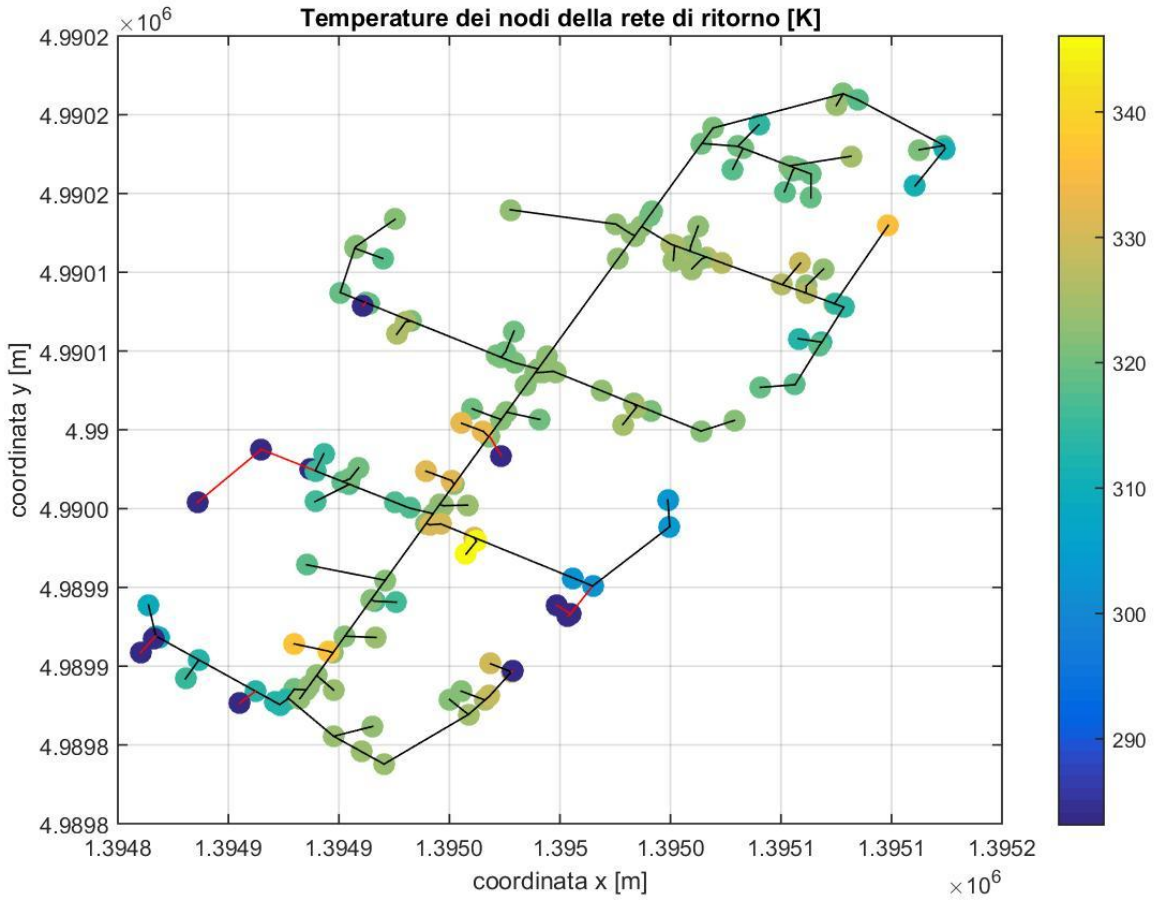
Per la rete di ritorno, invece, la condizione al contorno di temperatura imposta è stata utilizzata per tutti i nodi corrispondenti alle utenze. La temperatura da imporre è stata calcolata con

$$T_{r,i} = T_{m,i} - \frac{\dot{Q}_i}{G_{ext,i} * c_{p,w}} \qquad (4.8)$$

Dove

- $T_{r,i}$ = temperatura del nodo corrispondente all'utenza i sulla rete di ritorno
- $T_{m,i}$ = temperatura del nodo corrispondente all'utenza i sulla rete di mandata
- Q_i' = carico richiesto dall'utenza i
- $G_{ext,i}$ = portata estratta dalla rete di mandata per servire le utenze

Nella figura 4.8 troviamo le temperature dei nodi della rete di ritorno



(Figura 4.8)

Anche in questo caso i nodi esterni in cui non passa alcuna portata hanno una temperatura pari a quella del terreno.

Come possiamo osservare, se si segue un percorso che va da una qualsiasi utenza verso il baricentro, non sempre la temperatura decresce: ciò è causato dalla presenza di nodi in cui si immettono più rami ed in corrispondenza dei quali si ha una miscelazione tra diverse portate.

<u>Calcolo della potenza da fornire e delle dispersioni della rete</u>

La potenza da fornire alla rete è stata calcolata con la seguente equazione

$$\dot{Q} = G_{in,bar} * c_{p,w} * \left(T_{bar,m} - T_{bar,r}\right) \qquad (4.9)$$

Dove

- $G_{in,bar}$ è la portata in ingresso nella rete di distribuzione di mandata
- $T_{bar,m}$ è la temperatura del nodo della rete di mandata corrispondente al baricentro
- $T_{bar,r}$ è la temperatura del nodo della rete di ritorno corrispondente al baricentro, ovvero la temperatura di ritorno nella rete di trasporto

Per ottenere la dispersione della rete di distribuzione BCT408 si è sottratto a Q' la sommatoria dei carichi richiesti dalle utenze.

4.2.3 SEMPLIFICAZIONE DELLA RETE ORIGINALE

In questa fase, mediante il modello di semplificazione visto nel capitolo 3, sono state costruite in successione delle reti semplificate aumentandone di volta in volta il grado di compattazione.
Gli input da fornire al modello di compattazione sono i seguenti:

- matrice di incidenza;
- lunghezze dei rami;
- diametri interni dei rami;
- coefficienti di scambio termico lineare delle tubazioni in W/m/k;
- diametri esterni dei rami (supposti pari a 1.1* d_int);
- portate non nulle calcolate nello step precedente;
- densità del fluido che percorre i rami (acqua);
- densità del metallo delle tubazioni (supposto pari a 7750 kg/m^3);
- calore specifico del metallo delle tubazioni supposto pari a 502 J/Kg/K;
- calore specifico dell'acqua supposto costante e pari a 4186 J/Kg/K.

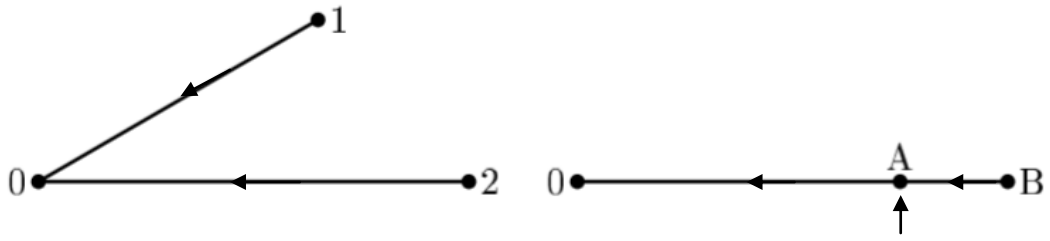
Dati riferiti alla rete in figura 4.2

Gli output ottenuti sono qui sotto elencati:

- matrici di incidenza delle reti semplificate;
- lunghezze dei rami delle reti semplificate;
- diametri interni dei rami delle reti semplificate;
- coefficienti di scambio termico lineare delle tubazioni in W/m/k;
- carichi delle reti semplificate.

4.2.4 SIMULAZIONI SULLE RETI SEMPLIFICATE E CONFRONTO DEI RISULTATI

Il procedimento per calcolare le portate e le temperature è analogo a quello utilizzato per la rete originale; bisogna solo prestare attenzione all'applicazione delle BC sulla rete di ritorno.
Consideriamo ad esempio le seguenti reti di ritorno di cui quella sulla sinistra corrisponde alla rete originale e l'altra alla rete semplificata.

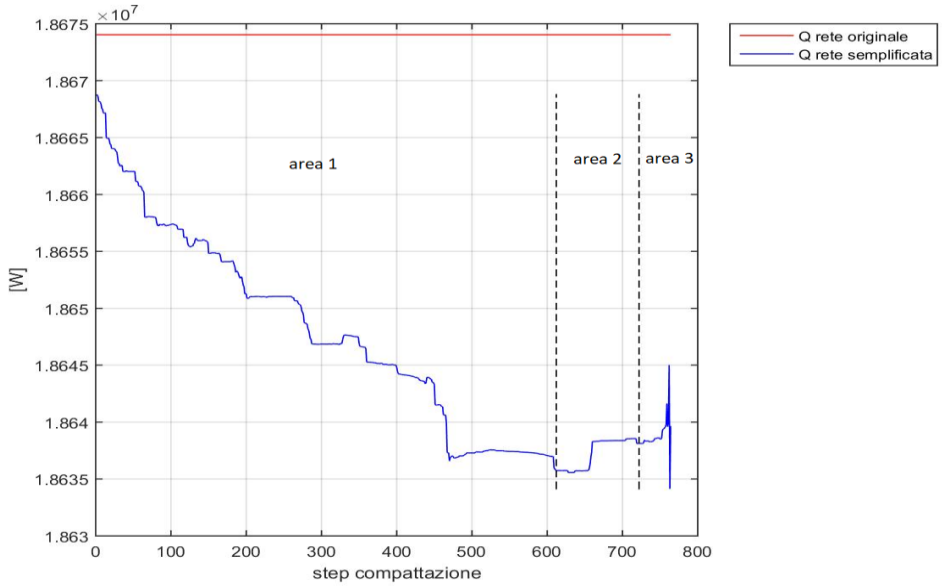


(Figura 4.9)

Le utenze A e B della rete semplificata rappresentano le utenze 1 e 2 della rete originale.
Come possiamo osservare nella rete originale le condizioni di temperatura imposta devono essere applicate ad entrambe le utenze, mentre nella rete equivalente solo per l'utenza B; inoltre nel nodo dell'utenza A bisogna aggiungere una miscelazione con la portata esterna $G_{ext,A}$ che si trova ad una temperatura $T_{r,A}$ calcolata con l'equazione 4.8.
Più in generale nella rete di ritorno equivalente la condizione al contorno di temperatura imposta viene applicata ad un solo nodo e a tutti gli altri nodi che si allacciano ad un'utenza bisogna aggiungere la miscelazione con la portata esterna proveniente dalla sottostazione dell'utenza stessa.

<u>Potenza da fornire e dispersioni delle reti</u>

Anche per la rete equivalente, la potenza da fornire al baricentro è stata calcolata con l'equazione 4.9 e le dispersioni sono state ottenute per differenza con il carico totale.
Nell'immagine sottostante abbiamo i risultati ottenuti in termini di potenza da fornire alla rete.

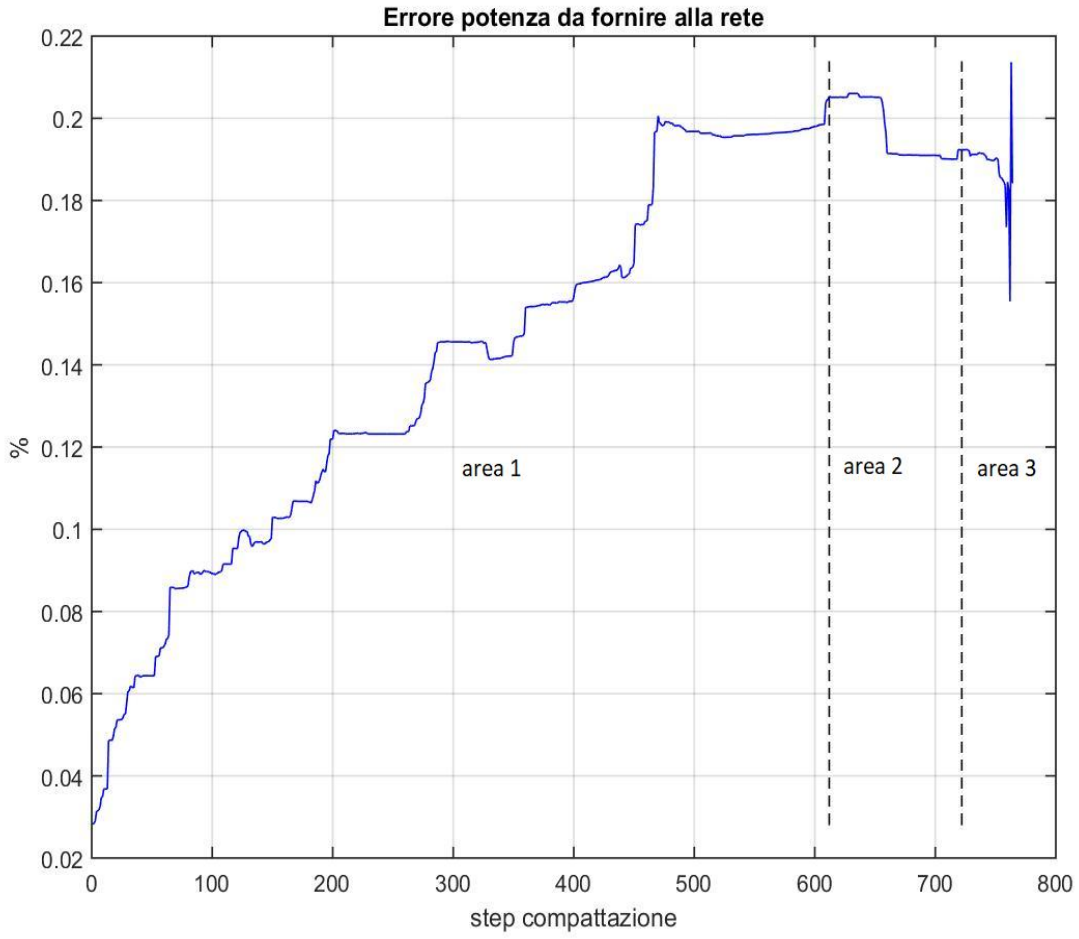


(Figura 4.10)

Sull'asse delle ascisse abbiamo il numero di step di compattazione e sull'asse delle ordinate troviamo la potenza da fornire corrispondente in W.
Il grafico è stato suddiviso in 3 aree:

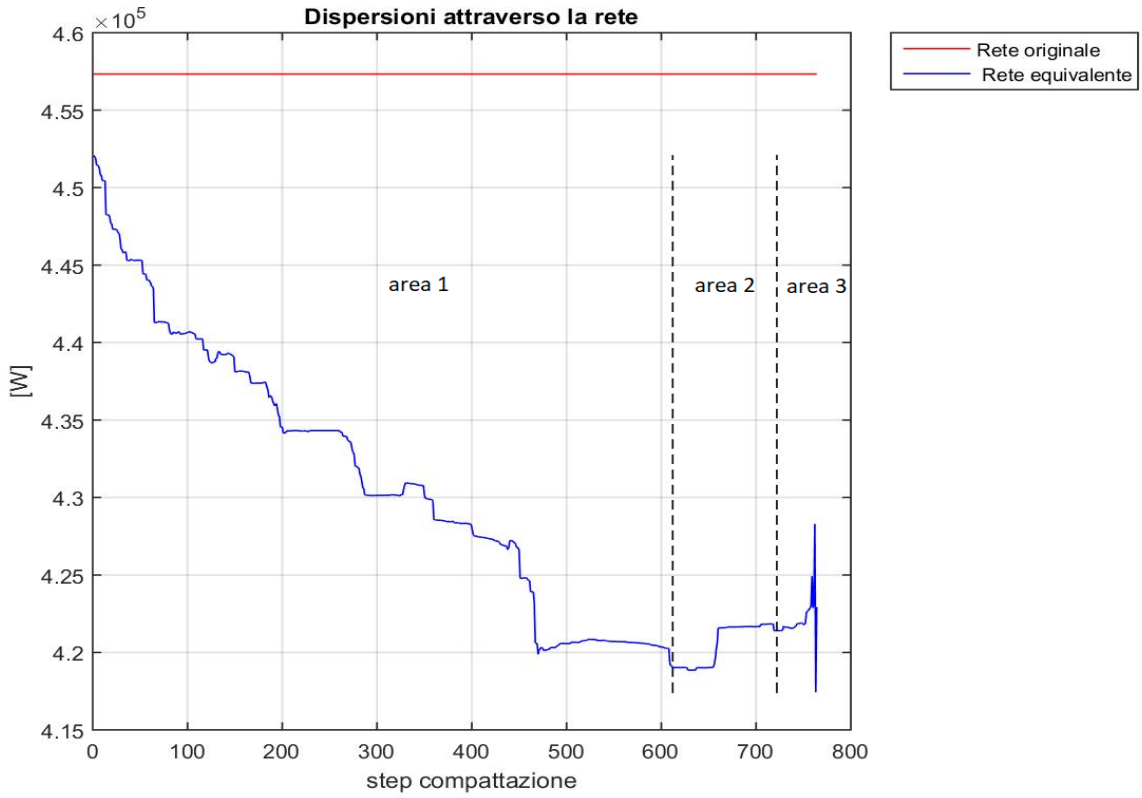
- nell'area 1 sono state semplificate le ramificazioni (vedere capitolo 3.3).
  Come possiamo osservare la maggior parte degli step di compattazione della rete è dedicata al passaggio dalla struttura originale ad albero ad una struttura lineare.
  In questa sezione del grafico lo scostamento tra la linea blu e quella rossa aumenta con il grado di compattazione della rete;
- nell' area 2 si ha l'eliminazione dei rami che non servono alcun utenza.
  Escludendo il gradino presente circa allo step 650, lo scostamento tra le due linee non varia;
- nell'area 3 si effettua un' ulteriore eliminazione dei rami rimasti; di conseguenza avviene anche un'aggregazione delle utenze.
  Lo scostamento tra le due curve è inizialmente costante per poi oscillare negli ultimi step.

Nella figura 4.11 è riportato l'andamento al variare del grado di compattazione dello scostamento percentuale della potenza da fornire calcolata sulle reti equivalenti, rispetto alla potenza calcolata sulla rete originale

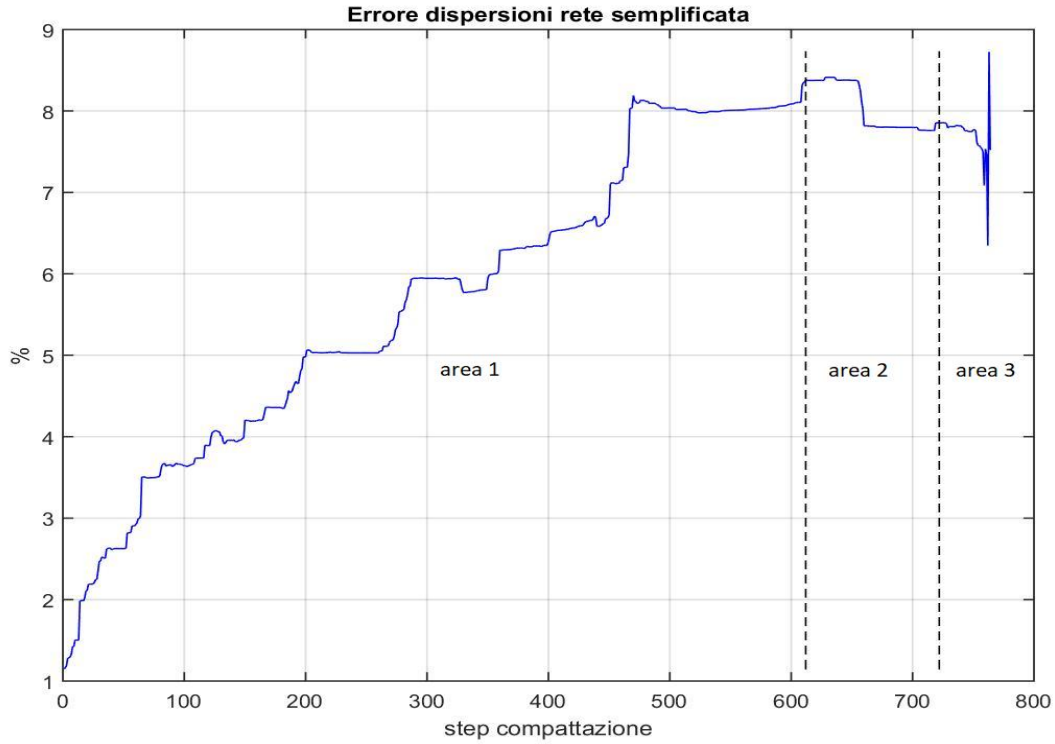


(Figura 4.11)

Poiché il carico totale della rete originale coincide con quello delle reti equivalenti, al fine di valutare la precisione del modello di compattazione, risulta opportuno confrontare le dispersioni termiche attraverso le reti di distribuzione.



(figura 4.12)



(Figura 4.13)

La figura 4.12 mostra le dispersioni calcolate sulla rete originale e l'andamento delle dispersioni calcolate sulle reti equivalenti al progredire delle semplificazioni di rete effettuate.
Nella figura 4.13 troviamo invece l'andamento al variare del grado di compattazione dello scostamento, espresso in percentuale, tra le dissipazioni della rete originale e quelle delle reti semplificate.

La suddivisione delle aree dei grafici è analoga a quella presente nelle immagini relative alle potenze da fornire alla rete.

Come possiamo vedere, rimanendo costante il carico totale durante la compattazione, gli andamenti qualitativi delle figure 4.12 e 4.13 sono identici a quelli delle figure 4.10 e 4.11.

4.3 SIMULAZIONI IN CONDIZIONI TRANSITORIE 1

In questo sottocapitolo si intende descrivere il comportamento del modello in condizioni transitorie, con temperatura di ingresso nella rete di mandata costante e portate variabili a seconda dei carichi termici richiesti dalle utenze.

La procedura adottata è simile a quella utilizzata per le condizioni stazionarie:

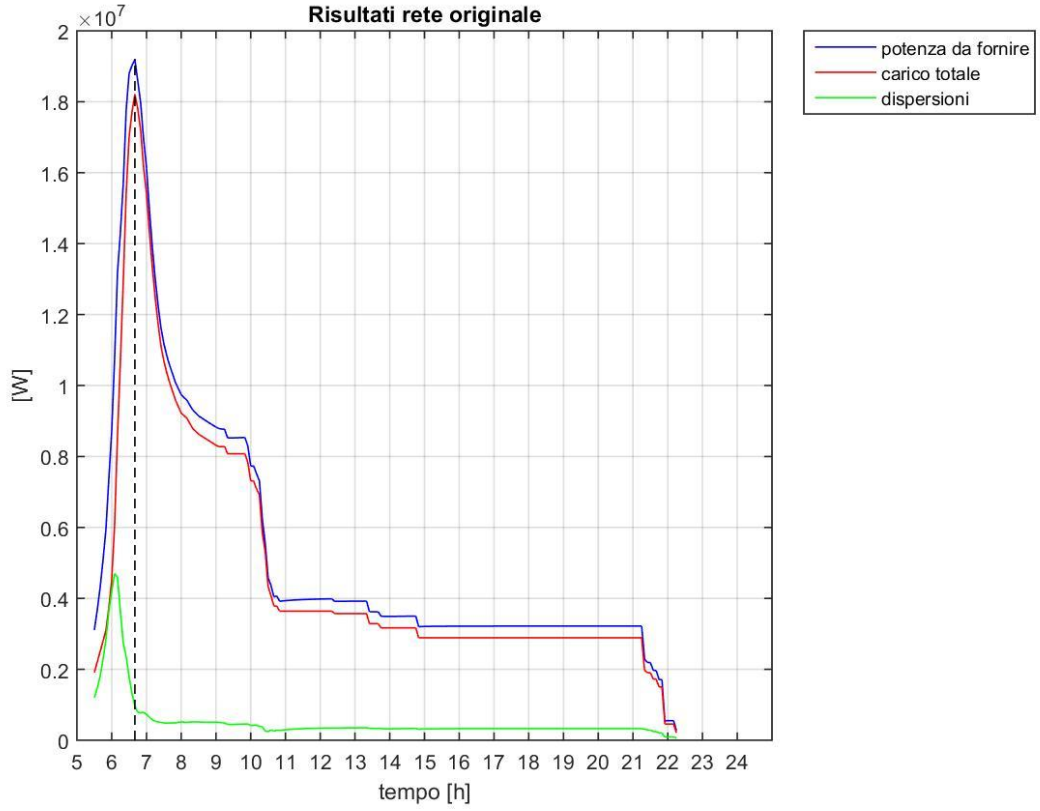
- risoluzione del problema termico e fluido-dinamico sulla rete originale di mandata e di ritorno (figura 4.1) nei vari istanti di tempo considerati e calcolo in ognuno di questi della potenza che bisogna fornire al baricentro e delle dispersioni durante la distribuzione;
- ripetizione dei calcoli effettuati per la rete originale sulle reti equivalenti generate dal modello di compattazione applicato al caso stazionario;
- confronto dei risultati ottenuti dalla simulazione sulla rete originale con quelli ottenuti dalle simulazioni sulle reti equivalenti.

4.3.1 SIMULAZIONE SULLA RETE ORIGINALE

Il calcolo delle temperature è stato eseguito come riportato nel paragrafo 2.3.2 e 4.1.3 mentre per il calcolo delle portate è stata utilizzata l'equazione 2.37 con $\mathbf{r}^t$ vettore nullo in quanto, come spiegato nel paragrafo 4.1.1, la densità è stata supposta costante.
Per quanto riguarda le condizioni iniziali, sia per la rete di mandata che per la rete di ritorno è stata supposta una temperatura pari a quella del terreno mentre le condizioni al contorno sono state imposte in modo analogo a quanto fatto per le simulazioni in stazionario.
La potenza che deve essere fornita alla rete viene calcolata con l'equazione 4.9 applicata ad ogni istante di tempo considerato. In seguito, per ottenere le dispersioni, si è sottratto il carico totale alla potenza precedentemente calcolata.

<u>Potenza da fornire e dispersioni della rete originale</u>

I risultati in termini di potenza da fornire alla rete e dissipazione termica attraverso la stessa sono rappresentati nell'immagine sottostante.



(Figura 4.14)

La linea verticale nera tratteggiata indica l'istante di tempo in cui si ha il carico massimo.
Dall'immagine 4.14 possiamo osservare un tipico andamento del carico richiesto: durante la notte le utenze restano spente, nelle prime ore del mattino la richiesta è massima in quanto è necessario riscaldare gli ambienti raffreddatisi durante le ore notturne, dopo questo picco il carico totale diminuisce ed in seguito rimane costante fino alla sera.

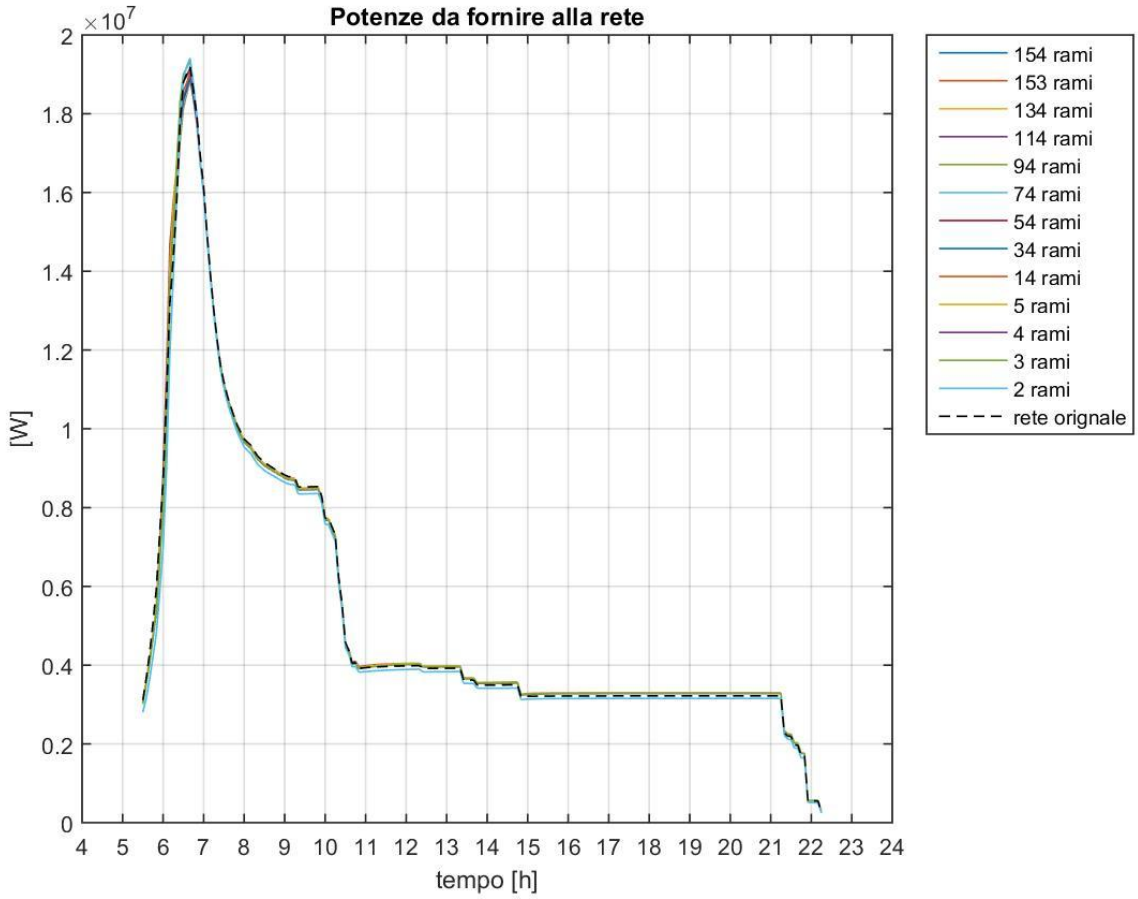
Come possiamo notare osservando la curva verde, durante l'accensione dell'impianto si ha un picco delle dispersioni dovute al riscaldamento del sistema. Avendo infatti assunto una temperatura iniziale del fluido pari a quella del terreno, nei primi istanti di funzionamento, oltre a fornire la potenza richiesta dalle utenze, la centrale deve produrre anche l'energia necessaria per portare il fluido termovettore alla temperatura di progetto. Essendo le dispersioni calcolate come differenza tra la potenza prodotta dalla centrale e il carico totale, l'energia necessaria per il riscaldamento dell'acqua viene inglobata nella suddetta dispersione.
Terminato il transitorio di riscaldamento del sistema, le dispersioni si assestano ad un valore costante di circa 5*1e5 W.

4.3.2 SIMULAZIONI SULLE RETI EQUIVALENTI

Il procedimento per calcolare le portate, le temperature, la potenza da fornire alla rete e le dispersioni attraverso la stessa è analogo a quello utilizzato per la rete originale; come già spiegato nel paragrafo 4.2.4, l'unica differenza è presente nell' applicazione delle BC sulla rete di ritorno.
Le condizioni iniziali sulla temperatura naturalmente sono quelle utilizzate nella rete originale (temperatura iniziale pari a quella del terreno).

Potenze da fornire e dispersioni

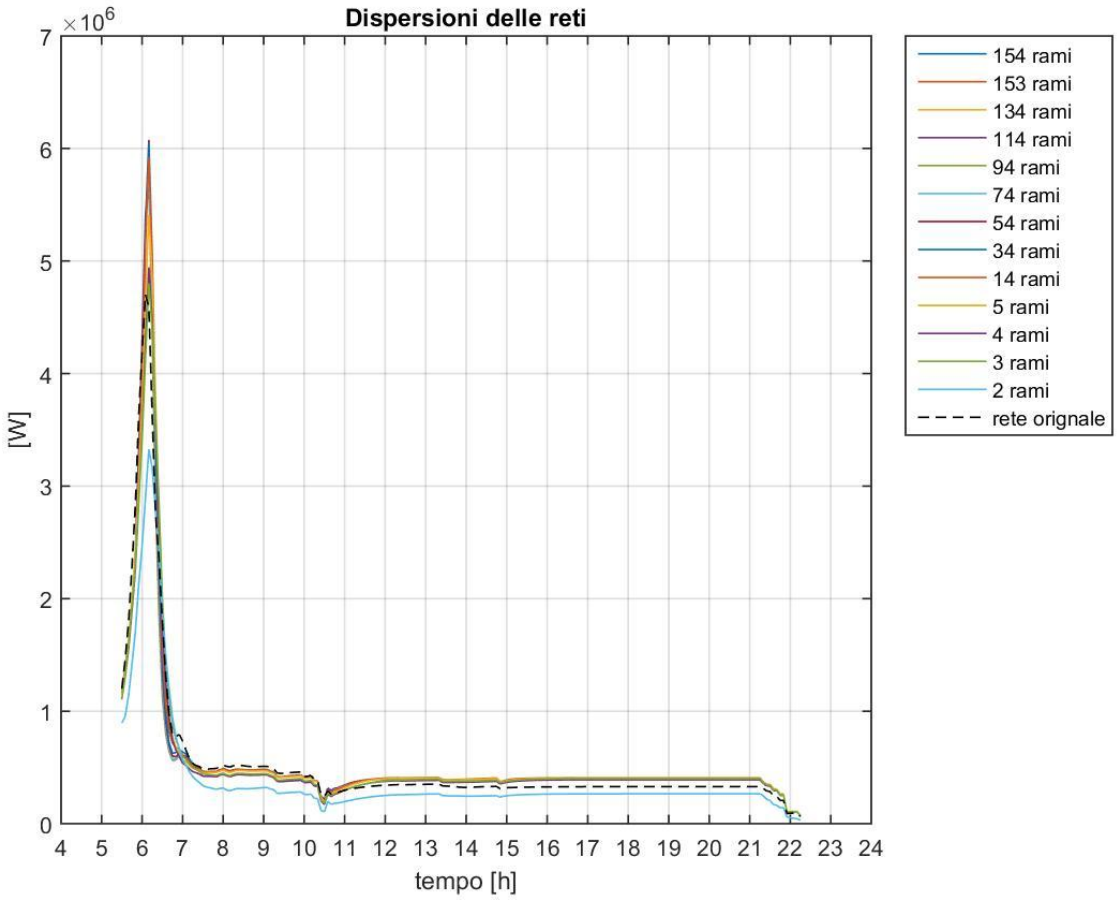
Nella figura 4.15 sono rappresentati gli andamenti temporali delle potenze da fornire alle reti equivalenti indicate nella legenda.



(Figura 4.15)

La curva relativa a 154 rami (numero di rami della rete originale) è riferita alla rete equivalente ottenuta dopo la semplificazione di tutte le ramificazioni; le altre curve si riferiscono alla reti equivalenti in cui, oltre alla semplificazione delle ramificazioni, sono state effettuate delle compattazioni fino all'ottenimento del numero di rami indicato nella legenda.
Come possiamo osservare, nonostante il modello di semplificazione adottato sia costruito per essere applicato in condizioni stazionarie, le curve delle reti equivalenti seguono abbastanza

fedelmente il profilo temporale di potenza da fornire calcolato con le simulazioni effettuate sulla rete di distribuzione originale.
Come esposto nello studio del modello in condizioni stazionarie, conservandosi il carico totale tra la rete originale e quelle equivalenti, può essere interessante valutare le dispersioni.



(Figura 4.16)

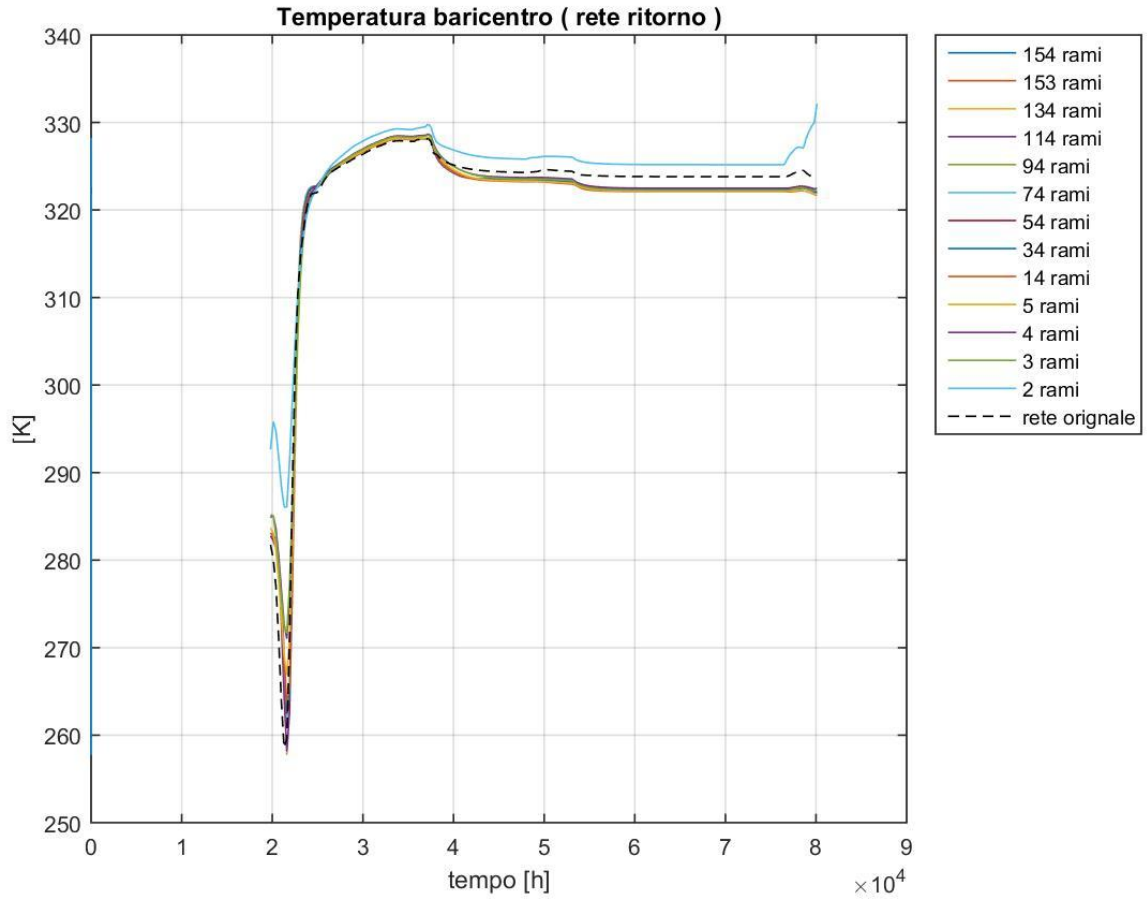
Dall'immagine 4.16 notiamo che, indipendentemente dal grado di compattazione, in corrispondenza del transitorio di accensione il modello è alquanto impreciso; ciò è dovuto alla natura stessa del modello che, essendo valido solo in condizioni stazionarie, non considera variazioni di temperatura nel tempo del fluido termovettore.
Una volta terminato il transitorio di accensione, escludendo la curva relativa alla rete equivalente composta da 2 rami, gli scostamenti tra le dispersioni relative alla rete originale e quelle relative alle reti equivalenti sono minimi.

Temperatura di ritorno nella rete di trasporto

Con riferimento all'equazione 4.9:

- nel modello di semplificazione utilizzato in questa tesi, durante l'eliminazione delle ramificazioni le portate esterne non variano e durante il processo di compattazione vengono ridistribuite tra i vari nodi attraverso i coefficienti g_{AA}, g_{AB}, g_{BA} , g_{BB} (vedere capitolo 3); di conseguenza la sommatoria di queste non varia. Poichè la portata entrante/uscente nella/dalla rete attraverso il baricentro è pari alla sommatoria delle portate esterne, è possibile affermare che il termine $G_{in,bar}$ non varia con la semplificazione della rete originale;
- nella rete di mandata, sia per la rete originale che per le reti equivalenti, la temperatura del nodo corrispondente al baricentro è imposta pari a 120 °C

Di conseguenza l'unico termine che può variare con la compattazione ed influenzare il valore della potenza $\dot{Q}$ è la temperatura del nodo sulla rete di ritorno corrispondente al baricentro; nell' immagine sottostante ne troviamo l'evoluzione temporale per alcune reti equivalenti.



(Figura 4.17)

Come per le potenze da fornire, anche per le temperature di ritorno, gli scostamenti maggiori tra la curva riferita alla rete originale e quelle relative alle reti semplificate si verificano durante il transitorio di accensione.

Errori

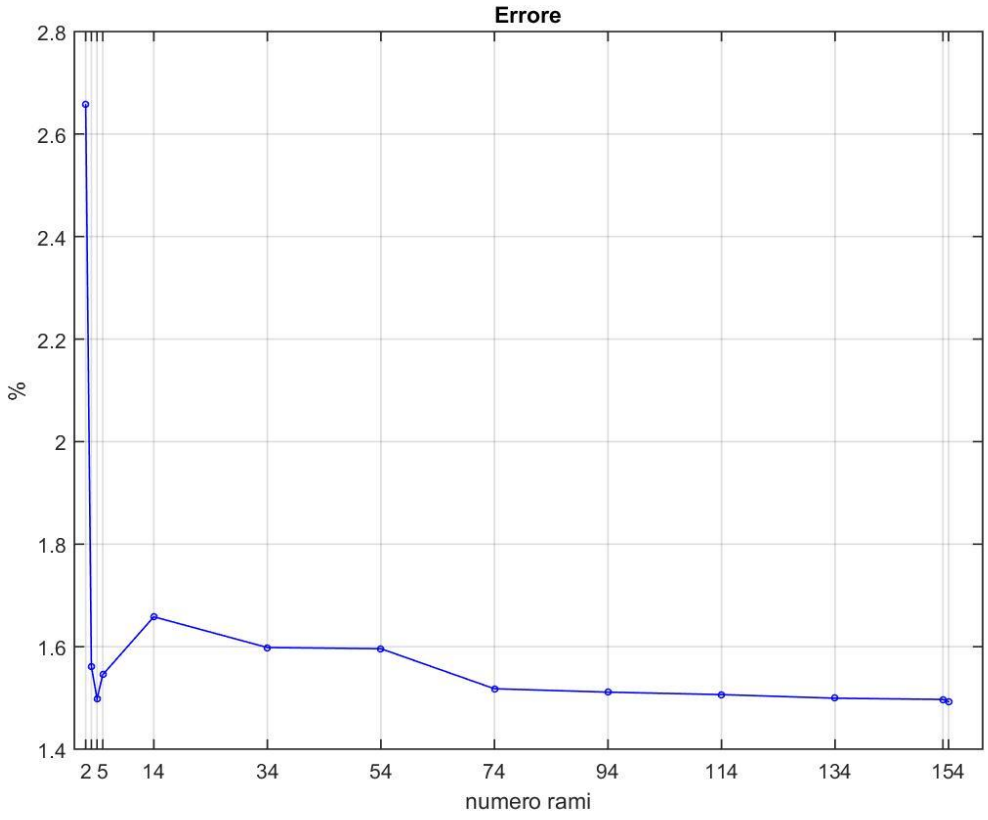
Quantifichiamo ora l'errore generato usando la rete semplificata e studiamone la relazione con il numero di rami.
L'errore nel transitorio è stato definito come la sommatoria di tutti gli scostamenti in valore assoluto nei vari istanti di tempo tra la potenza da fornire al baricentro calcolata sulla rete originale e quella calcolata sulla rete equivalente.

$$err = \sum_{t=t_iniz}^{t=t_fin} |\dot{Q}_{orig}^{t} - \dot{Q}_{equiv}^{t}| \qquad (4.10)$$

Quest'errore è stato successivamente espresso in percentuale rispetto alla sommatoria nel tempo della potenza da fornire al baricentro calcolato sulla rete originale.

$$err_{perc} = \frac{\sum_{t=t_{iniz}}^{t=t_{fin}} |\dot{Q}_{orig}^{t} - \dot{Q}_{equiv}^{t}|}{\sum_{t=t_{iniz}}^{t=t_{fin}} \dot{Q}_{orig}^{t}} * 100 \qquad (4.11)$$

Nell'immagine sottostante è raffigurato il risultato ottenuto.

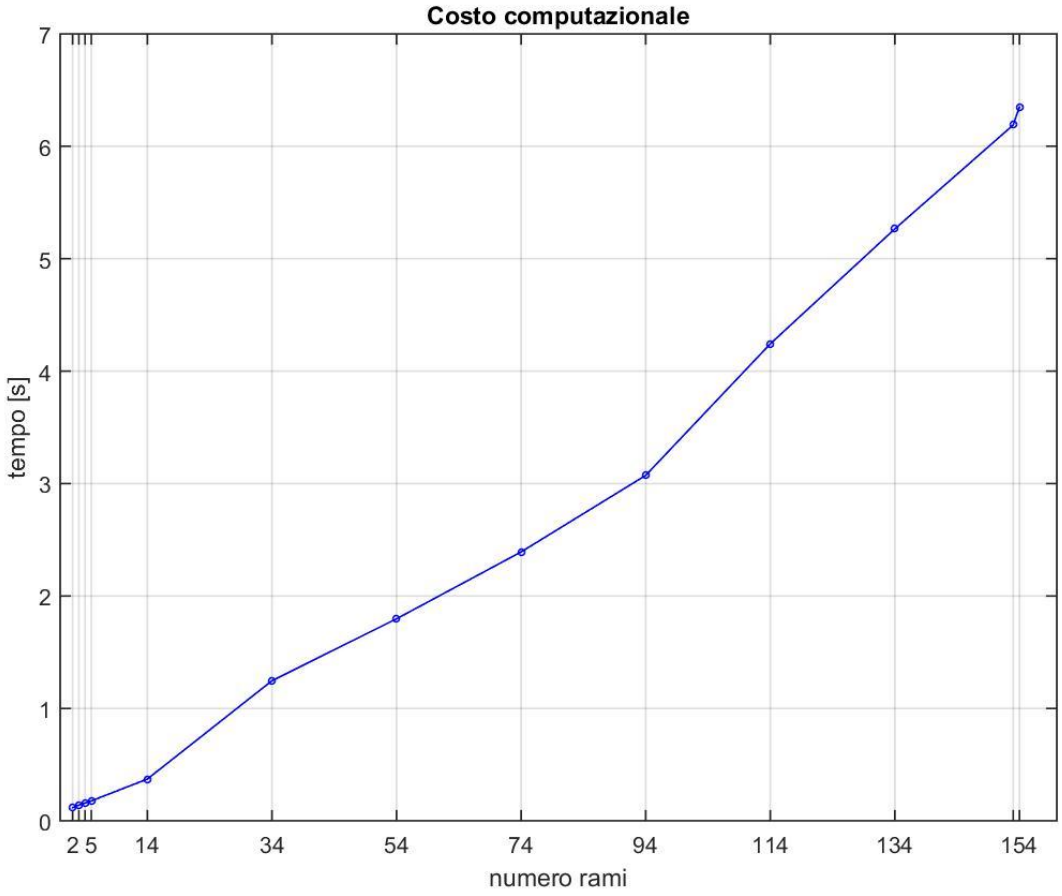


(Figura 4.18)

Il grafico ci mostra che, dopo aver semplificato tutte le ramificazioni, si ottiene un errore di circa 1.5%. In seguito, al diminuire del numero dei rami, l'errore tende a crescere in maniera lieve e nel tratto finale, corrispondente ad un alto grado di compattazione, oscilla ed in corrispondenza di una rete composta da 2 rami assume un valore superiore al 2.6%.

Costo computazionale

Il costo computazionale di un problema è definito come il tempo impiegato dal calcolatore per trovarne la soluzione.
L'immagine sottostante mostra quindi il tempo impiegato dal computer (CPU da 2.2 GHz) nella risoluzione del problema fluidodinamico e del problema termico in funzione del numero dei rami della rete equivalente.



(Figura 4.19)

Come possiamo osservare, al diminuire del numero dei rami il costo computazionale si riduce.
Quest'andamento si spiega considerando che, riducendo il numero dei rami (e quindi anche il numero dei nodi), le dimensioni delle matrici e dei vettori delle equazioni matriciali 2.37 e 2.66 diminuiscono e di conseguenza si riducono anche le operazioni che il calcolatore deve effettuare per trovare i vettori risultati delle portate e delle temperature.
L'andamento della linea è inoltre approssimabile ad un andamento lineare. Ciò probabilmente è dovuto al fatto che il numero di operazioni svolte è direttamente proporzionale alle dimensioni delle matrici **H** e **K** e di conseguenza al numero dei nodi e dei rami (vedere capitolo 3).

4.4 SIMULAZIONI IN CONDIZIONI TRANSITORIE 2

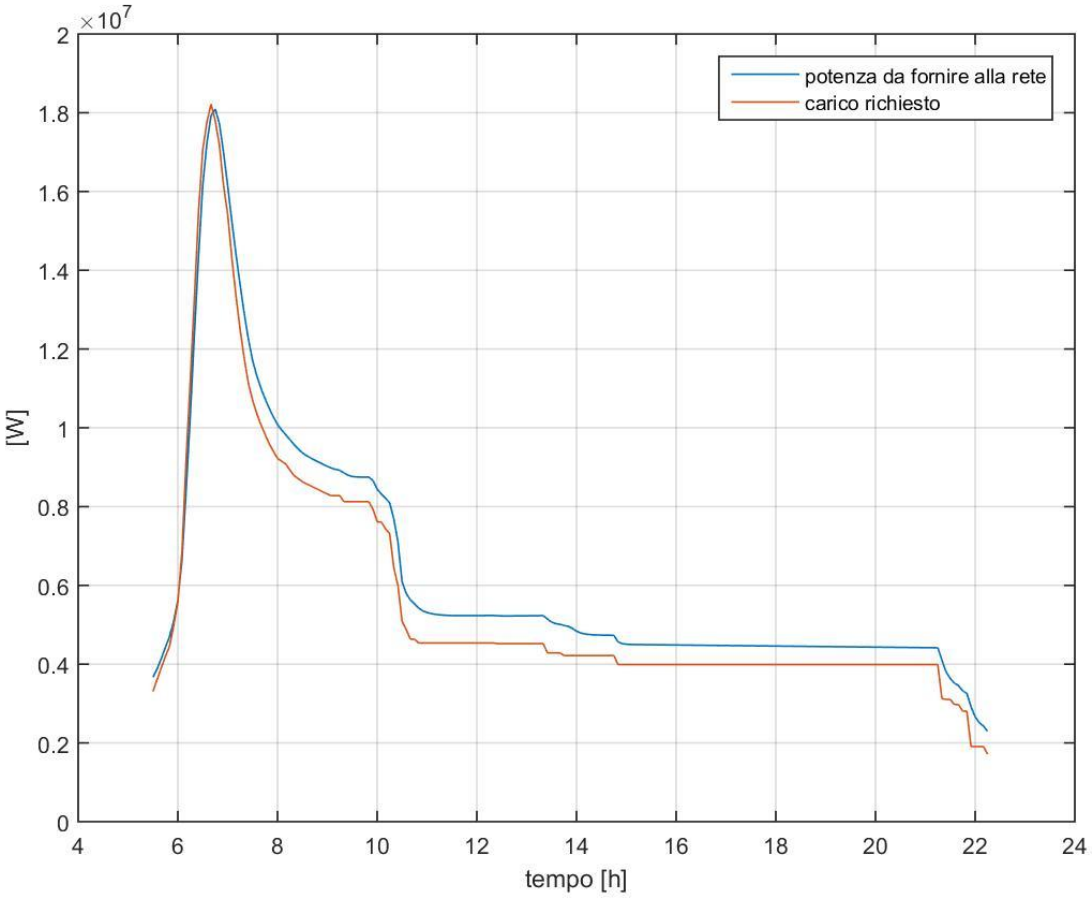
In questo sottocapitolo si intende studiare il comportamento del modello in condizioni transitorie con temperatura di ingresso nella rete di mandata variabile e portate nei rami costanti.
Come possiamo osservare dall' equazione 4.8, essendo le portate costanti, al variare dei carichi si ha una variazione delle temperature in uscita dalle utenze e quindi anche delle temperature sulla rete di ritorno . Nelle simulazioni a portata variabile, invece, le temperature in uscita dalle utenze rimangono pressoché costanti in quanto le portate seguono l'andamento dei carichi.
Mediante le variazioni di temperatura è quindi possibile evidenziare il tempo impiegato dal fronte di temperatura a percorrere le tubazioni della rete e osservare come questo influenza il comportamento della rete stessa.
Infine si può verificare se anche le reti equivalenti generate dal modello di compattazione permettono di simulare il ritardo del fronte di temperatura.
Al fine di evitare il più possibile la presenza di temperature molto basse, è stata fatta la scelta di utilizzare le portate relative all'istante di tempo in cui si ha il carico massimo.

4.4.1 SIMULAZIONE SULLA RETE ORIGINALE

Il calcolo delle temperature è stato eseguito come riportato nel paragrafo 2.3.2 e 4.1.3, mentre per il calcolo delle portate è stata utilizzata l'equazione 2.37 con $\mathbf{r}^t$ vettore nullo.
La potenza che deve essere fornita alla rete viene calcolata con l'equazione 4.9 applicata ad ogni istante di tempo considerato.

Potenza da fornire

Nell'immagine sottostante troviamo gli andamenti nel tempo del carico totale richiesto dalle utenze e della potenza da fornire alla rete.

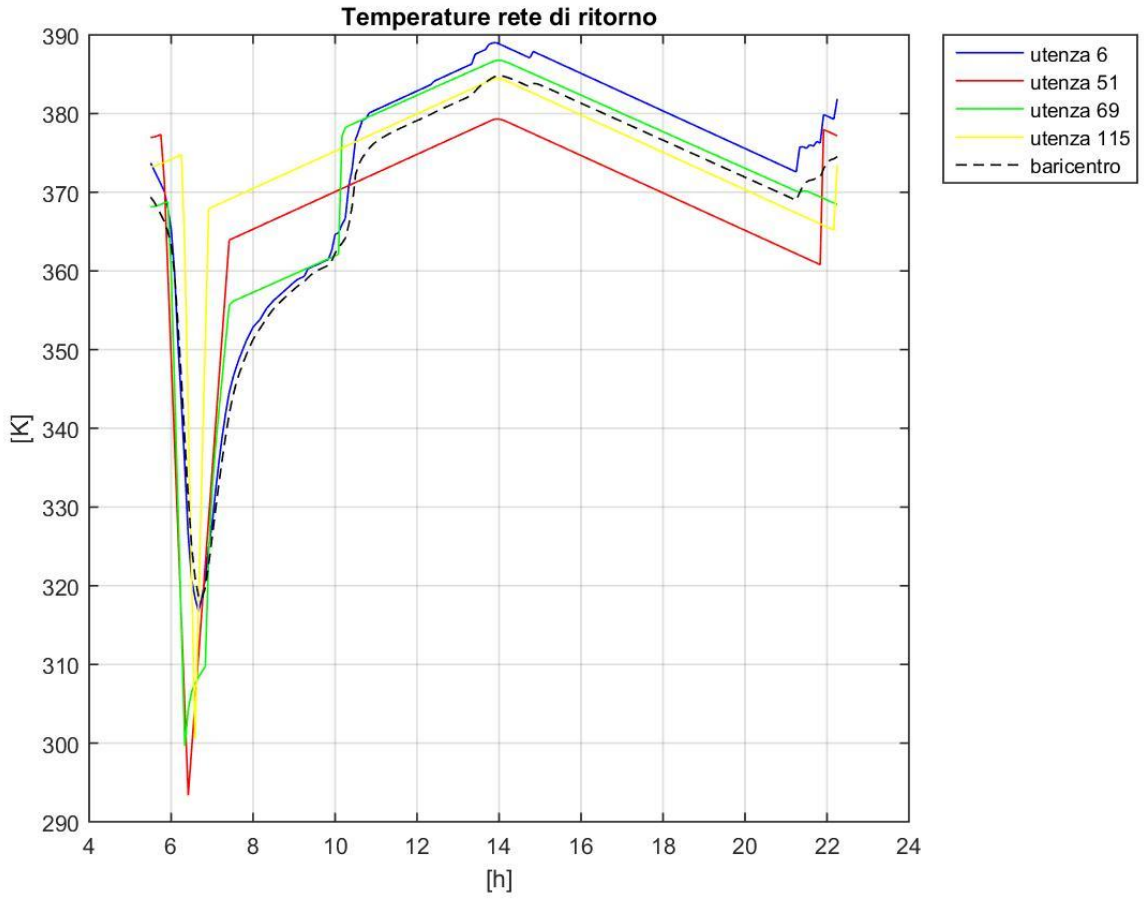


(Figura 4.20)

Come possiamo osservare le due curve si intersecano ed il picco della potenza da fornire è traslato ed attenuato rispetto al picco del carico richiesto.
Questo comportamento è dovuto al ritardo temporale del fronte di temperatura sulla rete di ritorno.
Avendo imposto sempre la portata totale massima, negli istanti iniziali le variazioni di temperatura a cavallo delle utenze sono minime in quanto i carichi sono bassi; di conseguenza le temperature dei nodi sulla rete di ritorno sono alte (simili a quelle della rete di mandata).

Man mano che i carichi aumentano, le variazioni di temperatura a cavallo delle utenze crescono, ma il fluido freddo uscente dalle utenze impiega del tempo a percorrere i rami della rete di ritorno.
Pertanto, per l'intervallo di tempo corrispondente al ritardo del fronte di temperatura bassa, la temperatura del baricentro della rete di ritorno e quindi la potenza da fornire alla rete non risente della variazione dei carichi.
Il comportamento della rete appena descritto causa perciò l'intersezione tra le due curve della figura 4.20.

Quanto suddetto viene evidenziato nel seguente grafico riportante l'andamento temporale della temperatura di uscita da alcune utenze e della temperatura del baricentro sulla rete di ritorno.



(Figura 4.21)

Nei primissimi istanti di funzionamento del sistema, la temperatura delle utenze è sempre maggiore rispetto alla temperatura del baricentro. Dopo pochi step temporali il carico aumenta e di conseguenza le temperature all'uscita delle utenze diminuiscono. Come possiamo osservare dalla figura soprastante, la curva riferita al baricentro interseca le curve riferite alle utenze ed inoltre, come mostrato nello zoom dell'immagine 4.21, il baricentro raggiunge la sua temperatura minima dopo che tutte le utenze hanno già raggiunto il picco negativo. Ciò conferma la presenza di un ritardo del fronte di temperatura.

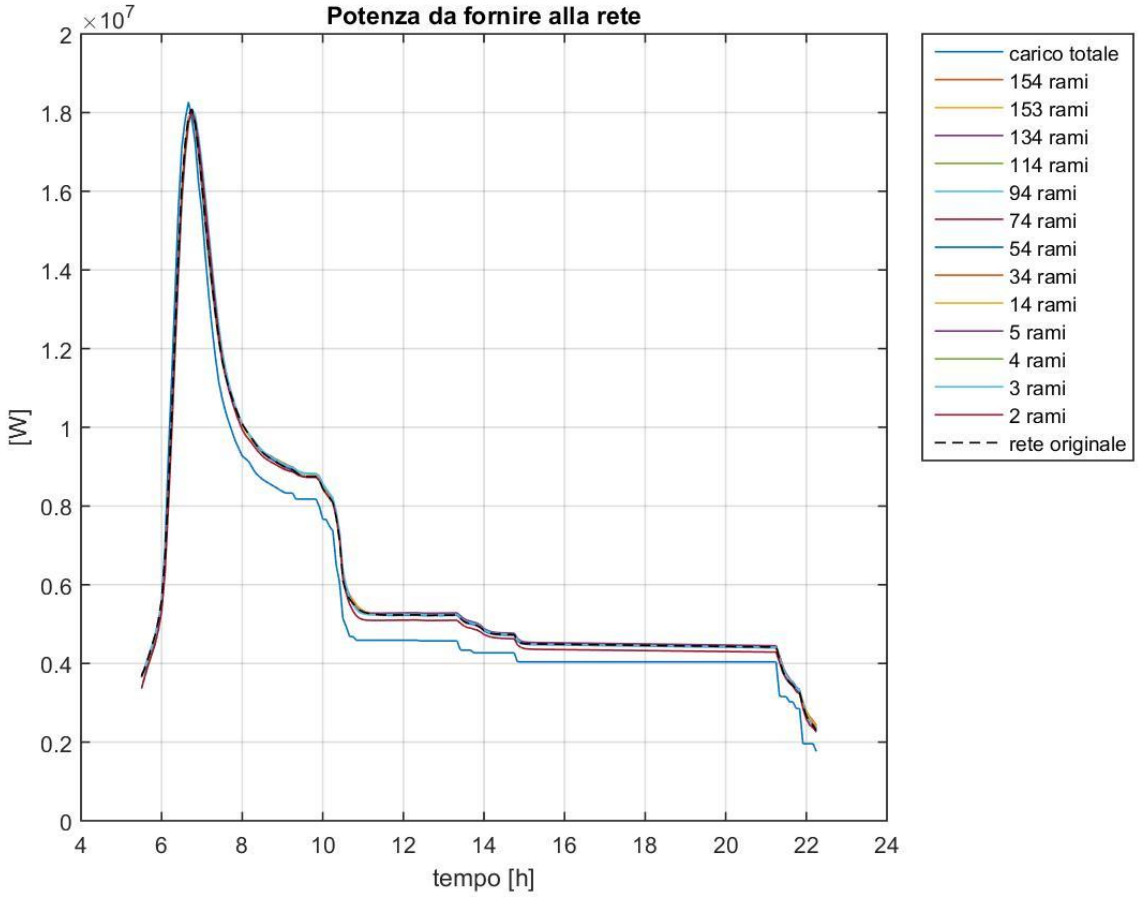
(zoom immagine 4.21)

4.4.2 SIMULAZIONI SULLE RETI EQUIVALENTI

Il procedimento per calcolare le portate, le temperature, la potenza da fornire alla rete e le dispersioni attraverso la stessa è analogo a quello utilizzato per la rete originale; come già spiegato nel paragrafo 4.2.4, l'unica differenza risiede nell' applicazione delle BC sulla rete di ritorno.
Anche per le reti equivalenti le condizioni iniziali sono state calcolate mediante una simulazione in stazionario.

Potenza da fornire

Nella figura 4.22 troviamo gli andamenti temporali del carico totale e delle potenze da fornire alle reti equivalenti con un numero di rami pari a quello indicato nella legenda.

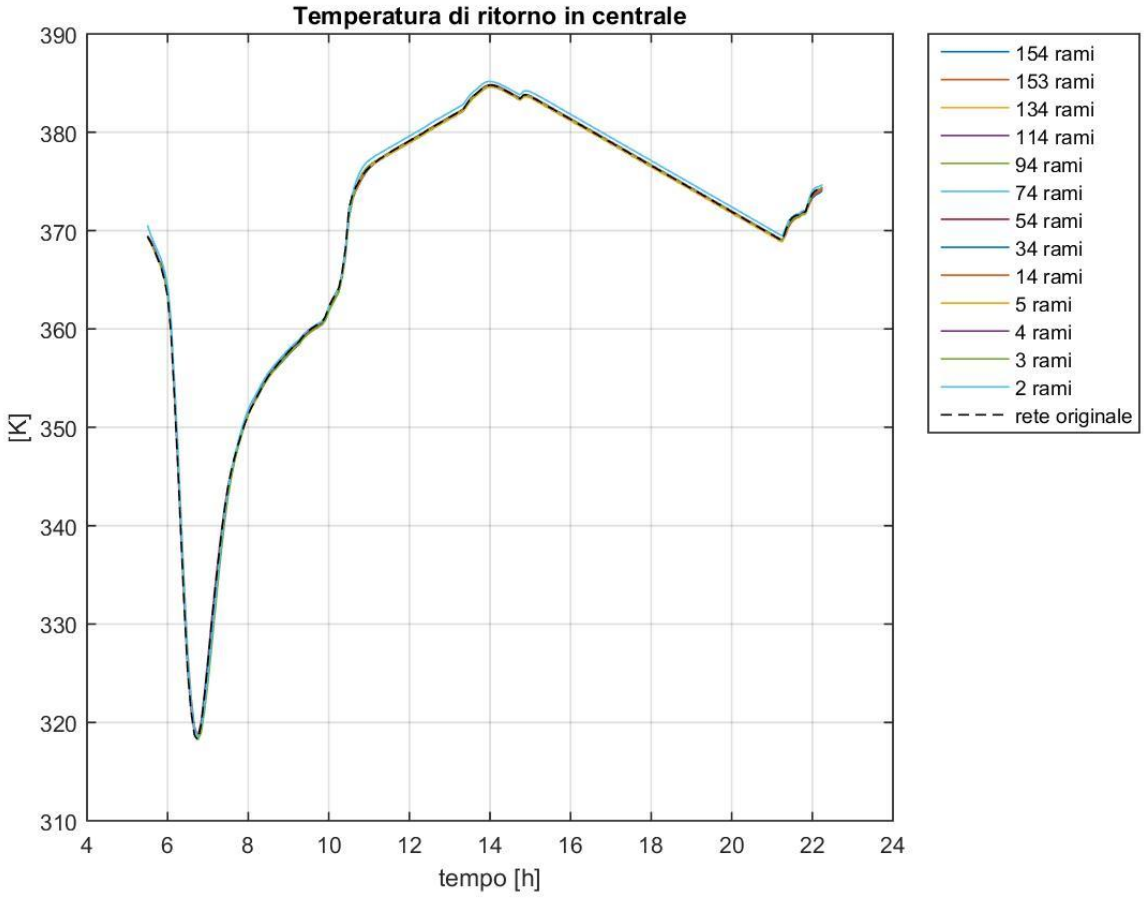


(Figura 4.22)

Come possiamo osservare, indipendentemente dal grado di compattazione, anche nelle reti equivalenti si verifica lo scostamento e l'attenuazione del picco di potenza da fornire.

Temperatura di ritorno nella rete di trasporto

Come già fatto per le simulazioni con portata variabile, qui di seguito mostriamo l'andamento temporale dei profili di temperatura del baricentro della rete di ritorno per alcuni stadi di aggregazione. Anche in questo caso, infatti, la temperatura di ritorno nella rete di trasporto è l'unica grandezza che può variare con la compattazione.

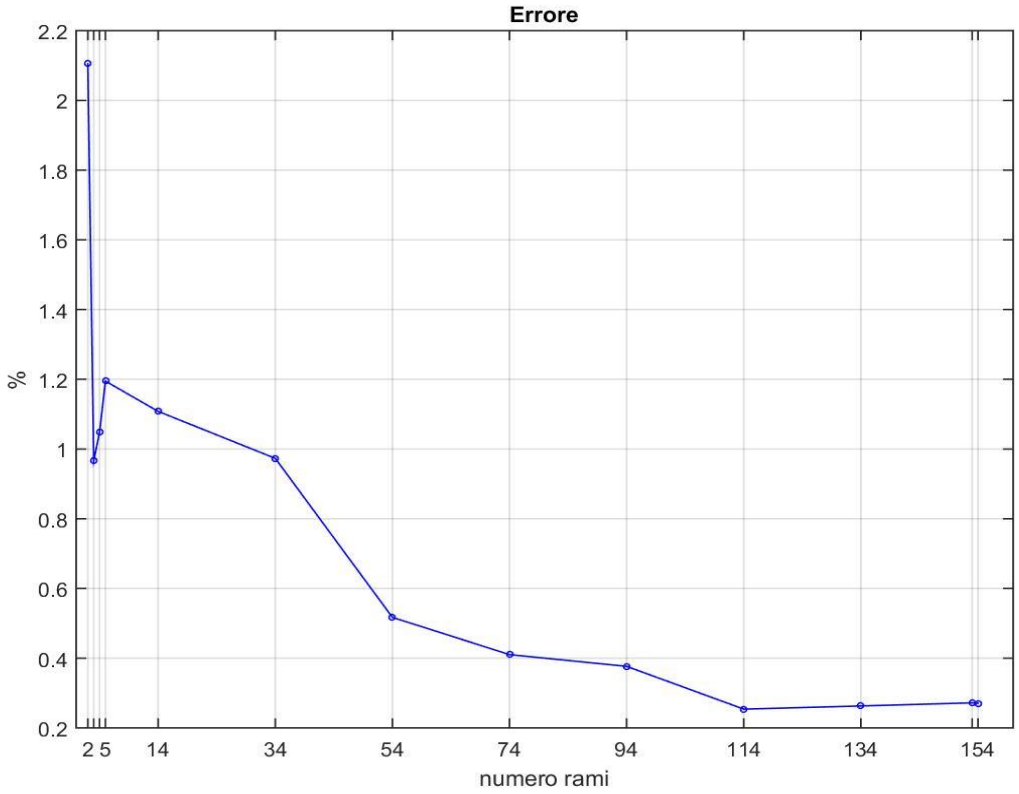


(Figura 4.23)

Come posiamo osservare, se escludiamo la linea riferita alla rete equivalente composta da due soli rami, i profili di temperatura delle reti equivalenti coincidono con quello della rete originale.

Errori

Quantificando l'errore sulla potenza da fornire mediante l'equazione 4.10 ed esprimendolo in percentuale utilizzando l'espressione 4.11, al variare del numero dei rami si ottiene l'andamento rappresentato nell'immagine qui di seguito.



(Figura 4.24)

Il grafico ci mostra che, dopo aver semplificato tutte le ramificazioni, si ottiene un errore di circa 0.2%. In seguito, al diminuire del numero dei rami, l'errore tende a crescere gradualmente ed oscilla nel tratto finale corrispondente ad un alto grado di compattazione.

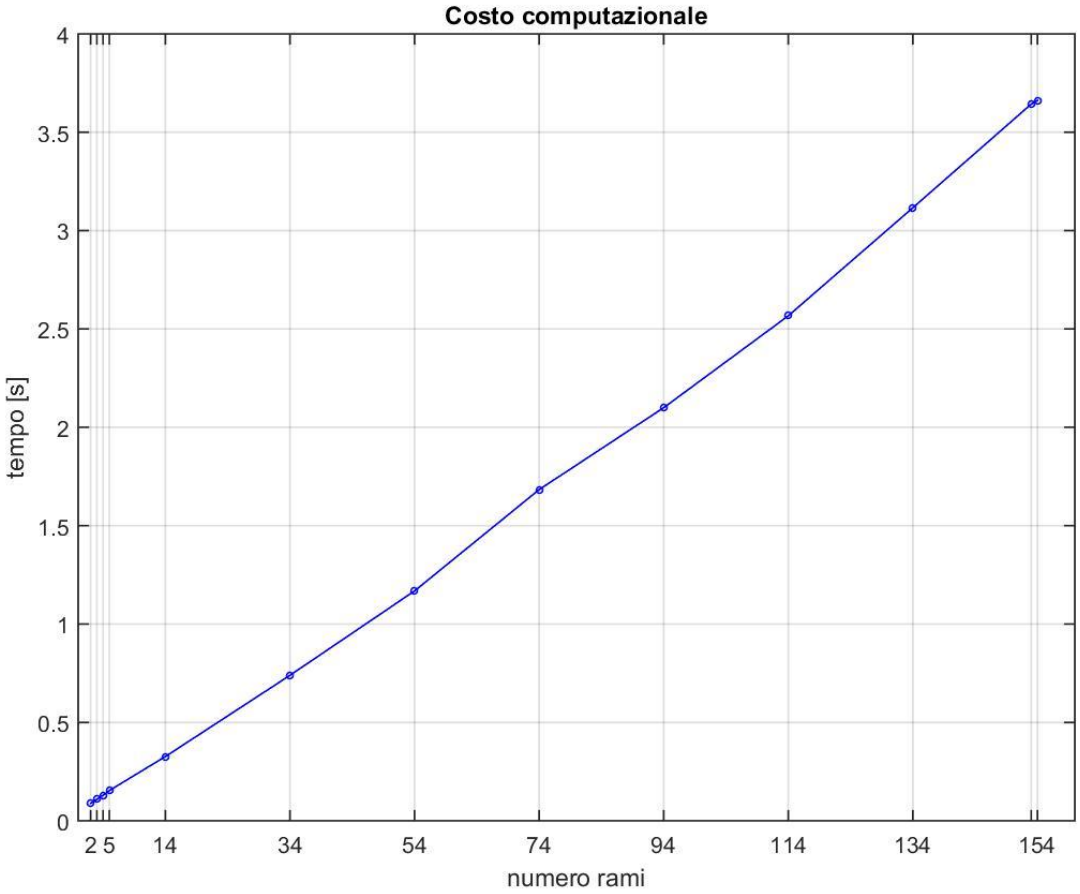
Come è possibile osservare nel paragrafo 3.3.3, i valori delle lunghezze, dei volumi, dei diametri interni, dei diametri esterni e dei coefficienti di scambio termico ottenuti dopo la semplificazione delle ramificazioni sono funzioni dalla variabile ausiliare α che, come riportato nella tabella 3.1, dipende dalle portate estratte e di conseguenza dalle portate che percorrono i rami.

Pertanto le caratteristiche della rete equivalente dipendono anche dalle portate che scorrono nei condotti nell'istante di tempo t* scelto come riferimento per la compattazione. E' inoltre possibile ipotizzare che, in un determinato istante di tempo t, la rete equivalente riesce a descrivere il comportamento della rete originale tanto meglio quanto più le portate sulla rete reale si avvicinano a quelle presenti nell'istante di tempo t*.

A conferma di ciò possiamo notare come nella figura 4.24 (relativa a delle simulazioni effettuate con portata costante nel tempo e pari a quella usata come input nel modello di compattazione) gli errori sono minori rispetto a quelli mostrati nella figura 4.18 (che si riferisce a delle simulazioni con portate variabili).

Costo computazionale

Il grafico sottostante mostra il tempo impiegato dal computer (CPU da 2.2 GHz) nella risoluzione del problema fluidodinamico e del problema termico in funzione del numero dei rami della rete equivalente.



(Figura 4.25)

L'andamento qualitativo è analogo a quello della figura 4.19: lineare e decrescente con il progredire della compattazione. Di conseguenza, ancora una volta, sembra che il costo computazionale sia direttamente proporzionale al numero dei rami e quindi anche al numero dei nodi.

Confrontando questo grafico con quello della figura 4.19 possiamo notare che in generale il costo computazionale è minore. Ciò accade in quanto , essendo la portata costante con il tempo, l'equazione 2.37 è stata risolta una volta soltanto per ogni rete aggregata; nel caso delle simulazioni a portata variabile, invece, è necessario risolvere il problema fluidodinamico in ogni istante di tempo considerato.

5 CONCLUSIONI E SVILUPPI FUTURI

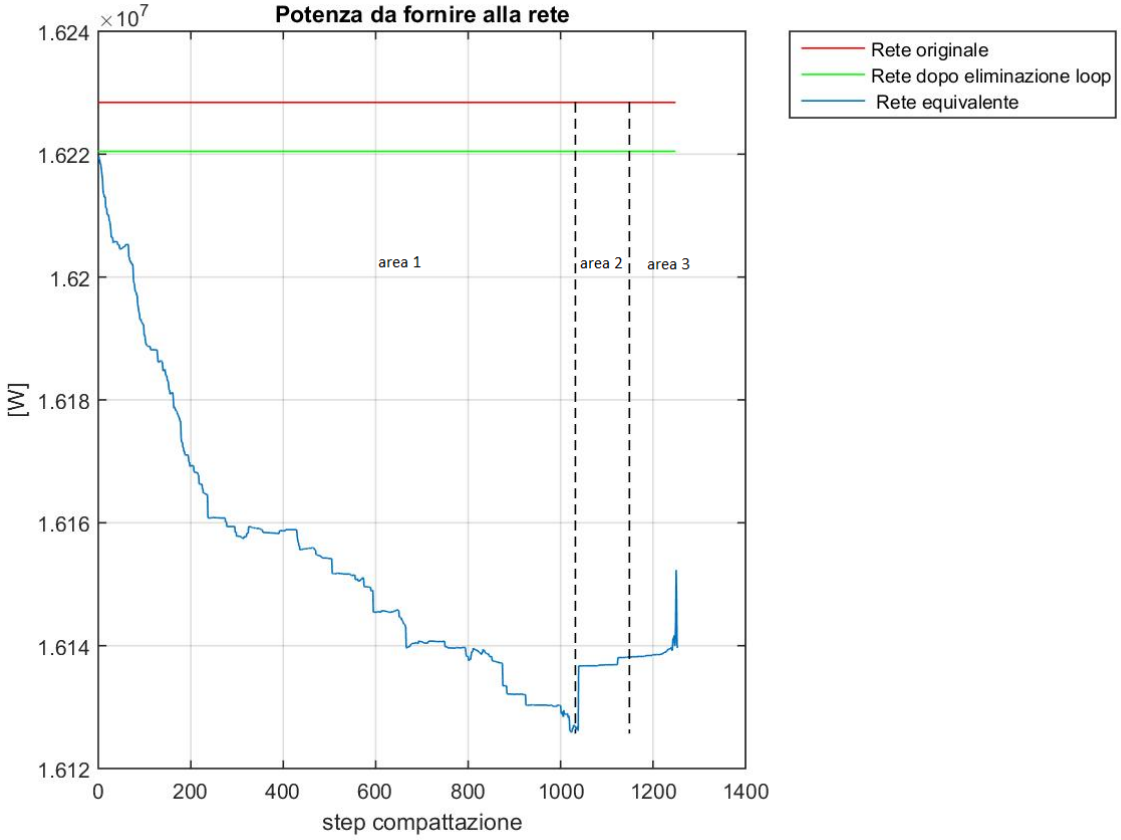
In questo lavoro è stato presentato ed utilizzato il modello di aggregazione danese nel quale, al fine di ridurre il costo computazionale relativo alla risoluzione del problema termico e del problema fluidodinamico, le reti di trasporto e distribuzione reali vengono semplificate generando delle reti equivalenti con un numero esiguo di nodi e di rami.
Per verificare l'efficacia di questo modello di aggregazione è stato effettuato un confronto tra i risultati ottenuti risolvendo il problema termico e fluidodinamico sulla rete originale e quelli ottenuti sulle reti semplificate equivalenti.

Applicando il modello di aggregazione danese alla rete BCT408 in condizioni stazionarie, accettando un errore sulla potenza da fornire alla rete del 0.2% ed uno sulle dispersioni termiche attraverso la stessa del 7-8%, è possibile ridurre significativamente il numero di nodi (e di rami) e con esso il costo computazionale.
Sebbene il modello di aggregazione esposto nel capitolo 3 sia stato costruito considerando un comportamento stazionario della rete, è possibile utilizzare le reti aggregate equivalenti per studiare la rete originale in condizioni transitorie. I risultati riguardanti lo studio della rete BCT408 in condizioni transitorie, infatti, mostrano che, passando da 154 a 3 rami, si ottiene una riduzione del costo computazionale del 98.4% (si passa da 6.4 a 0.1 secondi) a fronte di un errore percentuale di potenza in uscita dalla centrale di generazione (definito dall'equazione 4.11) pari a circa 1.6%.
Nel paragrafo 4.4 si è inoltre verificato che anche le reti equivalenti riescono a descrivere il fenomeno del ritardo temporale del fronte di temperatura che si ha quando la temperatura all'uscita dalle sottostazioni varia bruscamente.
Possiamo quindi concludere che il modello di aggregazione presentato, al prezzo di un errore relativamente piccolo, riesce ad abbattere il costo computazionale.

Il limite principale del metodo di aggregazione utilizzato in questo lavoro consiste nell'impossibilità di compattare reti con loop in quanto non è prevista alcuna semplificazione per le strutture in cui due o più portate non esterne convergono in un unico nodo.
Nella risoluzione del problema fluidodinamico di reti con circuiti chiusi, a differenza di come è stato fatto per la rete BCT408, non è possibile applicare esclusivamente la conservazione della massa, ma bisogna legare questa alla conservazione della quantità di moto utilizzando il SIMPLE ALGHORITM all'interno del quale troviamo due cicli iterativi. Risulterebbe dunque molto vantaggioso poter eliminare i loop in modo da evitare l'utilizzo di metodi iterativi che, dal punto di vista computazionale, sono molto dispendiosi.

Una possibile soluzione consiste nell'eliminazione di un ramo e nell'aggiunta di due portate esterne per ogni circuito chiuso.
Supponendo di eliminare il ramo j avente in ingresso il nodo i-1 ed in uscita il nodo i, si può aggiungere una portata entrante nel nodo i di valore $-G_j$ ed una uscente dal nodo i-1 di valore G_j.

Nella scelta del ramo da eliminare in ogni loop, al fine di apportare un errore più piccolo possibile, si seleziona il ramo con la portata minore tra tutti quelli costituenti il loop stesso.
Adottando questa soluzione per la rete di distribuzione BCT414 si sono ottenuti dei buoni risultati per il caso stazionario. Non è stato invece possibile effettuare uno studio in transitorio in quanto, nei rami che nella rete originale formano i loop, vi è spesso una variazione del verso di percorrenza del fluido; di conseguenza alcuni nodi che in un certo instante di tempo sono dei nodi di divergenza (nodi dai quali si diramano due o più portate non esterne), in altri istanti di tempo diventano dei nodi di convergenza (nodi nei quali confluiscono due o più portate non esterne).
Poichè durante il processo di aggregazione è fondamentale individuare i nodi di divergenza e su questi si effettuano delle precise operazioni (vedere capitolo 3), la rete aggregata equivalente, ottenuta dalla semplificazione della rete originale alle condizioni di funzionamento relative all' istante t, ha caratteristiche molto diverse da quella che si otterrebbe considerando un istante di tempo nel quale delle portate invertono la loro direzione.
Qui di seguito sono mostrati i risultati ottenuti dallo studio in condizioni stazionarie della rete BCT414 presentante 4 loop e quelli relativi alle simulazioni effettuate sulle reti equivalenti generate dal modello di aggregazione.
Come già fatto per la rete BCT408, anche in questo caso il coefficiente di scambio termico lineare pari a 0.06 W/m/K è stato aumentato con l'intento di mettere in evidenza le dispersioni termiche



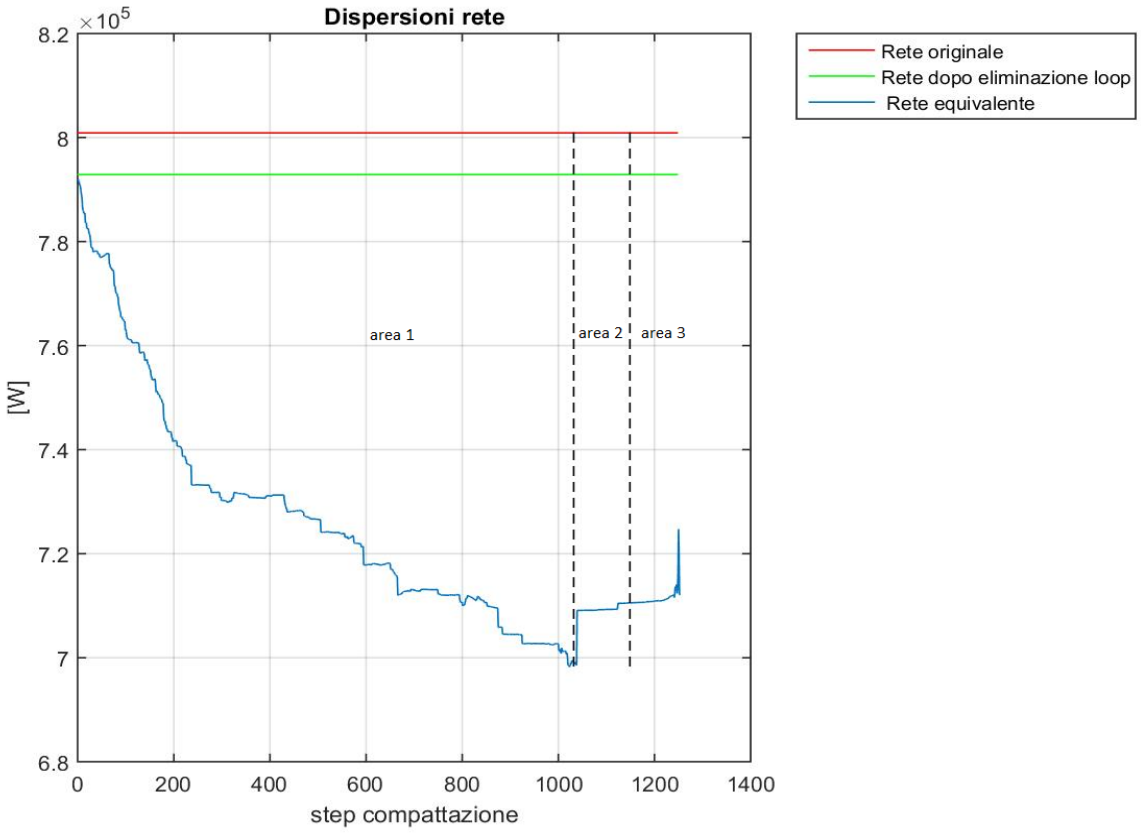
(Figura 5.1)

Nella figura 5.1 la linea rossa rappresenta la potenza che la centrale di generazione deve fornire alla rete originale per soddisfare le richieste delle utenze, mentre la linea verde mostra la potenza

che deve essere fornita alla rete ottenuta dopo l'apertura dei circuiti chiusi mediante l'eliminazione dei rami con portata minore. In blu invece troviamo l'andamento con il numero dei rami della potenza da fornire calcolata sulle reti equivalenti.
Come già fatto per la rete BCT 408, il grafico è stato suddiviso in 3 aree:

- nell'area 1 sono state semplificate le ramificazioni (vedere capitolo 3.3).
  La maggior parte degli step di compattazione della rete è dedicata al passaggio dalla struttura originale ad albero ad una struttura lineare.
  In questa sezione del grafico lo scostamento tra la linea blu e quella rossa aumenta con il grado di compattazione della rete;
- nell' area 2 si ha l'eliminazione dei rami che non servono alcun utenza.
  Escludendo il gradino presente circa allo step 1100, l'andamento della linea blu è pressoché costante;
- nell'area 3 si effettua un' ulteriore eliminazione dei rami rimasti; di conseguenza avviene anche un'aggregazione delle utenze.
  Anche per la rete BCT414, proprio come accade per la rete BCT408, nel tratto finale si ha una forte oscillazione della potenza da fornire.

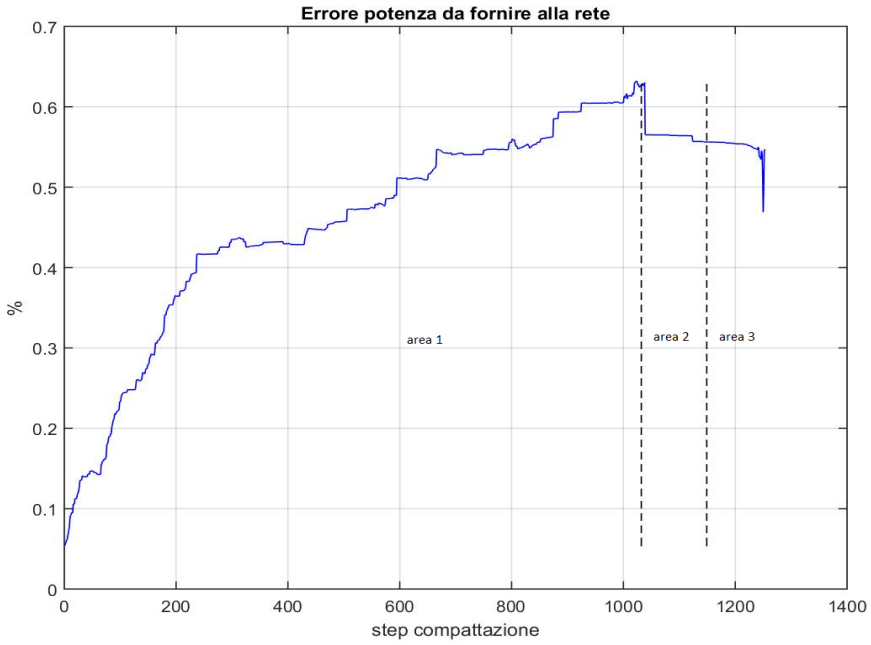
Nell'immagine sottostante sono rappresentati i risultati relativi alle dispersioni termiche attraverso le reti di distribuzione.



(Figura 5.2)

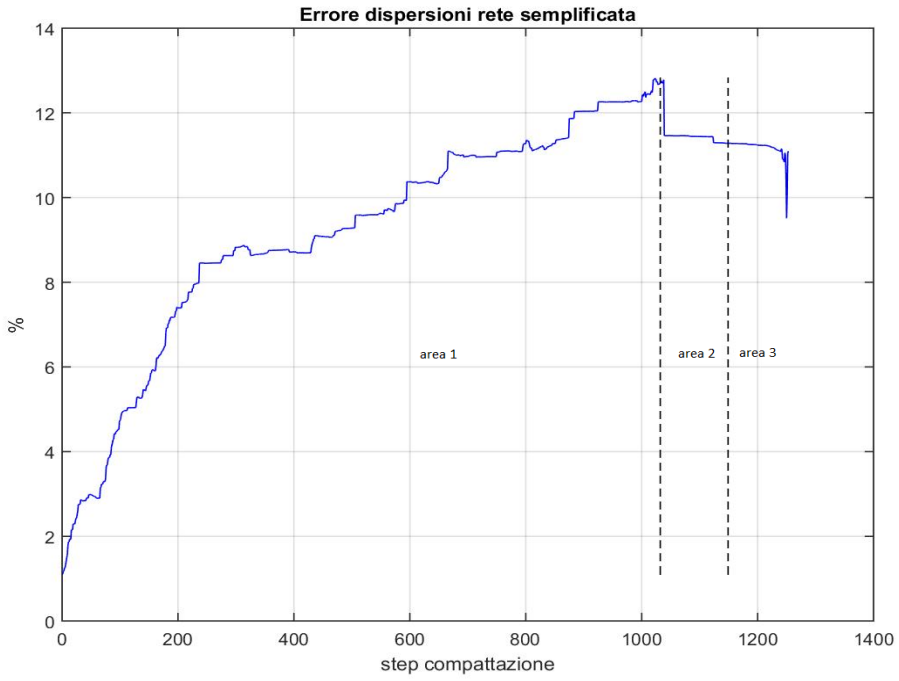
Siccome il carico totale della rete non varia durante la compattazione, l'andamento qualitativo delle curve è analogo a quello presente nella figura 5.1 .

Nella figura 5.3 è riportato l'andamento al variare del grado di compattazione dello scostamento percentuale della potenza da fornire calcolata sulle reti equivalenti, rispetto alla potenza calcolata sulla rete originale.



(Figura 5.3)

Nella figura 5.4 è riportato l'andamento al variare del grado di compattazione dello scostamento percentuale della dispersione termica calcolata sulle reti equivalenti, rispetto a quella calcolata sulla rete originale.



(Figura 5.4)

Possibili studi futuri sul metodo di aggregazione danese, potrebbero riguardare quindi l'adattamento di questo alle reti con loop, anche per simulazioni in condizioni transitorie.

APPENDICE 1: BACKWARD EULER METHOD

Nell'ambito della computazione scientifica il *backward Euler method* è uno dei metodi più usati per risolvere le equazioni differenziali ordinarie .
Consideriamo quindi la seguente equazione differenziale

$$\dot{y}(t) = f(y,t) \qquad \text{(A1.1)}$$

Dove

- t indica l'ascissa che nel nostro caso è il tempo
- f è una funzione dipendente da y e dal tempo t
- y è la soluzione dell'equazione differenziale

Indicando con t_n il tempo allo step n, Δt l'intervallo di tempo tra due step successivi e applicando l'espansione di Taylor fermandosii al primo ordine:

$$y(t_n) = y(t_{n+1} - \Delta t) = y(t_{n+1}) - \Delta t * \frac{dy(t_{n+1})}{dt} + O(h^2) \qquad \text{(A1.2)}$$

La derivata che compare nell'equazione A1.1 può essere espressa mediante l'equazione A1.2 con

$$\frac{dy(t_{n+1})}{dt} = \frac{y(t_{n+1}) - t_n}{\Delta t} = f(y(t_{n+1}), t_{n+1}) \qquad \text{(A1.3)}$$

Semplificando la scrittura con $y(t_n) = y_n$ e considerando le equazioni A1.2 e A1.3 si ha

$$y_n = y_{n+1} - \Delta t * f(y_{n+1}, t_{n+1}) \rightarrow y_{n+1} = y_n + \Delta t * f(y_{n+1}, t_{n+1}) \qquad \text{(A1.4)}$$

Quest'equazione è implicita in quanto la funzione f dipende dalla soluzione y_{n+1}.
Nel caso in cui f non sia lineare la risoluzione dell'equazione A1.4 può essere complessa.

In realtà anche il *forward Euler method* può essere usato per calcolare la soluzione dell'equazione A1.1. Nel foreward Euler method, per il calcolo di y(n+1) si parte da

$$y(t_{n+1}) = y(t_n + \Delta t) \qquad \text{(A1.5)}$$

E, con gli stessi passaggi che sono stati fatti per il *backward Euler method*, si ottiene la seguente formula esplicita

$$y_{n+1} = y_n + \Delta t * f(y_n, t_n) \qquad \text{(A1.6)}$$

Nello svolgimento della tesi è stato utilizzato il *backward Euler method* in quanto sia nel problema termico che in quello fluidodinamico, le funzioni f sono lineari e quindi i calcoli non sono troppo complessi ed inoltre , al contrario di ciò che accade con l'utilizzo del *forward Euler method* , la soluzione è stabile per qualsiasi Δt scelto.

A titolo di esempio applichiamo ora il *backward Euler method* all'equazione 2.33 ottenendo così l'equazione 2.34
L'equazione di partenza è quindi:

$$\frac{d\rho}{dt} * \Delta V_i + \sum_j G_j + G_{ext} = 0 \qquad \text{(A1.7)}$$

Ponendo $f = -\frac{\sum_j G_j + G_{ext}}{\Delta V_i}$ si ha:

$$\frac{d\rho}{dt} = f \qquad \text{(A1.8)}$$

L'equazione A1.8 è analoga all'equazione A1.1.

Applicando quindi il *backward Euler method*:

$$\rho_{n+1} = \rho_n + \Delta t * f(\rho_{n+1}, t_{n+1}) \quad \text{(A1.9)}$$

Tenendo conto dell'espressione di f si ha

$$\rho_{n+1} = \rho_n - \Delta t * \frac{\sum_j G_{j,n+1} + G_{ext,n+1}}{\Delta V_i} \quad \text{(A1.10)}$$

L'equazione A1.10 può infine essere riscritta come segue

$$\frac{\rho_{n+1} - \rho_n}{\Delta t} * \Delta V_i + \sum_j G_{j,n+1} + G_{ext,n+1} = 0 \quad \text{(A1.11)}$$

E' quindi stata ottenuta un'espressione analoga alla 2.34 in cui invece degli apice t e t-Δt si hanno rispettivamente i pedici n+1 ed n.

APPENDICE 2: FIXED POINT METHOD

L'algoritmo SIMPLE per un sistema studiato in condizioni stazionarie richiede la risoluzione della seguente equazione ad ogni iterazione

$$\boldsymbol{G}^* = \boldsymbol{Y}^* * \boldsymbol{A}^T * \boldsymbol{P}^* + \boldsymbol{Y}^* * \boldsymbol{t} \qquad (A2.1)$$

Y dipende da **G**, perciò bisogna adottare un metodo numerico per calcolare **G***.

L'equazione A2.1 può essere riformulata nel seguente modo

$$\boldsymbol{G}^* = \boldsymbol{\psi}(\boldsymbol{G}^*) \qquad (A2.2)$$

Dove $\boldsymbol{\psi}(\boldsymbol{G}^*) = \boldsymbol{Y}(\boldsymbol{G}^*) * \boldsymbol{A}^T * \boldsymbol{P}^* + \boldsymbol{Y}(\boldsymbol{G}^*) * \boldsymbol{t}$ (A2.3)

In questo lavoro è stato utilizzato il fixed-point method che può essere espresso da

$$\boldsymbol{G}^*_{k+1} = \boldsymbol{\psi}(\boldsymbol{G}^*_k) \quad \text{con k=0,1,2…} \qquad (A2.4)$$

Dall'equazione precedente si può notare come questo metodo sia iterativo: ad ogni iterazione k+1 si valuta il vettore delle portate $\mathbf{G}^*_{k+1}$ utilizzando quello calcolato durante l'iterazione precedente $\boldsymbol{G}^*_k$. Per poter utilizzare questo algoritmo, al fine di avere un vettore di portate in input per la prima iterazione, bisogna ipotizzare un $\boldsymbol{G}^*_0$ in modo arbitrario.

Se $\boldsymbol{G}^*_0$ non è scelto in modo appropriato, il processo iterativo di cui sopra potrebbe non convergere. Per evitare quindi problemi di convergenza è possibile utilizzare delle sotto-relazioni: l'equazione A2.4 viene sostituita da:

$$\boldsymbol{G}^*_{k+1} = \boldsymbol{\psi}(\boldsymbol{G}^*_r) \qquad (A2.5)$$

Dove $\boldsymbol{G}^*_r$ viene calcolato partendo dalle portate calcolate nelle due iterazioni precedenti

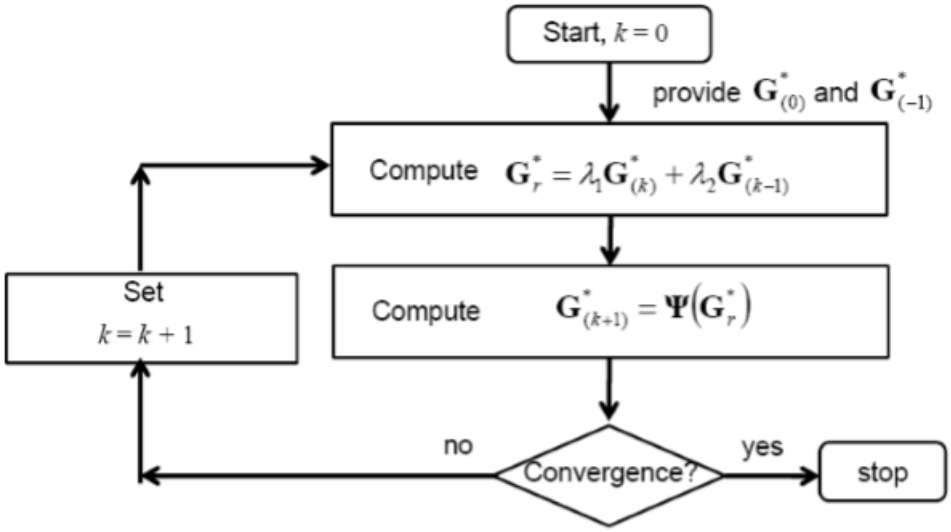
$$\boldsymbol{G}^*_r = a_1 * \boldsymbol{G}^*_k + a_2 * \boldsymbol{G}^*_{k-1} \qquad (A2.6)$$

Dove a_1 e a_2 rispettano la seguente condizione

$$a_1 + a_2 = 1 \qquad (A2.7)$$

Come possiamo notare dall'equazione A2.6, al fine di avere gli input necessari per calcolare $\boldsymbol{G}^*_r$ sin dalla prima iterazione, bisogna supporre due vettori di portate iniziali: $\boldsymbol{G}^*_0$ e $\boldsymbol{G}^*_{-1}$.

Qui di seguito troviamo il flow-chart relativo al *fixed point method*



(Figura A2.1)

APPENDICE 3: CODICE

SIMULAZIONE IN STAZIONARIO PER LA RETE ORIGINALE

```
clear all
close all
clc

load DatiPerNicolaBCT408
load A_modificata; n_nodi=size(A,1); n_rami=size(A,2);
L=NetData.L; D=NetData.D;
MGext=BC.b; MTout_ut=BC.T; nt=size(MGext,1);
h=1.2; %[W/m/K]
TG=10+273.15; %temperatura del terreno [K]
H=h*L; %coeff di dispersione termica [W/K]
cp_w=4186; % [J/Kg/K]
TBC_in=120+273.15;
%individuo inlet ed outlet node
inlet_node_m=NetData.BCT;
z=0;
for i=1:n_nodi
    if isempty(find(A(i,:)==1))==1
       %allora il nodo i non ha rami uscenti
       z=z+1;
       outlet_node_m(z)=i;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;

% individuo il numero massimo di utenze accese contemporaneamente
for i=1:nt
   utenze_accese=find(MGext(i,:)>0);
   n_utenze_accese(i)=length(utenze_accese);
end

%trovo gli outlet node in cui non ho mai utenze accese
z=0;
for i=1:length(outlet_node_m)
    w=outlet_node_m(i);
    if isempty(find(MGext(:,w)))
       % allora l'utenza è sempre spenta
       z=z+1;
       utenza_sempre_spenta(z)=w;
    end
end


%tra i vari istanti di tempo in cui il numero di utenze accese è massimo
seleziono quello con portate maggiori
Gtot_max=0;
for i=1:nt
   Gext=MGext(i,:);
   Gtot=sum(Gext);
   if (Gtot>Gtot_max) & (length(find(Gext))==max(n_utenze_accese))
      Gtot_max=Gtot;
      t_Gmax=i;
   end
end
```

```
%Quindi prendiamo tutti i dati al tempo t_Gmax
%a) mandata
Gext_m=MGext(t_Gmax,:)'; Gext_m(inlet_node_m)=-sum(Gext_m);
A_m=A;
%b) ritorno
Gext_r=-Gext_m; A_r=-A_m;
%altri dati
phi=BC.Phi(t_Gmax,:);


%|||||||| RISOLUZIONE PROBLEMA FLUIDODINAMICO MANDATA ||||||||||||||||
G_m=A_m\(-Gext_m);
for i=1:n_rami
   if G_m(i)<0.0001
      G_m(i)=0;
   end
end


rami_G0=find(G_m==0);
for i=1:length(rami_G0)
   nodi_out_rami_G0(i)=find(A_m(:,rami_G0(i))==-1);  %sono  i  nodi  in  cui
entrano i rami a portata nulla
end

%|||||||| RISOLUZIONE PROBLEMA TERMICO MANDATA ||||||||||||
%costruzione di K e di f
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for i=1:n_nodi
    ramo_in=find(A_m(i,:)==-1); %ramo in ngreso al nodo
    rami_out=find(A_m(i,:)==1); %rami in uscita dal nodo
    rami_coll=[ramo_in rami_out]; %rami collegati al nodo
    %calcolo la portata uscente dal nodo (non al contorno)
    Gsum_out=0;
    for j=1:length(rami_out)
        Gsum_out=Gsum_out+G_m(rami_out(j));
    end
    %calcolo la sommatoria dei coeff di scambio termico dei rami collegati
al nodo i
    Hsum=0;
    for j=1:length(rami_coll)
        Hsum=Hsum+H(rami_coll(j));
    end
    %assegno i valori agli elementi di K e f
    k(i,i)=cp_w*Gsum_out+0.5*Hsum;
    if isempty(ramo_in)==0
      nodo_prec=find(A_m(:,ramo_in)==1); %nodo precedente al nodo i
      k(i,nodo_prec)=-cp_w*G_m(ramo_in);
    end
    f(i)=0.5*Hsum*TG;
end
%imposizione della condizione al contorno al nodo di ingresso
k(inlet_node_m,:)=zeros(1,n_nodi); k(inlet_node_m,inlet_node_m)=1;
f(inlet_node_m)=TBC_in; % temperatura in ingresso nel baricentro [K]
%imposizione delle condizioni al contorno ai nodi in uscita
for i=1:length(outlet_node_m)
    j=outlet_node_m(i); % chiamo con j il nodo di uscita in esame
    if Gext_m(j)~=0
       ramo_in=find(A_m(j,:)==-1); %ramo in ngreso al nodo j
```

```
       nodo_prec=find(A_m(:,ramo_in)==1); %nodo precedente al nodo j
       k(j,j)=-cp_w*G_m(ramo_in)-H(ramo_in)/2;
       k(j,nodo_prec)=cp_w*G_m(ramo_in)-H(ramo_in)/2;
       f(j)=-H(ramo_in)*TG;
    else
       k(j,j)=1; f(j)=TG;
    end
end

T_m=k\f;

%identifico sul disegno i nodi uscenti da rami a portata nulla
X=NetData.X; Y=NetData.Y;
figure(1)
hold on
grid on
box on
for i=1:n_nodi
    rami_in=find(A(i,:)==-1);  %vettore  contenente  gli  indici  dei  rami  in
ingresso al nodo i
    n_in=length(rami_in); % numero di rami in ingresso
    for j=1:n_in;
       w=rami_in(j);
       nodo_prec=find(A(:,w)==1);
       plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'k');
    end
    if isempty(find(nodi_out_rami_G0==i))==0
       scatter(X(i),Y(i),'r');
    end
end

clear ramo_in rami_out rami_coll nodo_prec
%|||||||| RISOLUZIONE PROBLEMA FLUIDODINAMICO RITORNO ||||||||||||||||
G_r=A_r\(-Gext_r);
for i=1:n_rami
   if G_r(i)<0.0001
      G_r(i)=0;
   end
end
%|||||||| RISOLUZIONE PROBLEMA TERMICO RITORNO ||||||||||||
%trovo Tut_out dal carico
Tut_out=zeros(n_nodi,1);
for i=1:length(outlet_node_m)
    w=outlet_node_m(i);  %  chiamo  w  il  nodo  in  uscita  della  mandata
considerato
    if Gext_m(w)~=0
       Tut_out(w)=T_m(w)-phi(w)/Gext_m(w)/cp_w;
    else
       Tut_out(w)=T_m(w); % per le utenze spente
    end
end
%costruzione di K e di f
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for i=1:n_nodi
    rami_in=find(A_r(i,:)==-1); %rami in ingreso al nodo i
    ramo_out=find(A_r(i,:)==1); %ramo in uscita al nodo i
    rami_coll=[rami_in ramo_out]; %rami collegati al nodo i
    %calcolo la sommatoria dei coeff di scambio termico dei rami collegati
al nodo i
```

```
    Hsum=0;
    for j=1:length(rami_coll)
        Hsum=Hsum+H(rami_coll(j));
    end
    %assegno i valori non nulli agli elementi di k e di f
    if isempty(ramo_out)==0
       k(i,i)=cp_w*G_r(ramo_out)+0.5*Hsum;
    end
    for j=1:length(rami_in)
        nodo_prec=find(A_r(:,rami_in(j))==1);
        k(i,nodo_prec)=-cp_w*G_r(rami_in(j));
    end
    f(i)=0.5*Hsum*TG;
end

%imposizione della condizione al contorno ai nodi di ingresso
for i=1:length(inlet_node_r)
    j=inlet_node_r(i);
    k(j,:)=zeros(1,n_nodi); k(j,j)=1; f(j)=Tut_out(j);
end

%imposizione della condizione al contorno al nodo di uscita
ramo_in=find(A_r(outlet_node_r,:)==-1);    %ramo    in    ingreso    al    nodo
outlet_node_r
nodo_prec=find(A_r(:,ramo_in)==1); %nodo precedente al nodo outlet_node_r
k(outlet_node_r,outlet_node_r)=-cp_w*G_m(ramo_in)-H(ramo_in)/2;
k(outlet_node_r,nodo_prec)=cp_w*G_m(ramo_in)-H(ramo_in)/2;
f(outlet_node_r)=-H(ramo_in)*TG;

%trovo T_r
T_r=k\f; Tout_r=T_r(outlet_node_r);
Q_staz=cp_w*abs(Gext_m(inlet_node_m))*(T_m(inlet_node_m)-
T_r(outlet_node_r));
disp_staz=Q_staz-sum(phi); %dispersioni nel caso stazionario
save('t_Gmax','t_Gmax');
G_staz=G_m;
save('G_staz','G_staz');
save('T_m_nodale','T_m');
save('T_r_nodale','T_r');



%cerco le dispersioni della sola rete di mandata
disp_staz_m=0;
for i=1:n_rami
    ramo=i;
    nodi_coll=find(A_m(:,ramo));
    disp_staz_m=disp_staz_m+abs(cp_w*G_m(ramo)*(T_m(nodi_coll(1))-
T_m(nodi_coll(2))));
end

save('disp_staz_nod','disp_staz');
save('disp_staz_nod_m','disp_staz_m');
save('Q_staz_nod','Q_staz');
save('G_m_staz','G_m');
```

RIMOZIONE RAMIFICAZIONI

```
function
[A,L,d_int,d_est,tao,phi,G,H,sez_int,sez_est,Vol_int,Vol_est,V_p]=rimozione_
ramificazioni(A,L,d_int,d_est,tao,sez_int,sez_est,Vol_int,Vol_est,V_p,phi,G,
rho_w,H,div_node,T,m_o_r)
n_nodi=size(A,1);
cp_w=4186; cp_p=502; %[J/Kg/K]
rho_p=7750; %[kg/m3]
TG=10+273.15;

j=div_node;
%trovo i rami uscenti dal nodo j preso in esame
rami_out=find(A(j,:)==1);
%considero solo due rami in uscita ( aggrego solo due rami alla volta
rami_out=rami_out(1,1:2);
%ordino  rami_out in modo da avere nella posizione 1 quello con il tao
minore
if tao(rami_out(2))<tao(rami_out(1))
   vv=rami_out(1); rami_out(1)=rami_out(2); rami_out(2)=vv;
end
i1=rami_out(1); i2=rami_out(2); %indicatori dei rami
%a  questo  punto  il  ramo  rami_out(1)  deve  essere  inglobato  nel  ramo
rami_out(2)
%becco i nodi che si ricombinano in modo diverso( da modificare)
nodo_mod=find(A(:,i1)==-1); % nodo da cambiare nella matrice A oltre a
quello della divergenza
%modifico la posizione dei nodi
A(j,i2)=0;
A(nodo_mod,i2)=1;

%dopo  aver  cambiato  la  disposizione  dei  rami  e  dei  nodi  calcolo  le
caratteristiche equivalenti

alfa=G(i2)/G(i1);
beta=1+alfa;
csi=phi(i2)/phi(i1);
gamma=alfa*Vol_int(i1)/Vol_int(i2);

G(i1)=G(i1)+G(i2);

V_A=Vol_int(i1)*(1+alfa*csi);
L_A=L(i1)*(1+alfa*alfa*csi*sez_int(i1)/sez_int(i2))/beta;
A_A=V_A/L_A;
d_A=2*(A_A/pi)^0.5;
phi_A=(phi(i1)+alfa*phi(i2))/(1+alfa);
D_A=d_A*(1+rho_w*cp_w/rho_p/cp_p*(1-phi_A)/phi_A)^0.5;

L_B=L(i2)*(1-gamma*csi);
V_B=Vol_int(i2)*(1-gamma*csi);
tao_B=tao(i2)-tao(i1);

Vol_int(i1)=V_A;  L(i1)=L_A;  sez_int(i1)=A_A;  d_int(i1)=d_A;  d_est(i1)=D_A;
phi(i1)=phi_A;
L(i2)=L_B; Vol_int(i2)=V_B; tao(i2)=tao_B;

if H(i2)/H(i1)>=alfa
    H_A=(1+alfa)*H(i1); H_B=H(i2)-alfa*H(i1);
    H(i1)=H_A; H(i2)=H_B;
```

```
else
    if m_o_r==0
        %rete di mandata
        H_A=H(i1)+gamma*csi*H(i2); H_B=H(i2)*(1-gamma*csi);
        H(i1)=H_A; H(i2)=H_B;
    elseif m_o_r==1
        %rete di ritorno
        nodo_i1=find(A(:,i1)==1); nodo_i2=find(A(:,i2)==1);
        omega=(T(nodo_i2)-TG)/(T(nodo_i1)-TG);
        H_A=(1+alfa)*(H(i1)+H(i2)*gamma*csi*omega)/(1+alfa*omega);
H_B=H(i2)*(1-gamma*csi);
        H(i1)=H_A; H(i2)=H_B;
    end
end
```

RIMOZIONE 1

```
function
[A,L,d_int,d_est,tao,phi,G,H,sez_int,sez_est,Vol_int,Vol_est,V_p,G_ext,Tut_o
ut,G_f,MM1]=rimozione1(A,L,d_int,d_est,tao,sez_int,sez_est,Vol_int,Vol_est,V
_p,phi,G,G_ext,rho_w,H,Tut_out,G_f,m_o_r)

n_nodi=size(A,1);
cp_w=4186; cp_p=502; %[J/Kg/K]
rho_p=7750; %[Kg/m3]
X=rho_p*cp_p/rho_w/cp_w;

% se r2 è il primo ramo r1 non esiste---> imponiamo r2 come secondo ramo
n_nodi=size(A,1);
for i=1:n_nodi
    if isempty(find(A(i,:)==1)==1)
       ultimo_nodo=i;
    elseif isempty(find(A(i,:)==-1)==1)
       primo_nodo=i;
    end
end
ultimo_ramo=find(A(ultimo_nodo,:)==-1);
primo_ramo=find(A(primo_nodo,:)==1);

nodi_no_load=find(G_ext==0); n_out=nodi_no_load(1); r2=find(A(n_out,:)==-1);
if r2==primo_ramo
      nodo_succ=find(A(:,primo_ramo)==-1);
      ramo_succ=find(A(nodo_succ,:)==1);
      r2=ramo_succ;
end

n_ing=find(A(:,r2)==1);   n_out=find(A(:,r2)==-1);   r1=find(A(n_ing,:)==-1);
r3=find(A(n_out,:)==1);
nodo_prec_r1=find(A(:,r1)==1); nodo_succ_r3=find(A(:,r3)==-1);


m1=G(r1)-G(r2); m2=G(r2)-G(r3);
if (m1+m2)~=0;
   f_A=m2/(m1+m2); f_B=1-f_A;
else
   f_A=0.5; f_B=1-f_A;
end

g_AA=f_A*tao(r2)/(tao(r1)+f_A*tao(r2));
g_AB=tao(r1)/(tao(r1)+f_A*tao(r2));

g_BB=f_B*tao(r2)/(tao(r3)+f_B*tao(r2));
g_BA=tao(r3)/(tao(r3)+f_B*tao(r2));
if g_AA<0.0001
   g_AA=0;
end
if g_BB<0.0001
   g_BB=0;
end
G_A=G(r1)-g_AA*m1;                                    G_B=G(r2)-g_BA*m2;

L_A=L(r1)+f_A*L(r2);
phi_A=( 1+X*(V_p(r1)+f_A*V_p(r2))/(Vol_int(r1)+f_A*Vol_int(r2)) )^-1;
```

```
V_A=Vol_int(r1)*G_A/G(r1)*phi_A/phi(r1)+f_A*Vol_int(r2)*G_A/G(r2)*phi_A/phi(
r2);
V_A_p=V_A*(1-phi_A)/(phi_A*X);
d_A=2*(V_A/pi/L_A)^0.5;
D_A=2*((V_A+V_A_p)/(pi*L_A))^0.5;
H_A=H(r1)+f_A*H(r2);
tao_A=rho_w*V_A/(G_A*phi_A);

L_B=L(r3)+f_B*L(r2);
phi_B=(1+X*(V_p(r3)+f_B*V_p(r2))/(Vol_int(r3)+f_B*Vol_int(r2)))^-1;
V_B=Vol_int(r3)*G_B/G(r3)*phi_B/phi(r3)+f_B*Vol_int(r2)*G_B/G(r2)*phi_B/phi(
r2);
V_B_p=V_B*(1-phi_B)/X/phi_B;
d_B=2*(V_B/pi/L_B)^0.5;
D_B=2*((V_B+V_B_p)/pi/L_B)^0.5;
H_B=H(r3)+f_B*H(r2);
tao_B=rho_w*V_B/(G_B*phi_B);

G(r1)=G_A; L(r1)=L_A; phi(r1)=phi_A; Vol_int(r1)=V_A; Vol_est(r1)=V_A+V_A_p;
d_int(r1)=d_A; d_est(r1)=D_A; H(r1)=H_A; tao(r1)=tao_A;

G(r3)=G_B; L(r3)=L_B; phi(r3)=phi_B; Vol_int(r3)=V_B; Vol_est(r3)=V_B+V_B_p;
d_int(r3)=d_B; d_est(r3)=D_B; H(r3)=H_B; tao(r3)=tao_B;

if m_o_r==0;
    %cambio le temperature di uscita dall'utenza
    if (G_ext(nodo_prec_r1)+m1*g_AA)~=0

Tut_out(nodo_prec_r1)=(G_ext(nodo_prec_r1)*Tut_out(nodo_prec_r1)+m1*g_AA*Tut
_out(n_ing))/(G_ext(nodo_prec_r1)+m1*g_AA);
    else
       % Tut_out non cambia
    end

    if (m1*g_AB+m2*g_BA)~=0

Tut_out(n_ing)=(m1*g_AB*Tut_out(n_ing)+m2*g_BA*Tut_out(n_out))/(m1*g_AB+m2*g
_BA);
    else
       % Tut_out non cambia
    end

    if (m2*g_BB+G_ext(nodo_succ_r3))~=0

Tut_out(nodo_succ_r3)=(G_ext(nodo_succ_r3)*Tut_out(nodo_succ_r3)+m2*g_BB*Tut
_out(n_out))/(m2*g_BB+G_ext(nodo_succ_r3));
    else
        % Tut_out non cambia
    end
end
%correggiamo le portate in uscita verso i carichi
G_ext(n_ing)=m1*g_AB+m2*g_BA;
G_ext(nodo_prec_r1)=G_ext(nodo_prec_r1)+m1*g_AA;
G_ext(nodo_succ_r3)=G_ext(nodo_succ_r3)+m2*g_BB;
%portata fittizia

if r1==primo_ramo && m_o_r==0
   G_f=G_f+m1*g_AA;
end
```

```
%ramo r3 preceduto da n_in e non da n_out
A(n_ing,r3)=1; %A(n_out,r3)=0
%n_out non esiste più
A(n_out,:)=[]; G_ext(n_out)=[];
if m_o_r==0
   Tut_out(n_out)=[];
end
%r2 non esiste più
A(:,r2)=[]; G(r2)=[]; d_est(r2)=[]; d_int(r2)=[]; H(r2)=[];
L(r2)=[];    phi(r2)=[];    sez_est(r2)=[];    sez_int(r2)=[];    tao(r2)=[];
V_p(r2)=[];
Vol_est(r2)=[]; Vol_int(r2)=[];
%costruzione della matrice per la suddivisione delle portate
if m_o_r==0
   MM1=[nodo_prec_r1 n_ing n_out nodo_succ_r3 g_AA g_BB];
end
end
```

RIMOZIONE 2

```
function
[A,L,d_int,d_est,tao,phi,G,H,sez_int,sez_est,Vol_int,Vol_est,V_p,G_ext,MM2,G
_f,Tut_out]=rimozione2(A,L,d_int,d_est,tao,sez_int,sez_est,Vol_int,Vol_est,V
_p,phi,G,G_ext,rho_w,H,Tut_out,G_f,m_o_r)
cp_w=4186; cp_p=502; %[J/Kg/K]
rho_p=7750; %[Kg/m3]
X=rho_p*cp_p/rho_w/cp_w;

%il modello nn va applicato se r2 è l'ultimo ramo ( perchè r3 non
esisterebbe)
%il modello nn va applicato se r2 è il primo ramo ( perchè r1 non
esisterebbe)
%come primo passo becchiamo il primo ramo e l'ultimo ramo
n_nodi=size(A,1);
for i=1:n_nodi
    if isempty(find(A(i,:)==1)==1)
       ultimo_nodo=i;
    elseif isempty(find(A(i,:)==-1)==1)
       primo_nodo=i;
    end
end
ultimo_ramo=find(A(ultimo_nodo,:)==-1);
primo_ramo=find(A(primo_nodo,:)==1);
%evito che venga preso r2 come primo o ultimo ramo
r2=find(L==min(L));
if r2==ultimo_ramo
   nodo_prec=find(A(:,ultimo_ramo)==1);
   ramo_prec=find(A(nodo_prec,:)==-1);
   r2=ramo_prec;
elseif r2==primo_ramo
      nodo_succ=find(A(:,primo_ramo)==-1);
      ramo_succ=find(A(nodo_succ,:)==1);
      r2=ramo_succ;
end
n_out=find(A(:,r2)==-1); n_ing=find(A(:,r2)==1);
r1=find(A(n_ing,:)==-1); r3=find(A(n_out,:)==1);
nodo_prec_r1=find(A(:,r1)==1); nodo_succ_r3=find(A(:,r3)==-1);

m1=G(r1)-G(r2); m2=G(r2)-G(r3);
f_A=m2/(m1+m2); f_B=1-f_A;

g_AA=f_A*tao(r2)/(tao(r1)+f_A*tao(r2));
g_AB=tao(r1)/(tao(r1)+f_A*tao(r2));
g_BB=f_B*tao(r2)/(tao(r3)+f_B*tao(r2));
g_BA=tao(r3)/(tao(r3)+f_B*tao(r2));
if g_AA<0.0001
   g_AA=0;
end
if g_BB<0.0001
   g_BB=0;
end
G_A=G(r1)-g_AA*m1;
G_B=G(r2)-g_BA*m2;

L_A=L(r1)+f_A*L(r2);
phi_A=( 1+X*(V_p(r1)+f_A*V_p(r2))/(Vol_int(r1)+f_A*Vol_int(r2)) )^-1;
```

```
V_A=Vol_int(r1)*G_A/G(r1)*phi_A/phi(r1)+f_A*Vol_int(r2)*G_A/G(r2)*phi_A/phi(
r2);
V_A_p=V_A*(1-phi_A)/(phi_A*X);
d_A=2*(V_A/pi/L_A)^0.5;
D_A=2*((V_A+V_A_p)/(pi*L_A))^0.5;
H_A=H(r1)+f_A*H(r2);
tao_A=rho_w*V_A/(G_A*phi_A);

L_B=L(r3)+f_B*L(r2);
phi_B=(1+X*(V_p(r3)+f_B*V_p(r2))/(Vol_int(r3)+f_B*Vol_int(r2)))^-1;
V_B=Vol_int(r3)*G_B/G(r3)*phi_B/phi(r3)+f_B*Vol_int(r2)*G_B/G(r2)*phi_B/phi(
r2);
V_B_p=V_B*(1-phi_B)/X/phi_B;
d_B=2*(V_B/pi/L_B)^0.5;
D_B=2*((V_B+V_B_p)/pi/L_B)^0.5;
H_B=H(r3)+f_B*H(r2);
tao_B=rho_w*V_B/(G_B*phi_B);

G(r1)=G_A; L(r1)=L_A; phi(r1)=phi_A; Vol_int(r1)=V_A; Vol_est(r1)=V_A+V_A_p;
d_int(r1)=d_A; d_est(r1)=D_A; H(r1)=H_A; tao(r1)=tao_A;

G(r3)=G_B; L(r3)=L_B; phi(r3)=phi_B; Vol_int(r3)=V_B; Vol_est(r3)=V_B+V_B_p;
d_int(r3)=d_B; d_est(r3)=D_B; H(r3)=H_B; tao(r3)=tao_B;

if m_o_r==0
    %cambio le temperature di uscita dall'utenza
    if nodo_prec_r1==primo_nodo

Tut_out(nodo_prec_r1)=(G_f*Tut_out(nodo_prec_r1)+m1*g_AA*Tut_out(n_ing))/(G_
f+m1*g_AA);
    else

Tut_out(nodo_prec_r1)=(G_ext(nodo_prec_r1)*Tut_out(nodo_prec_r1)+m1*g_AA*Tut
_out(n_ing))/(G_ext(nodo_prec_r1)+m1*g_AA);
    end

Tut_out(n_ing)=(m1*g_AB*Tut_out(n_ing)+m2*g_BA*Tut_out(n_out))/(m1*g_AB+m2*g
_BA);

Tut_out(nodo_succ_r3)=(G_ext(nodo_succ_r3)*Tut_out(nodo_succ_r3)+m2*g_BB*Tut
_out(n_out))/(m2*g_BB+G_ext(nodo_succ_r3));
end
%correggiamo le portate in uscita verso i carichi
G_ext(n_ing)=m1*g_AB+m2*g_BA;
G_ext(nodo_prec_r1)=G_ext(nodo_prec_r1)+m1*g_AA;
G_ext(nodo_succ_r3)=G_ext(nodo_succ_r3)+m2*g_BB;
%portata fittizia
if r1==primo_ramo
   G_f=G_f+m1*g_AA;
end
%ramo r3 preceduto da n_in e non da n_out
A(n_ing,r3)=1; %A(n_out,r3)=0
%n_out non esiste più
A(n_out,:)=[]; G_ext(n_out)=[];
if m_o_r==0
   Tut_out(n_out)=[];
end
%r2 non esiste più
A(:,r2)=[]; G(r2)=[]; d_est(r2)=[]; d_int(r2)=[]; H(r2)=[];
```

```
L(r2)=[];    phi(r2)=[];    sez_est(r2)=[];    sez_int(r2)=[];    tao(r2)=[];
V_p(r2)=[];
Vol_est(r2)=[]; Vol_int(r2)=[];
%costruzione della matrice per la suddivisione delle portate
if m_o_r==0
   MM2=[nodo_prec_r1 n_ing n_out nodo_succ_r3 g_AA g_BB];
end

end
```

COMPATTAZIONE RETE E SIMULAZIONI IN STAZIONARIO DELLE RETI EQUIVALENTI

```
clear all
close all
clc
load A_modificata;
load DatiPerNicolaBCT408;
load t_Gmax;
load T_m_nodale;
load T_r_nodale;
inlet_node=NetData.BCT;
Gext=BC.b(t_Gmax,:); Gext(inlet_node)=-sum(Gext) ;  carico=BC.Phi(t_Gmax,:);
% portate esterne e carichi definiti ai nodi
L=NetData.L; d_int=NetData.D;
h=1.2; %[W/m/K]
H=h*L;
X=NetData.X; Y=NetData.Y;
%tolgo i rami con portate nulle ed i nodi uscenti da questi rami
load G_staz; flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[];       Gext(nodo_out)=[];       carico(nodo_out)=[];
X(nodo_out)=[]; Y(nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[];     L(ramo)=[];     d_int(ramo)=[];     G_staz(ramo)=[];
H(ramo)=[];
   else
       flag=1;
   end
end
% Verifico la rete con il disegno
n_nodi=size(A,1); n_rami=size(A,2);
figure(1)
hold on
grid on
box on
for i=1:n_nodi
    rami_in=find(A(i,:)==-1);  %vettore  contenente  gli  indici  dei  rami  in
ingresso al nodo i
    n_in=length(rami_in); % numero di rami in ingresso
    for j=1:n_in;
       w=rami_in(j);
       nodo_prec=find(A(:,w)==1);
       plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'k');
    end
end
xlabel ('coordinata X [m]'); ylabel ('coordinata Y [m]');
title ('RETE ORIGINALE DOPO ELIMINAZIONE RAMI');

%||||| DATI GENERALI ||||||||
TG=10+273.15;
cp_w=4186; cp_p=502;%[J/kg/K]
rho_p=7750; %[Kg/m3]
TBC_in=120+273.15;
```

```
%||||| DATI MANDATA |||||||||
A_m=A; n_nodi=size(A,1); n_rami=size(A,2); clear A;
rho_w_m=951.2; %[Kg/m3]
d_int_m=d_int; sez_int_m=0.25*pi*d_int_m.^2;
d_est_m=1.05*d_int_m; sez_est_m=0.25*pi*d_est_m.^2;
L_m=L;
H_m=H;
phi_m=cp_w*rho_w_m*sez_int_m./(cp_w*rho_w_m*sez_int_m+cp_p*rho_p*(sez_est_m-
sez_int_m));
Gext_m=Gext';
Vol_int_m=sez_int_m.*L_m;    Vol_est_m=sez_est_m.*L_m;    Vol_p_m=Vol_est_m-
Vol_int_m;
G_m=A_m\(-Gext_m);
tao_m=rho_w_m*Vol_int_m./(G_m.*phi_m); %[secondi]

%||||||||| DATI RITORNO ||||||||||||
%definisco A e Gext come quella di mandata e poi cambio il segno dopo la
deramificazione
A_r=A_m; %cambio il segno dopo la dermificazione
rho_w_r=983.3; %[Kg/m3]
d_int_r=d_int_m; sez_int_r=sez_int_m;
d_est_r=d_est_m; sez_est_r=sez_est_m;
L_r=L_m;
phi_r=cp_w*rho_w_r*sez_int_r./(cp_w*rho_w_r*sez_int_r+cp_p*rho_p*(sez_est_r-
sez_int_r));
H_r=H_m;
Gext_r=Gext_m; %cambio il segno dopo la dermificazione
G_r=A_r\(-Gext_r);
Vol_int_r=Vol_int_m; Vol_est_r=Vol_est_m; Vol_p_r=Vol_p_m;
tao_r=rho_w_r*Vol_int_r./(G_r.*phi_r); %[secondi]

%|||||||||||||| INIZIALIZZAZIONE DATI DA PASSARE AL TRANSITORIO ||||||||||||
rami_fine=2; %numero di rami finali che vogliamo ottenere
ML=zeros(n_rami-rami_fine+1,n_rami); %matrice delle lunghezze ai vari stadi
di compattazione (righe)
Md_int_m=ML;  Md_int_r=ML;%matrice  dei  diametri  interni  ai  vari  stadi  di
compattazione (righe)
MH_m=ML; MH_r=ML;%matrice dei coeff H ai vari stadi di compattazione (righe)
A_m_3d=zeros(n_nodi,n_rami,n_rami-rami_fine+1);    %procedo    sull'ase    z
all'aumentare del grado di compattazione
VG_f=zeros(1,n_rami-rami_fine+1);  %  vettore  dei  valori  della  portata
fittizia


%|||||||||||| RIMOZIONE RAMIFICAZIONI ( PER MANDATA E RITORNO ) ||||||||||||
%trovo un nodo con la divergenza
div_node=1;
c_dr=0; % contatore deramificazioni
while div_node~=0
    div_node=0;
    for i=1:n_nodi
        if length(find(A_m(i,:)==1))>=2;
            div_node=i;
        end
    end
    if div_node~=0
       c_dr=c_dr+1;
       %mandata
       %introduciamo m_o_r che vale 0 per la mandata e 1 per il ritorno
```

```
       m_o_r=0; %mandata

[A_m,L_m,d_int_m,d_est_m,tao_m,phi_m,G_m,H_m,sez_int_m,sez_est_m,Vol_int_m,V
ol_est_m,Vol_p_m]=rimozione_ramificazioni(A_m,L_m,d_int_m,d_est_m,tao_m,sez_
int_m,sez_est_m,Vol_int_m,Vol_est_m,Vol_p_m,phi_m,G_m,rho_w_m,H_m,div_node,T
_m,m_o_r);
       A_m_der_3d(1:1:n_nodi,1:1:n_rami,c_dr)=A_m;
       ML_m_der(c_dr,1:1:n_rami)=L_m;
       Md_int_m_der(c_dr,1:1:n_rami)=d_int_m;
       MH_m_der(c_dr,1:1:n_rami)=H_m;
       %ritorno
       m_o_r=1; %ritorno

[A_r,L_r,d_int_r,d_est_r,tao_r,phi_r,G_r,H_r,sez_int_r,sez_est_r,Vol_int_r,V
ol_est_r,Vol_p_r]=rimozione_ramificazioni(A_r,L_r,d_int_r,d_est_r,tao_r,sez_
int_r,sez_est_r,Vol_int_r,Vol_est_r,Vol_p_r,phi_r,G_r,rho_w_r,H_r,div_node,T
_r,m_o_r);
       ML_r_der(c_dr,1:1:n_rami)=L_r;
       Md_int_r_der(c_dr,1:1:n_rami)=d_int_r;
       MH_r_der(c_dr,1:1:n_rami)=H_r;
       z=0; w=0;
       for i=1:n_nodi
           if isempty(find(A_m(i,:)==1))==1
              %allora il nodo i non ha rami uscenti
              z=z+1;
              outlet_node_m(z)=i;
           elseif isempty(find(A_m(i,:)==-1))==1
              w=w+1;
              inlet_node_m(w)=i;
           end
       end

        %|||||||| RISOLUZIONE PROBLEMA TERMICO MANDATA ||||||||||||
        %costruzione di K e di f
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for i=1:n_nodi
            ramo_in=find(A_m(i,:)==-1); %ramo in ingreso al nodo
            rami_out=find(A_m(i,:)==1); %rami in uscita dal nodo
            rami_coll=[ramo_in rami_out]; %rami collegati al nodo
            %calcolo la portata uscente dal nodo (non al contorno)
            Gsum_out=0;
            for j=1:length(rami_out)
                Gsum_out=Gsum_out+G_m(rami_out(j));
            end
            %calcolo la sommatoria dei coeff di scambio termico dei rami
collegati al nodo i
            Hsum=0;
            for j=1:length(rami_coll)
                Hsum=Hsum+H_m(rami_coll(j));
            end
            %assegno i valori agli elementi di K e f
            k(i,i)=cp_w*(Gsum_out+Gext_m(i))+0.5*Hsum;
            if isempty(ramo_in)==0
              nodo_prec=find(A_m(:,ramo_in)==1); %nodo precedente al nodo i
              k(i,nodo_prec)=-cp_w*G_m(ramo_in);
            end
            f(i)=0.5*Hsum*TG;
        end
        %imposizione della condizione al contorno al nodo di ingresso
```

```
        k(inlet_node_m,:)=zeros(1,n_nodi); k(inlet_node_m,inlet_node_m)=1;
        f(inlet_node_m)=TBC_in; % temperatura in ingresso nel baricentro [K]
        %imposizione delle condizioni al contorno ai nodi in uscita
        for i=1:length(outlet_node_m)
            j=outlet_node_m(i); % chiamo con j il nodo di uscita in esame
            ramo_in=find(A_m(j,:)==-1); %ramo in ngreso al nodo j
            nodo_prec=find(A_m(:,ramo_in)==1); %nodo precedente al nodo j
            k(j,j)=-cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
            k(j,nodo_prec)=cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
            f(j)=-H_m(ramo_in)*TG;
        end

        T_m=k\f;

        %calcolo le dispersioni sulla rete di mandata
        disp_dr_m(c_dr)=0;
        for i=1:n_rami
            ramo=i;
            nodi_coll=find(A_m(:,ramo));

disp_dr_m(c_dr)=disp_dr_m(c_dr)+abs(cp_w*G_m(ramo)*(T_m(nodi_coll(1))-
T_m(nodi_coll(2))));
        end


        %preparo la risoluzione del problema termico al ritorno
        inlet_node_r=outlet_node_m;
        outlet_node_r=inlet_node_m;
        A_r=-A_r; Gext_r=-Gext_r;
        Tut_out=zeros(n_nodi,1);
        for i=1:n_nodi
            if Gext_m(i)~=0
               Tut_out(i)=T_m(i)-carico(i)/Gext_m(i)/cp_w;
            end
        end
        %|||||||| RISOLUZIONE PROBLEMA TERMICO RITORNO ||||||||||||
        %costruzione di K e di f
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for i=1:n_nodi
            rami_in=find(A_r(i,:)==-1); %rami in ingreso al nodo i
            ramo_out=find(A_r(i,:)==1); %ramo in uscita al nodo i
            rami_coll=[rami_in ramo_out]; %rami collegati al nodo i
            %calcolo la sommatoria dei coeff di scambio termico dei rami
collegati al nodo i
            Hsum=0;
            for j=1:length(rami_coll)
                Hsum=Hsum+H_r(rami_coll(j));
            end
            %assegno i valori non nulli agli elementi di k e di f
            if isempty(ramo_out)==0
               k(i,i)=cp_w*G_r(ramo_out)+0.5*Hsum;
            end
            for j=1:length(rami_in)
                nodo_prec=find(A_r(:,rami_in(j))==1);
                k(i,nodo_prec)=-cp_w*G_r(rami_in(j));
            end
            f(i)=0.5*Hsum*TG-cp_w*Gext_r(i)*Tut_out(i);
        end
```

```
        %imposizione della condizione al contorno ai nodi di ingresso
        for i=1:length(inlet_node_r)
            j=inlet_node_r(i);
            k(j,:)=zeros(1,n_nodi); k(j,j)=1; f(j)=Tut_out(j);
        end

        %imposizione della condizione al contorno al nodo di uscita
        ramo_in=find(A_r(outlet_node_r,:)==-1);  %ramo  in  ingreso  al  nodo
outlet_node_r
        nodo_prec=find(A_r(:,ramo_in)==1);   %nodo   precedente   al   nodo
outlet_node_r
        k(outlet_node_r,outlet_node_r)=-cp_w*G_m(ramo_in)-H_r(ramo_in)/2;
        k(outlet_node_r,nodo_prec)=cp_w*G_m(ramo_in)-H_r(ramo_in)/2;
        f(outlet_node_r)=-H_r(ramo_in)*TG;

        %trovo T_r
        T_r=k\f;
        %preparo le variabili per la nuova compattazione al ritorno
        A_r=-A_r; Gext_r=-Gext_r;
        %calcolo la potenza primaria
        Q_dr(c_dr)=cp_w*abs(Gext_m(inlet_node_m))*(T_m(inlet_node_m)-
T_r(outlet_node_r));
        disp_dr(c_dr)=Q_dr(c_dr)-sum(carico);    %dispersioni    nel    caso
stazionario
        %evito  errori  per  definizione  di  variabili  con  lunghezza  non
specificata
        clear outlet_node_m inlet_node_m
    end
end

ML(1,:)=L_m;
Md_int_m(1,:)=d_int_m; Md_int_r(1,:)=d_int_r;
MH_m(1,:)=H_m; MH_r(1,:)=H_r;
A_m_3d(:,:,1)=A_m;
VG_f(1)=0;

%|||||||| RIMOZIONE RAMI CHE NON SERVONO UTENZE ||||||||||||
c_rim1=0; %contatore degli step per la rimozione dei rami senza utenze
G_f=0; %portata fittizia
flag=0; %usato per il G_f
while isempty(find((Gext_m)==0))==0
    c_rim1=c_rim1+1;
    %mandata
    m_o_r=0;

[A_m,L_m,d_int_m,d_est_m,tao_m,phi_m,G_m,H_m,sez_int_m,sez_est_m,Vol_int_m,V
ol_est_m,Vol_p_m,Gext_m,Tut_out,G_f,MM1]=rimozione1(A_m,L_m,d_int_m,d_est_m,
tao_m,sez_int_m,sez_est_m,Vol_int_m,Vol_est_m,Vol_p_m,phi_m,G_m,Gext_m,rho_w
_m,H_m,Tut_out,G_f,m_o_r);
    n_nodi=size(A_m,1); n_rami=size(A_m,2);
    if c_rim1==1
       MM=MM1;
    else
       MM=[MM;MM1];
    end
    if G_f<0.0001
       G_f=0;
    end
    ML(c_rim1+1,1:1:length(L_m))=L_m;
```

```
    Md_int_m(c_rim1+1,1:1:length(d_int_m))=d_int_m;
    MH_m(c_rim1+1,1:1:length(H_m))=H_m;
    A_m_3d(1:1:n_nodi,1:1:n_rami,c_rim1+1)=A_m;
    VG_f(c_rim1+1)=G_f;

    for i=1:n_nodi
       if (Gext_m(i)<0.0001) && (Gext_m(i)>-0.0001)
          Gext_m(i)=0;
       end
    end
    %|||||||| RISOLUZIONE PROBLEMA TERMICO MANDATA ||||||||||||
    %trovo inlet ed outlet node per la mandata
    for i=1:n_nodi
        if isempty(find(A_m(i,:)==1))
           outlet_node_m=i;
        end
        if isempty(find(A_m(i,:)==-1))
            inlet_node_m=i;
        end
    end
    %costruisco k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for i=1:n_nodi
       ramo_in=find(A_m(i,:)==-1);
       ramo_out=find(A_m(i,:)==1);
       nodo_prec=find(A_m(:,ramo_in)==1);
       if (i~=inlet_node_m) && (i~=outlet_node_m)

k(i,i)=cp_w*(G_m(ramo_out)+Gext_m(i))+0.5*(H_m(ramo_in)+H_m(ramo_out));
           k(i,nodo_prec)=-cp_w*G_m(ramo_in);
           f(i)=0.5*(H_m(ramo_in)+H_m(ramo_out))*TG;
       end
    end
    % condizioni al nodo di ingresso
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %condizioni al nodo di uscita
    ramo_in=find(A_m(outlet_node_m,:)==-1);
    nodo_prec=find(A_m(:,ramo_in)==1);
    k(outlet_node_m,outlet_node_m)=-cp_w*G_m(ramo_in)-0.5*H_m(ramo_in);
    k(outlet_node_m,nodo_prec)=cp_w*G_m(ramo_in)-0.5*H_m(ramo_in);
    f(outlet_node_m)=-H_m(ramo_in)*TG;
    %soluzione
    T_m=k\f;

    %calcolo le dispersioni sulla rete di mandata
    disp_rim1_m(c_rim1)=0;
    for i=1:n_rami
        ramo=i;
        nodi_coll=find(A_m(:,ramo));

disp_rim1_m(c_rim1)=disp_rim1_m(c_rim1)+abs(cp_w*G_m(ramo)*(T_m(nodi_coll(1)
)-T_m(nodi_coll(2))));
    end

    %ritorno
    m_o_r=1;

[A_r,L_r,d_int_r,d_est_r,tao_r,phi_r,G_r,H_r,sez_int_r,sez_est_r,Vol_int_r,V
ol_est_r,Vol_p_r,Gext_r]=rimozione1(A_r,L_r,d_int_r,d_est_r,tao_r,sez_int_r,
```

```
sez_est_r,Vol_int_r,Vol_est_r,Vol_p_r,phi_r,G_r,Gext_r,rho_w_r,H_r,Tut_out,G
_f,m_o_r);
    for i=1:n_nodi
       if (Gext_r(i)<0.0001) & (Gext_r(i)>-0.0001)
          Gext_r(i)=0;
       end
    end
    Md_int_r(c_rim1+1,1:1:length(d_int_r))=d_int_r;
    MH_r(c_rim1+1,1:1:length(H_r))=H_r;
    %preparo la risoluzione del problema termico al ritorno
    inlet_node_r=outlet_node_m;
    outlet_node_r=inlet_node_m;
    A_r=-A_r; Gext_r=-Gext_r;

    %|||||||| RISOLUZIONE PROBLEMA TERMICO RITORNO ||||||||||||
    %costruisco k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for i=1:n_nodi
        ramo_in=find(A_r(i,:)==-1);
        ramo_out=find(A_r(i,:)==1);
        nodo_prec=find(A_r(:,ramo_in)==1);
        if (i~=inlet_node_r) && (i~=outlet_node_r)
            k(i,i)=cp_w*G_r(ramo_out)+0.5*(H_r(ramo_in)+H_r(ramo_out));
            k(i,nodo_prec)=-cp_w*G_r(ramo_in);
            f(i)=0.5*(H_r(ramo_in)+H_r(ramo_out))*TG-
cp_w*Gext_r(i)*Tut_out(i);
        end
    end
    %condizioni al contorno del nodo di ingresso
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %condizioni al contorno del nodo di uscita
    ramo_in=find(A_r(outlet_node_r,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    k(outlet_node_r,outlet_node_r)=-cp_w*G_r(ramo_in)-0.5*H_r(ramo_in);
    k(outlet_node_r,nodo_prec)=cp_w*G_r(ramo_in)-0.5*H_r(ramo_in);
    f(outlet_node_r)=-H_r(ramo_in)*TG;
    %soluzione problema termico ritorno
    T_r=k\f;
    ultimo_ramo_r=find(A_r(outlet_node_r,:)==-1);

T_r(outlet_node_r)=(G_r(ultimo_ramo_r)*T_r(outlet_node_r)+G_f*Tut_out(outlet
_node_r))/(G_r(ultimo_ramo_r)+G_f);
    %preparo le variabili per la nuova compattazione al ritorno
    A_r=-A_r; Gext_r=-Gext_r;
    %calcolo la potenza primaria
    primo_ramo=find(A_m(inlet_node_m,:)==1);
    Q_rim1(c_rim1)=cp_w*(G_m(primo_ramo)+G_f)*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    disp_rim1(c_rim1)=Q_rim1(c_rim1)-sum(carico);   %dispersioni   nel   caso
stazionario

end


%|||||||| RIMOZIONE RAMI IN BASE ALLA LUNGHEZZA ||||||||||||
rami_fine=2; c_rim2=0;
while length(G_m)>rami_fine
    c_rim2=c_rim2+1;
    %mandata
```

```
    m_o_r=0;

[A_m,L_m,d_int_m,d_est_m,tao_m,phi_m,G_m,H_m,sez_int_m,sez_est_m,Vol_int_m,V
ol_est_m,Vol_p_m,Gext_m,MM2,G_f,Tut_out]=rimozione2(A_m,L_m,d_int_m,d_est_m,
tao_m,sez_int_m,sez_est_m,Vol_int_m,Vol_est_m,Vol_p_m,phi_m,G_m,Gext_m,rho_w
_m,H_m,Tut_out,G_f,m_o_r);
    n_nodi=size(A_m,1); n_rami=size(A_m,2);
    if G_f<0.0001
       G_f=0;
    end
    MM=[MM;MM2];
    ML(c_rim1+c_rim2+1,1:1:length(L_m))=L_m;
    Md_int_m(c_rim1+c_rim2+1,1:1:length(d_int_m))=d_int_m;
    MH_m(c_rim1+c_rim2+1,1:1:length(H_m))=H_m;
    A_m_3d(1:1:n_nodi,1:1:n_rami,c_rim1+c_rim2+1)=A_m;
    VG_f(c_rim1+c_rim2+1)=G_f;
    %|||||||| RISOLUZIONE PROBLEMA TERMICO MANDATA ||||||||||||
    %trovo inlet ed outlet node per la mandata
    for i=1:n_nodi
        if isempty(find(A_m(i,:)==1))
           outlet_node_m=i;
        end
        if isempty(find(A_m(i,:)==-1))
            inlet_node_m=i;
        end
    end
    %costruisco k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for i=1:n_nodi
       ramo_in=find(A_m(i,:)==-1);
       ramo_out=find(A_m(i,:)==1);
       nodo_prec=find(A_m(:,ramo_in)==1);
       if (i~=inlet_node_m) && (i~=outlet_node_m)

k(i,i)=cp_w*(G_m(ramo_out)+Gext_m(i))+0.5*(H_m(ramo_in)+H_m(ramo_out));
           k(i,nodo_prec)=-cp_w*G_m(ramo_in);
           f(i)=0.5*(H_m(ramo_in)+H_m(ramo_out))*TG;
       end
    end
    % condizioni al nodo di ingresso
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %condizioni al nodo di uscita
    ramo_in=find(A_m(outlet_node_m,:)==-1);
    nodo_prec=find(A_m(:,ramo_in)==1);
    k(outlet_node_m,outlet_node_m)=-cp_w*G_m(ramo_in)-0.5*H_m(ramo_in);
    k(outlet_node_m,nodo_prec)=cp_w*G_m(ramo_in)-0.5*H_m(ramo_in);
    f(outlet_node_m)=-H_m(ramo_in)*TG;
    %soluzione
    T_m=k\f;

    %calcolo le dispersioni sulla rete di mandata
    disp_rim2_m(c_rim2)=0;
    for i=1:n_rami
        ramo=i;
        nodi_coll=find(A_m(:,ramo));

disp_rim2_m(c_rim2)=disp_rim2_m(c_rim2)+abs(cp_w*G_m(ramo)*(T_m(nodi_coll(1)
)-T_m(nodi_coll(2))));
    end
```

```
    %ritorno
    m_o_r=1;

[A_r,L_r,d_int_r,d_est_r,tao_r,phi_r,G_r,H_r,sez_int_r,sez_est_r,Vol_int_r,V
ol_est_r,Vol_p_r,Gext_r]=rimozione2(A_r,L_r,d_int_r,d_est_r,tao_r,sez_int_r,
sez_est_r,Vol_int_r,Vol_est_r,Vol_p_r,phi_r,G_r,Gext_r,rho_w_r,H_r,Tut_out,G
_f,m_o_r);
    Md_int_r(c_rim1+1+c_rim2,1:1:length(d_int_r))=d_int_r;
    MH_r(c_rim1+c_rim2+1,1:1:length(H_r))=H_r;
    %preparo la risoluzione del problema termico al ritorno
    inlet_node_r=outlet_node_m;
    outlet_node_r=inlet_node_m;
    A_r=-A_r; Gext_r=-Gext_r;

    %|||||||| RISOLUZIONE PROBLEMA TERMICO RITORNO ||||||||||||
    %costruisco k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for i=1:n_nodi
        ramo_in=find(A_r(i,:)==-1);
        ramo_out=find(A_r(i,:)==1);
        nodo_prec=find(A_r(:,ramo_in)==1);
        if (i~=inlet_node_r) && (i~=outlet_node_r)
            k(i,i)=cp_w*G_r(ramo_out)+0.5*(H_r(ramo_in)+H_r(ramo_out));
            k(i,nodo_prec)=-cp_w*G_r(ramo_in);
            f(i)=0.5*(H_r(ramo_in)+H_r(ramo_out))*TG-
cp_w*Gext_r(i)*Tut_out(i);
        end
    end
    %condizioni al contorno del nodo di ingresso
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %condizioni al contorno del nodo di uscita
    ramo_in=find(A_r(outlet_node_r,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    k(outlet_node_r,outlet_node_r)=-cp_w*G_r(ramo_in)-0.5*H_r(ramo_in);
    k(outlet_node_r,nodo_prec)=cp_w*G_r(ramo_in)-0.5*H_r(ramo_in);
    f(outlet_node_r)=-H_r(ramo_in)*TG;
    %soluzione problema termico ritorno
    T_r=k\f;
    ultimo_ramo_r=find(A_r(outlet_node_r,:)==-1);

T_r(outlet_node_r)=(G_r(ultimo_ramo_r)*T_r(outlet_node_r)+G_f*Tut_out(outlet
_node_r))/(G_r(ultimo_ramo_r)+G_f);
    %preparo le variabili per la nuova compattazione al ritorno
    A_r=-A_r; Gext_r=-Gext_r;
    %calcolo la potenza primaria
    primo_ramo=find(A_m(inlet_node_m,:)==1);
    Q_rim2(c_rim2)=cp_w*(G_m(primo_ramo)+G_f)*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    disp_rim2(c_rim2)=Q_rim2(c_rim2)-sum(carico);   %dispersioni   nel   caso
stazionario
end

Q=[Q_dr Q_rim1 Q_rim2];
disp=[disp_dr    disp_rim1    disp_rim2];    disp_m=[disp_dr_m    disp_rim1_m
disp_rim2_m];
load disp_staz_nod; load disp_staz_nod_m; load Q_staz_nod;
err_Q=abs(Q-Q_staz)./Q_staz*100;
err_disp=abs(disp-disp_staz)./disp_staz*100;
```

```
err_disp_m=abs(disp_m-disp_staz_m)./disp_staz_m*100;
%step_tot=[1:1:c_dr+c_rim1+c_rim2]; ns=length(step_tot);

figure(2)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, Q_staz*ones(1,c_dr+c_rim1+c_rim2) ,'r');
plot(1:1:c_dr+c_rim1+c_rim2,Q,'b');
legend('Q nodale','Q compatta','location','neo');
plot([c_dr c_dr],[min(Q) max(Q)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(Q) max(Q)],'k--');
xlabel('step compattazione'); ylabel('[W]');


figure(3)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2,disp_staz*ones(1,c_dr+c_rim1+c_rim2) ,'r');
plot(1:1:c_dr+c_rim1+c_rim2,disp,'b');
legend('Rete originale',' Rete equivalente','location','neo');
title ('Dispersioni attraverso la rete');
plot([c_dr c_dr],[min(disp) max(disp)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(disp) max(disp)],'k--');
xlabel('step compattazione'); ylabel('[W]');


figure(4)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, disp_staz_m*ones(1,c_dr+c_rim1+c_rim2) ,'r');
plot(1:1:c_dr+c_rim1+c_rim2,disp_m,'b');
leg=legend('dispersioni mandata modello nodale','dispersioni mandata dopo la
compattazione','location','neo');
plot([c_dr c_dr],[min(disp_m) max(disp_m)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(disp_m) max(disp_m)],'k--');
xlabel('step compattazione'); ylabel('[W]');

figure(5)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, Q_staz*ones(1,c_dr+c_rim1+c_rim2) ,'r');
plot(1:1:c_dr+c_rim1+c_rim2,Q,'b');
legend('Q rete originale','Q rete semplificata','location','neo');
plot([c_dr c_dr],[min(Q) max(Q)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(Q) max(Q)],'k--');
xlabel('step compattazione'); ylabel('[W]');

figure(6)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, err_Q ,'b');
plot([c_dr c_dr],[min(err_Q) max(err_Q)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(err_Q) max(err_Q)],'k--');
xlabel('step compattazione'); ylabel('%');
```

```
title('Errore potenza da fornire alla rete');

figure(7)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, err_disp ,'b');
plot([c_dr c_dr],[min(err_disp) max(err_disp)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(err_disp) max(err_disp)],'k--');
xlabel('step compattazione'); ylabel('%');
title('Errore dispersioni rete semplificata');

figure(8)
hold on
grid on
box on
plot(1:1:c_dr+c_rim1+c_rim2, err_disp_m ,'b');
plot([c_dr c_dr],[min(err_disp_m) max(err_disp_m)],'k--');
plot([c_dr+c_rim1 c_dr+c_rim1],[min(err_disp_m) max(err_disp_m)],'k--');
xlabel('step compattazione'); ylabel('%');
title('errore percentuale dispersione mandata rete');

save('MM','MM');
save('ML','ML');
save('Md_int_m','Md_int_m');save('Md_int_r','Md_int_r')
save('MH_m','MH_m'); save('MH_r','MH_r');
save('A_m_3d','A_m_3d');

save('A_m_der_3d','A_m_der_3d');
save('ML_m_der','ML_m_der');
save('Md_int_m_der','Md_int_m_der');
save('MH_m_der','MH_m_der');

save('ML_r_der','ML_r_der');
save('Md_int_r_der','Md_int_r_der');
save('MH_r_der','MH_r_der');
```

SIMULAZIONE IN TRANSITORIO 1 SULLA RETE ORIGINALE

```
clear all
close all
clc

load DatiPerNicolaBCT408;
carichi=BC.Phi;
MGext=BC.b;
load A_modificata;  A_m=A; A_r=-A_m; n_nodi=size(A_m,1); n_rami=size(A_m,2);
d_int=NetData.D;
L=NetData.L; L_m=L; L_r=L_m;
TBC_in=273.15+120; TG=273.15+10;
dt=24*60*60/size(MGext,1); tt=[dt:dt:dt*size(MGext,1)]; nt=length(tt);
cp_w=4186;
h=1.2; H=h.*L; H_m=H; H_r=H;
rho_w_m=951.2; rho_w_r=983.3;
clear H L A;

%individuo il primo e l'ultimo istante di tempo in cui le utenze richiedono
calore
t_iniz=19800;
t_fin=0;
for i=1:nt
   if isempty(find(MGext(i,:)))==0 && tt(i)>t_fin
      t_fin=tt(i);
   end
end
tempo=t_iniz:dt:t_fin;
%individuo i nodi di ingresso e di uscita dalla rete per la mandata e per il
ritorno
inlet_node_m=NetData.BCT; z=0;
for i=1:n_nodi
    if isempty(find(A_m(i,:)==1))
       z=z+1;
       outlet_node_m(z)=i;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;
T_prec_m=TG*ones(n_nodi,1); %condizione iniziale mandata
T_prec_r=T_prec_m; %condizione iniziale ritorno
%calcolo Hsum ed SLsum( dipende solo dalla rete che è fissa ) ed è uguale
per la
%mandata ed il ritorno
sez_int=0.25*pi*(d_int).^2;
Hsum=zeros(n_nodi,1); SLsum=Hsum;
for i=1:n_nodi
    rami_coll=find(A_m(i,:));
    for j=1:length(rami_coll)
        Hsum(i)=Hsum(i)+H_m(rami_coll(j));
        SLsum(i)=SLsum(i)+sez_int(rami_coll(j))*L_m(rami_coll(j));
    end
end
Trit_centr_nod=zeros(length(t_iniz/dt:1:t_fin/dt),1);   %inizializzazione   T
ritorno in centrale
carico_tot=Trit_centr_nod; %inizializzazione carico_totale
```

```
disp_nod_trans=Trit_centr_nod; %inizializzazione dispersione rete
ist_spegnimento18=148;
T_m18=zeros(length(ist_spegnimento18:1:t_fin/dt),1);
n_operazioni_orig=0;
for i=t_iniz/dt:1:t_fin/dt
    %prendo i dati all'istante di tempo considerato i
    %i=t_iniz/dt; %da mettere in percentuale quando metto il ciclo for
    carico=carichi(i,:);
    Gext_m=MGext(i,:)'; Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;
    %|||||||||||||||||||| MANDATA ||||||||||||||||||||
    %-risoluzione problema fluidodinamico-
    tic
    G_m=A_m\(-Gext_m);
    n_operazioni_orig=n_operazioni_orig+toc;
    for j=1:length(G_m)
        if G_m(j)<0.0001 && G_m(j)>-0.0001
           G_m(j)=0;
        end
    end
    %-risoluzione problema termico
    tic % il toc relaticosi trova alla riga 114
    %costruzione delle matrici k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for j=1:n_nodi
        %individuo rami in ingresso,rami in uscita e nodi precedenti
        ramo_in=find(A_m(j,:)==-1);       nodo_prec=find(A_m(:,ramo_in)==1);
%siccome non abbiamo loop siamo sicuri di avere un solo ramo in ingresso
        rami_out=find(A_m(j,:)==1);
        if j~=inlet_node_m && isempty(find(outlet_node_m==j))
            %individuo la portata totale uscente
            Gsum_out=0;
            for z=1:length(rami_out)
                Gsum_out=Gsum_out+G_m(rami_out(z));
            end
            k(j,j)=0.5*rho_w_m*cp_w*SLsum(j)/dt+cp_w*Gsum_out+0.5*Hsum(j);
            k(j,nodo_prec)=-cp_w*G_m(ramo_in);
            f(j)=0.5*Hsum(j)*TG+0.5*rho_w_m*cp_w*SLsum(j)/dt*T_prec_m(j);
        end
    end
    %BC al nodo di ingresso alla rete di mandata
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %BC ai nodi di uscita dalla rete di mandata
    for j=1:length(outlet_node_m)
        w=outlet_node_m(j);
        if Gext_m(w)~=0
            ramo_in=find(A_m(w,:)==-1); nodo_prec=find(A_m(:,ramo_in)==1);
            Tmed_prec=(T_prec_m(w)+T_prec_m(nodo_prec))/2;

k(w,w)=0.5*rho_w_m*cp_w*sez_int(ramo_in)*L_m(ramo_in)/dt+cp_w*G_m(ramo_in)+0
.5*H_m(ramo_in);

k(w,nodo_prec)=0.5*rho_w_m*cp_w*sez_int(ramo_in)*L_m(ramo_in)/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(w)=rho_w_m*cp_w*sez_int(ramo_in)*L_m(ramo_in)/dt*Tmed_prec+H_m(ramo_in)*TG
;
        else
            ramo_in=find(A_m(w,:)==-1);
```

```
k(w,w)=rho_w_m*cp_w*sez_int(ramo_in)*L_m(ramo_in)/dt+H_m(ramo_in);

f(w)=H_m(ramo_in)*TG+rho_w_m*cp_w*sez_int(ramo_in)*L_m(ramo_in)/dt*T_prec_m(
w);
        end
    end

    T_m=k\f;
    T_prec_m=T_m;
    n_operazioni_orig=n_operazioni_orig+toc; % tick alla riga 76
    if i>=ist_spegnimento18
       T_m18(i-ist_spegnimento18+1)=T_m(18);
    end

    %trovo Tut_out dal carico
    Tut_out=zeros(n_nodi,1);
    for j=1:length(outlet_node_m)
        w=outlet_node_m(j);  %  chiamo  w  il  nodo  in  uscita  della  mandata
considerato
        if Gext_m(w)~=0
           Tut_out(w)=T_m(w)-carico(w)/Gext_m(w)/cp_w;
        end
    end

    %|||||||||||||||||||| RITORNO ||||||||||||||||||||
    %-risoluzione problema fluidodinamico-
    tic
    G_r=A_r\(-Gext_r);
    n_operazioni_orig=n_operazioni_orig+toc;
    for j=1:length(G_r)
        if G_r(j)<0.0001 && G_r(j)>-0.0001
           G_r(j)=0;
        end
    end
    %-risoluzione problema fluidodinamico-
    tic % il toc relativo si trova alla riga 177
    %costruzione delle matrici k ed f
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for j=1:n_nodi
        if j~=outlet_node_r && isempty(find(inlet_node_r==j))
            rami_in=find(A_r(j,:)==-1);
            ramo_out=find(A_r(j,:)==1);
            k(j,j)=rho_w_r*cp_w*SLsum(j)/2/dt+cp_w*G_r(ramo_out)+Hsum(j)/2;
            for z=1:length(rami_in)
                w=rami_in(z);
                nodo_prec=find(A_r(:,w)==1);
                k(j,nodo_prec)=-cp_w*G_r(w);
            end
            f(j)=Hsum(j)/2*TG+rho_w_r*cp_w*SLsum(j)/2/dt*T_prec_r(j);
        end
    end
    %BC ai nodi di ingresso
    for j=1:length(inlet_node_r)
        w=inlet_node_r(j);
        if  Tut_out(w)~=0
            k(w,w)=1;
            f(w)=Tut_out(w);
        else
```

```
            ramo_out=find(A_r(w,:)==1);

k(w,w)=rho_w_r*cp_w*sez_int(ramo_out)*L_r(ramo_out)/dt+H_r(ramo_out);

f(w)=H_r(ramo_out)*TG+rho_w_r*cp_w*sez_int(ramo_out)*L_r(ramo_out)/dt*T_prec
_r(w);
        end
    end
    %BC al nodo di uscita
    w=outlet_node_r;
    ramo_in=find(A_r(w,:)==-1); nodo_prec=find(A_r(:,ramo_in)==1);
    Tmed_prec=(T_prec_r(w)+T_prec_r(nodo_prec))/2;

k(w,w)=0.5*rho_w_r*cp_w*sez_int(ramo_in)*L_r(ramo_in)/dt+cp_w*G_r(ramo_in)+0
.5*H_r(ramo_in);
    k(w,nodo_prec)=0.5*rho_w_r*cp_w*sez_int(ramo_in)*L_r(ramo_in)/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(w)=rho_w_r*cp_w*sez_int(ramo_in)*L_r(ramo_in)/dt*Tmed_prec+H_r(ramo_in)*TG
;
    %trovo T_r
    T_r=k\f;
    T_prec_r=T_r;
    n_operazioni_orig=n_operazioni_orig+toc; %tic alla riga 139
    Q_nod_trans(i)=abs(Gext_m(inlet_node_m))*cp_w*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    carico_tot(i)=sum(carico);
    disp_nod_trans(i)=Q_nod_trans(i)-carico_tot(i);
    Trit_centr_nod(i)=T_r(outlet_node_r);
end

X=NetData.X; Y=NetData.Y;

figure(1)
hold on
grid on
box on
scatter(X,Y,100,T_m','filled'); colorbar; caxis([min(T_m) max(T_m)]);
for i=1:n_nodi
    rami_in=find(A_m(i,:)==-1); %vettore contenente gli indici dei rami in
ingresso al nodo i
    n_in=length(rami_in); % numero di rami in ingresso
    for j=1:n_in;
       w=rami_in(j);
       nodo_prec=find(A_m(:,w)==1);
       if G_m(w)~=0
          plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'k');
       else
          plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'r');
       end
    end
end

figure(2)
hold on
grid on
box on
scatter(X,Y,100,T_r','filled'); colorbar; caxis([min(T_r) max(T_r)]);
for i=1:n_nodi
```

```
    rami_in=find(A_r(i,:)==-1); %vettore contenente gli indici dei rami in
ingresso al nodo i
    n_in=length(rami_in); % numero di rami in ingresso
    for j=1:n_in;
       w=rami_in(j);
       nodo_prec=find(A_r(:,w)==1);
       if G_r(w)~=0
          plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'k');
       else
          plot([X(nodo_prec) X(i)],[Y(nodo_prec) Y(i)],'r');
       end
    end
end


figure(3)
hold on
grid on
box on
plot(tempo./3600,Q_nod_trans(t_iniz/dt:1:t_fin/dt),'b',tempo./3600,carico_to
t(t_iniz/dt:1:t_fin/dt),'r',tempo./3600,disp_nod_trans(t_iniz/dt:1:t_fin/dt)
,'g');
plot(300*[80 80]./3600,[min(Q_nod_trans) max(Q_nod_trans)],'k--');
title('Risultati rete originale');
set(gca,'xtick',1:1:24);
legend('potenza da fornire','carico totale','dispersioni','location','neo');
xlabel('tempo [h]'); ylabel('[W]');


save('Q_nod_trans','Q_nod_trans');
save('t_iniz','t_iniz');
save('t_fin','t_fin');
save('disp_nod_trans','disp_nod_trans');
save('Trit_centr_nod','Trit_centr_nod');
save('n_operazioni_orig','n_operazioni_orig');

figure(4)
hold on
grid on
box on
plot([ist_spegnimento18:1:t_fin/dt]*300,T_m18,'b');
title('Temperatura del nodo 18 dopo lo spegnimento della relativa utenza ');
xlabel('tempo  [s]'); ylabel('[K]');
```

SIMULAZIONI IN TRANSITORIO 1 SULLE RETI EQUIVALENTI

```
clear all
close all
clc

%carico i dati della rete compattata
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\ML
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_m_3d
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MM
load t_iniz; load t_fin; %[s]
load disp_nod_trans; load Q_nod_trans;
%carico la portata stazionaria in condizioni di carico massimo e la matrice
%raddrizzata (A_modificata) per modificare i dati dei carichi e delle
portate Gext
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_modif
icata
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\G_staz

%carico i dati delle portate esterne e dei carichi e li modifico
load DatiPerNicolaBCT408
MGext_m=BC.b; Mcarichi=BC.Phi;
flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[]; MGext_m(:,nodo_out)=[]; Mcarichi(:,nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[]; G_staz(ramo)=[];
   else
       flag=1;
   end
end
clear A
clear G_staz

%|||||||||||| DATI |||||||||||||||
TBC_in=273.15+120; TG=273.15+10;
cp_w=4186;
rho_w_m=951.2; rho_w_r=983.3;
dt=300; %[s]
%i per i gradi di compattazione
%j per il tempo
```

```
%z per gli altri cicli interni al ciclo del tempo

%|||||||||||||||| RETE DERAMIFICATA |||||||||||||||||||||
A_m=A_m_3d(:,:,1); n_nodi=size(A_m,1); n_rami=size(A_m,2);
A_r=-A_m;
L_m=ML(1,:); L_r=L_m;
H_m=MH_m(1,:); H_r=MH_r(1,:);
d_int_m=Md_int_m(1,:); d_int_r=Md_int_r(1,:);
sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
T_prec_m=TG*ones(n_nodi,1); T_prec_r=T_prec_m; % IC
n_operazioni_deram=0; scost_Q_der=0;

%trovo i nodi di ingresso e di uscita dalla rete
for j=1:n_nodi
    if isempty(find(A_m(j,:)==-1))
        inlet_node_m=j;
    elseif isempty(find(A_m(j,:)==1))
        outlet_node_m=j;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;


%Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete (
quindi compattazione )
%e non dal tempo
Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
SLsum_m=Hsum_m; SLsum_r=Hsum_m;
for j=1:n_nodi
    if j~= inlet_node_m && j~=outlet_node_m
       rami_coll=find(A_m(j,:));
       Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
       Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
       SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
       SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
    end
end
n_operazioni_der=0;
T_rit_centr_der=zeros(length(t_iniz/dt:1:t_fin/dt),1);
Q_solo_der=T_rit_centr_der;
carichi_tempo=T_rit_centr_der;
disp_solo_der=T_rit_centr_der;
for j=t_iniz/dt:1:t_fin/dt
    %leggo i dati delle portate e dei carichi
    Gext_m=MGext_m(j,:)'; Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;

    carichi=Mcarichi(j,:);
    %--risoluzione problema fluidodinamico mandata--
    tic
    G_m=A_m\(-Gext_m);
    n_operazioni_deram=n_operazioni_deram+toc;
    for z=1:n_rami
        if G_m(z)<0.0001 && G_m(z)>-0.0001
            G_m(z)=0;
        end
    end
    %--risoluzione problema termico mandata
```

```
    tic
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_m && z~=outlet_node_m
           ramo_in=find(A_m(z,:)==-1);
           ramo_out=find(A_m(z,:)==1);
           nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
           k(z,nodo_prec)=-cp_w*G_m(ramo_in);
           f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
        end
    end
    %BC inlet node
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %BC outlet_node
    ramo_in=find(A_m(outlet_node_m,:)==-1);
    if Gext_m(outlet_node_m)~=0
        nodo_prec=find(A_m(:,ramo_in)==1);
        T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
        k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
    else

k(outlet_node_m,outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+H_m(ramo_i
n);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_prec_m(outlet_node_m)+
H_m(ramo_in)*TG;
    end
    T_m=k\f;
    T_prec_m=T_m;
    n_operazioni_deram=n_operazioni_deram+toc;
    %cerco la temperatura di uscita dalle utenze
    Tut_out=zeros(n_nodi,1);
    for z=1:n_nodi
        if Gext_m(z)~=0 && z~=inlet_node_m
           Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
        end
    end

    %--risoluzione problema fluidodinamico ritorno
    tic
    G_r=A_r\(-Gext_r);
    n_operazioni_deram=n_operazioni_deram+toc;
    for z=1:n_rami
        if G_r(z)<0.0001 && G_r(z)>-0.0001
            G_r(z)=0;
        end
    end
    %--risoluzione problema termico ritorno--
    tic
```

```
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_r && z~=outlet_node_r
           ramo_in=find(A_r(z,:)==-1);
           ramo_out=find(A_r(z,:)==1);
           nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
           k(z,nodo_prec)=-cp_w*G_r(ramo_in);
           f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
        end
    end
    %BC inlet node
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %BC outlet_node
    ramo_in=find(A_r(outlet_node_r,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
    k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
    %risoluzione
    T_r=k\f;
    T_prec_r=T_r;
    n_operazioni_deram=n_operazioni_deram+toc;
    T_rit_centr_der(j)=T_r(outlet_node_r);
    Q_solo_der(j)=abs(Gext_m(inlet_node_m))*cp_w*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    carichi_tempo(j)=sum(carichi);
    disp_solo_der(j)=Q_solo_der(j)-carichi_tempo(j);
    scost_Q_der=scost_Q_der+abs(Q_solo_der(j)-Q_nod_trans(j));
end

figure(1)
hold on
grid on
box on
plot(dt:dt:t_fin,carichi_tempo,dt:dt:t_fin,Q_solo_der);
plot(dt:dt:t_fin,Q_nod_trans,'k--');
legend('carico','potenza in ingresso nella rete equivalente','potenza in
ingresso nella rete equivalente originale','location','neo');
title('potenze con 154 rami (no ramificazioni)');

figure(2)
hold on
grid on
box on
plot(dt:dt:t_fin,disp_solo_der,dt:dt:t_fin,disp_nod_trans,'k--');
legend('dispersione modello compatto','dispersione modello nodale');
title('dispersioni con 154 rami (no ramificazioni)');

n_rami_in=n_rami;
%|||||||||||||||| RETE COMPATTATA ||||||||||||||||||||
```

```
comp_plot=[1  20  40  60  80  100  120  140  149  150  151  152];  %sono  le
compattazioni che voglio plottare
Q_comp_trans=zeros(length(comp_plot),length(dt:dt:t_fin));
disp_comp_trans=Q_comp_trans;
Trit_centr_comp=Q_comp_trans;
n_operazioni_comp=zeros(length(comp_plot),1);
scost_Q_comp=zeros(length(comp_plot),1);
n_rami_plot_comp=zeros(1,length(comp_plot));
for i=1:length(comp_plot)
    gc=comp_plot(i); %grado di compattazine
    n_rami=n_rami_in-gc
    n_rami_plot_comp(i)=n_rami;
    n_nodi=n_rami+1; %no loop
    %negli indicatori metto gc+1 invece che gc perchè i valori 1 si
    %riferivano al grad di compattazione zero
    A_m=A_m_3d(1:1:n_nodi,1:1:n_rami,gc+1);
    A_r=-A_m;
    L_m=ML(gc+1,1:1:n_rami); L_r=L_m;
    H_m=MH_m(gc+1,1:1:n_rami); H_r=MH_r(gc+1,1:1:n_rami);
    d_int_m=Md_int_m(gc+1,1:1:n_rami); d_int_r=Md_int_r(gc+1,1:1:n_rami);
    sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
    Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
    T_prec_m=TG*ones(n_nodi,1); T_prec_r=T_prec_m; % IC

    %trovo i nodi di ingresso e di uscita dalla rete
    for j=1:n_nodi
        if isempty(find(A_m(j,:)==-1))
            inlet_node_m=j;
        elseif isempty(find(A_m(j,:)==1))
            outlet_node_m=j;
        end
    end
    inlet_node_r=outlet_node_m;
    outlet_node_r=inlet_node_m;

    %Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete
( quindi compattazione )
    %e non dal tempo
    Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
    SLsum_m=Hsum_m; SLsum_r=Hsum_m;
    for j=1:n_nodi
        if j~= inlet_node_m && j~=outlet_node_m
           rami_coll=find(A_m(j,:));
           Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
           Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
           SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
           SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
        end
    end

    for j=t_iniz/dt:1:t_fin/dt
        %leggo i dati delle portate e dei carichi
        Gext_m=MGext_m(j,:)';
        carichi=Mcarichi(j,:);
        %li vario mediante la matrice dei coefficienti MM
        %vario i carichi
        for z=1:gc
            carico_A=carichi(MM(z,1))+MM(z,5)*carichi(MM(z,2));
```

```
            carico_B=(1-MM(z,5))*carichi(MM(z,2))+(1-
MM(z,6))*carichi(MM(z,3));
            carico_D=carichi(MM(z,4))+MM(z,6)*carichi(MM(z,3));
            carichi(MM(z,1))=carico_A;
            carichi(MM(z,2))=carico_B;
            carichi(MM(z,4))=carico_D;
            carichi(MM(z,3))=[];
        end
        %vario le portate uscenti
        for z=1:gc
            portata_A=Gext_m(MM(z,1))+MM(z,5)*Gext_m(MM(z,2));
            portata_B=(1-MM(z,5))*Gext_m(MM(z,2))+(1-
MM(z,6))*Gext_m(MM(z,3));
            portata_D=Gext_m(MM(z,4))+MM(z,6)*Gext_m(MM(z,3));
            Gext_m(MM(z,1))=portata_A;
            Gext_m(MM(z,2))=portata_B;
            Gext_m(MM(z,4))=portata_D;
            Gext_m(MM(z,3))=[];
        end
        G_f=Gext_m(inlet_node_m);
        Gext_m(inlet_node_m)=-sum(Gext_m)+G_f; Gext_r=-Gext_m;

        %--risoluzione problema fluidodinamico mandata--
        tic
        G_m=A_m\(-Gext_m);
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        for z=1:n_rami
            if G_m(z)<0.0001 && G_m(z)>-0.0001
                G_m(z)=0;
            end
        end

        %--risoluzione problema termico mandata
        tic
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_m && z~=outlet_node_m
               ramo_in=find(A_m(z,:)==-1);
               ramo_out=find(A_m(z,:)==1);
               nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
               k(z,nodo_prec)=-cp_w*G_m(ramo_in);

f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
            end
        end
        %BC inlet node
        k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
        %BC outlet_node
        ramo_in=find(A_m(outlet_node_m,:)==-1);
        if Gext_m(outlet_node_m)~=0
            nodo_prec=find(A_m(:,ramo_in)==1);
            T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
```

```
k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
        else

k(outlet_node_m,outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+H_m(ramo_i
n);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_prec_m(outlet_node_m)+
H_m(ramo_in)*TG;
        end
        T_m=k\f;
        T_prec_m=T_m;
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        %cerco la temperatura di uscita dalle utenze
        Tut_out=zeros(n_nodi,1);
        for z=1:n_nodi
            if Gext_m(z)~=0 && z~=inlet_node_m
               Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
            end
            if z==inlet_node_m && G_f~=0
               Tut_out(z)=T_m(z)-carichi(z)/G_f/cp_w;
            end
        end

        %--risoluzione problema fluidodinamico ritorno
        tic
        G_r=A_r\(-Gext_r);
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        for z=1:n_rami
            if G_r(z)<0.0001 && G_r(z)>-0.0001
                G_r(z)=0;
            end
        end

        %--risoluzione problema termico ritorno--
        tic
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_r && z~=outlet_node_r
               ramo_in=find(A_r(z,:)==-1);
               ramo_out=find(A_r(z,:)==1);
               nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
               k(z,nodo_prec)=-cp_w*G_r(ramo_in);

f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
            end
        end
        %BC inlet node
        k(inlet_node_r,inlet_node_r)=1;
f(inlet_node_r)=Tut_out(inlet_node_r);
        %BC outlet_node
        ramo_in=find(A_r(outlet_node_r,:)==-1);
```

```
        nodo_prec=find(A_r(:,ramo_in)==1);
        T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
        k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
        %risoluzione
        T_r=k\f;
        ultimo_ramo_r=find(A_r(outlet_node_r,:)==-1);

T_r(outlet_node_r)=(G_r(ultimo_ramo_r)*T_r(outlet_node_r)+G_f*Tut_out(outlet
_node_r))/(G_r(ultimo_ramo_r)+G_f);
        T_prec_r=T_r;
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;

Q_comp_trans(i,j)=cp_w*(G_f+abs(Gext_m(inlet_node_m)))*(T_m(inlet_node_m)-
T_r(outlet_node_r));
        disp_comp_trans(i,j)=Q_comp_trans(i,j)-sum(carichi);
        Trit_centr_comp(i,j)=T_r(outlet_node_r);
        scost_Q_comp(i)=scost_Q_comp(i)+abs(Q_comp_trans(i,j)-
Q_nod_trans(j));
     end

end

figure(3)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,Q_solo_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(Q_comp_trans,1)
    plot([t_iniz:dt:t_fin]./3600,Q_comp_trans(i,t_iniz/dt:1:t_fin/dt));
end
plot([t_iniz:dt:t_fin]./3600,Q_nod_trans(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[W]');
set(gca,'xtick',[1:1:24]);
title('Potenze da fornire alla rete');

figure(4)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,disp_solo_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(disp_comp_trans,1)
    plot([t_iniz:dt:t_fin]./3600,disp_comp_trans(i,t_iniz/dt:1:t_fin/dt));
end
plot([t_iniz:dt:t_fin]./3600,disp_nod_trans(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[W]');
set(gca,'xtick',[1:1:24]);
```

```
title('Dispersioni delle reti');

load Trit_centr_nod
figure(5)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,T_rit_centr_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(Trit_centr_comp,1)
    plot(t_iniz:dt:t_fin,Trit_centr_comp(i,t_iniz/dt:1:t_fin/dt));
end
plot(t_iniz:dt:t_fin,Trit_centr_nod(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[K]');
title('Temperatura baricentro ( rete ritorno )');

scost_Q_comp_perc=[scost_Q_der; scost_Q_comp]./sum(Q_nod_trans)*100;

n_rami_plot_comp=[154 n_rami_plot_comp];
figure(6)
hold on
grid on
box on
plot(n_rami_plot_comp,scost_Q_comp_perc,'bo-','markersize',3);
xlabel('numero rami'); ylabel('%');
set(gca,'xtick',sort(n_rami_plot_comp),'xticklabel',{'2','','','5','14','34'
,'54','74','94','114','134','','154'});
title('Errore');

n_operazioni_comp=[n_operazioni_deram; n_operazioni_comp];
figure(7)
hold on
grid on
box on
plot(n_rami_plot_comp,n_operazioni_comp,'bo-','markersize',3);
xlabel('numero rami'); ylabel('numero di operazioni');
set(gca,'xtick',sort(n_rami_plot_comp),'xticklabel',{'2','','','5','14','34'
,'54','74','94','114','134','','154'});
title('Costo computazionale');
```

SIMULAZIONE IN TRANSITORIO 2 SULLA RETE ORIGINALE

```
clear all
close all
clc

%carico i dati della rete compattata
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\ML
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_m_3d
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MM
load t_iniz; load t_fin; %[s]
load disp_nod_trans; load Q_nod_trans;
%carico la portata stazionaria in condizioni di carico massimo e la matrice
%raddrizzata (A_modificata) per modificare i dati dei carichi e delle
portate Gext
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_modif
icata
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\G_staz

%carico i dati delle portate esterne e dei carichi e li modifico
load DatiPerNicolaBCT408
MGext_m=BC.b; Mcarichi=BC.Phi;
flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[]; MGext_m(:,nodo_out)=[]; Mcarichi(:,nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[]; G_staz(ramo)=[];
   else
       flag=1;
   end
end
clear A
clear G_staz

%|||||||||||| DATI |||||||||||||||
TBC_in=273.15+120; TG=273.15+10;
cp_w=4186;
rho_w_m=951.2; rho_w_r=983.3;
dt=300; %[s]
%i per i gradi di compattazione
%j per il tempo
```

```
%z per gli altri cicli interni al ciclo del tempo

%|||||||||||||||| RETE DERAMIFICATA |||||||||||||||||||||
A_m=A_m_3d(:,:,1); n_nodi=size(A_m,1); n_rami=size(A_m,2);
A_r=-A_m;
L_m=ML(1,:); L_r=L_m;
H_m=MH_m(1,:); H_r=MH_r(1,:);
d_int_m=Md_int_m(1,:); d_int_r=Md_int_r(1,:);
sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
T_prec_m=TG*ones(n_nodi,1); T_prec_r=T_prec_m; % IC
n_operazioni_deram=0; scost_Q_der=0;

%trovo i nodi di ingresso e di uscita dalla rete
for j=1:n_nodi
    if isempty(find(A_m(j,:)==-1))
        inlet_node_m=j;
    elseif isempty(find(A_m(j,:)==1))
        outlet_node_m=j;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;


%Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete (
quindi compattazione )
%e non dal tempo
Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
SLsum_m=Hsum_m; SLsum_r=Hsum_m;
for j=1:n_nodi
    if j~= inlet_node_m && j~=outlet_node_m
       rami_coll=find(A_m(j,:));
       Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
       Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
       SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
       SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
    end
end
n_operazioni_der=0;
T_rit_centr_der=zeros(length(t_iniz/dt:1:t_fin/dt),1);
Q_solo_der=T_rit_centr_der;
carichi_tempo=T_rit_centr_der;
disp_solo_der=T_rit_centr_der;
for j=t_iniz/dt:1:t_fin/dt
    %leggo i dati delle portate e dei carichi
    Gext_m=MGext_m(j,:)'; Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;

    carichi=Mcarichi(j,:);
    %--risoluzione problema fluidodinamico mandata--
    tic
    G_m=A_m\(-Gext_m);
    n_operazioni_deram=n_operazioni_deram+toc;
    for z=1:n_rami
        if G_m(z)<0.0001 && G_m(z)>-0.0001
            G_m(z)=0;
        end
    end
    %--risoluzione problema termico mandata
```

```
    tic
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_m && z~=outlet_node_m
           ramo_in=find(A_m(z,:)==-1);
           ramo_out=find(A_m(z,:)==1);
           nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
           k(z,nodo_prec)=-cp_w*G_m(ramo_in);
           f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
        end
    end
    %BC inlet node
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %BC outlet_node
    ramo_in=find(A_m(outlet_node_m,:)==-1);
    if Gext_m(outlet_node_m)~=0
        nodo_prec=find(A_m(:,ramo_in)==1);
        T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
        k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
    else

k(outlet_node_m,outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+H_m(ramo_i
n);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_prec_m(outlet_node_m)+
H_m(ramo_in)*TG;
    end
    T_m=k\f;
    T_prec_m=T_m;
    n_operazioni_deram=n_operazioni_deram+toc;
    %cerco la temperatura di uscita dalle utenze
    Tut_out=zeros(n_nodi,1);
    for z=1:n_nodi
        if Gext_m(z)~=0 && z~=inlet_node_m
           Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
        end
    end

    %--risoluzione problema fluidodinamico ritorno
    tic
    G_r=A_r\(-Gext_r);
    n_operazioni_deram=n_operazioni_deram+toc;
    for z=1:n_rami
        if G_r(z)<0.0001 && G_r(z)>-0.0001
            G_r(z)=0;
        end
    end
    %--risoluzione problema termico ritorno--
    tic
```

```
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_r && z~=outlet_node_r
           ramo_in=find(A_r(z,:)==-1);
           ramo_out=find(A_r(z,:)==1);
           nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
           k(z,nodo_prec)=-cp_w*G_r(ramo_in);
           f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
        end
    end
    %BC inlet node
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %BC outlet_node
    ramo_in=find(A_r(outlet_node_r,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
    k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
    %risoluzione
    T_r=k\f;
    T_prec_r=T_r;
    n_operazioni_deram=n_operazioni_deram+toc;
    T_rit_centr_der(j)=T_r(outlet_node_r);
    Q_solo_der(j)=abs(Gext_m(inlet_node_m))*cp_w*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    carichi_tempo(j)=sum(carichi);
    disp_solo_der(j)=Q_solo_der(j)-carichi_tempo(j);
    scost_Q_der=scost_Q_der+abs(Q_solo_der(j)-Q_nod_trans(j));
end

figure(1)
hold on
grid on
box on
plot(dt:dt:t_fin,carichi_tempo,dt:dt:t_fin,Q_solo_der);
plot(dt:dt:t_fin,Q_nod_trans,'k--');
legend('carico','potenza in ingresso nella rete equivalente','potenza in
ingresso nella rete equivalente originale','location','neo');
title('potenze con 154 rami (no ramificazioni)');

figure(2)
hold on
grid on
box on
plot(dt:dt:t_fin,disp_solo_der,dt:dt:t_fin,disp_nod_trans,'k--');
legend('dispersione modello compatto','dispersione modello nodale');
title('dispersioni con 154 rami (no ramificazioni)');

n_rami_in=n_rami;
%|||||||||||||||| RETE COMPATTATA ||||||||||||||||||||
```

```
comp_plot=[1  20  40  60  80  100  120  140  149  150  151  152];  %sono  le
compattazioni che voglio plottare
Q_comp_trans=zeros(length(comp_plot),length(dt:dt:t_fin));
disp_comp_trans=Q_comp_trans;
Trit_centr_comp=Q_comp_trans;
n_operazioni_comp=zeros(length(comp_plot),1);
scost_Q_comp=zeros(length(comp_plot),1);
n_rami_plot_comp=zeros(1,length(comp_plot));
for i=1:length(comp_plot)
    gc=comp_plot(i); %grado di compattazine
    n_rami=n_rami_in-gc
    n_rami_plot_comp(i)=n_rami;
    n_nodi=n_rami+1; %no loop
    %negli indicatori metto gc+1 invece che gc perchè i valori 1 si
    %riferivano al grad di compattazione zero
    A_m=A_m_3d(1:1:n_nodi,1:1:n_rami,gc+1);
    A_r=-A_m;
    L_m=ML(gc+1,1:1:n_rami); L_r=L_m;
    H_m=MH_m(gc+1,1:1:n_rami); H_r=MH_r(gc+1,1:1:n_rami);
    d_int_m=Md_int_m(gc+1,1:1:n_rami); d_int_r=Md_int_r(gc+1,1:1:n_rami);
    sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
    Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
    T_prec_m=TG*ones(n_nodi,1); T_prec_r=T_prec_m; % IC

    %trovo i nodi di ingresso e di uscita dalla rete
    for j=1:n_nodi
        if isempty(find(A_m(j,:)==-1))
            inlet_node_m=j;
        elseif isempty(find(A_m(j,:)==1))
            outlet_node_m=j;
        end
    end
    inlet_node_r=outlet_node_m;
    outlet_node_r=inlet_node_m;

    %Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete
( quindi compattazione )
    %e non dal tempo
    Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
    SLsum_m=Hsum_m; SLsum_r=Hsum_m;
    for j=1:n_nodi
        if j~= inlet_node_m && j~=outlet_node_m
           rami_coll=find(A_m(j,:));
           Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
           Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
           SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
           SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
        end
    end

    for j=t_iniz/dt:1:t_fin/dt
        %leggo i dati delle portate e dei carichi
        Gext_m=MGext_m(j,:)';
        carichi=Mcarichi(j,:);
        %li vario mediante la matrice dei coefficienti MM
        %vario i carichi
        for z=1:gc
            carico_A=carichi(MM(z,1))+MM(z,5)*carichi(MM(z,2));
```

```
            carico_B=(1-MM(z,5))*carichi(MM(z,2))+(1-
MM(z,6))*carichi(MM(z,3));
            carico_D=carichi(MM(z,4))+MM(z,6)*carichi(MM(z,3));
            carichi(MM(z,1))=carico_A;
            carichi(MM(z,2))=carico_B;
            carichi(MM(z,4))=carico_D;
            carichi(MM(z,3))=[];
        end
        %vario le portate uscenti
        for z=1:gc
            portata_A=Gext_m(MM(z,1))+MM(z,5)*Gext_m(MM(z,2));
            portata_B=(1-MM(z,5))*Gext_m(MM(z,2))+(1-
MM(z,6))*Gext_m(MM(z,3));
            portata_D=Gext_m(MM(z,4))+MM(z,6)*Gext_m(MM(z,3));
            Gext_m(MM(z,1))=portata_A;
            Gext_m(MM(z,2))=portata_B;
            Gext_m(MM(z,4))=portata_D;
            Gext_m(MM(z,3))=[];
        end
        G_f=Gext_m(inlet_node_m);
        Gext_m(inlet_node_m)=-sum(Gext_m)+G_f; Gext_r=-Gext_m;

        %--risoluzione problema fluidodinamico mandata--
        tic
        G_m=A_m\(-Gext_m);
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        for z=1:n_rami
            if G_m(z)<0.0001 && G_m(z)>-0.0001
                G_m(z)=0;
            end
        end

        %--risoluzione problema termico mandata
        tic
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_m && z~=outlet_node_m
               ramo_in=find(A_m(z,:)==-1);
               ramo_out=find(A_m(z,:)==1);
               nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
               k(z,nodo_prec)=-cp_w*G_m(ramo_in);

f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
            end
        end
        %BC inlet node
        k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
        %BC outlet_node
        ramo_in=find(A_m(outlet_node_m,:)==-1);
        if Gext_m(outlet_node_m)~=0
            nodo_prec=find(A_m(:,ramo_in)==1);
            T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
```

```
k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
        else

k(outlet_node_m,outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+H_m(ramo_i
n);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_prec_m(outlet_node_m)+
H_m(ramo_in)*TG;
        end
        T_m=k\f;
        T_prec_m=T_m;
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        %cerco la temperatura di uscita dalle utenze
        Tut_out=zeros(n_nodi,1);
        for z=1:n_nodi
            if Gext_m(z)~=0 && z~=inlet_node_m
               Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
            end
            if z==inlet_node_m && G_f~=0
               Tut_out(z)=T_m(z)-carichi(z)/G_f/cp_w;
            end
        end

        %--risoluzione problema fluidodinamico ritorno
        tic
        G_r=A_r\(-Gext_r);
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
        for z=1:n_rami
            if G_r(z)<0.0001 && G_r(z)>-0.0001
                G_r(z)=0;
            end
        end

        %--risoluzione problema termico ritorno--
        tic
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_r && z~=outlet_node_r
               ramo_in=find(A_r(z,:)==-1);
               ramo_out=find(A_r(z,:)==1);
               nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
               k(z,nodo_prec)=-cp_w*G_r(ramo_in);

f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
            end
        end
        %BC inlet node
        k(inlet_node_r,inlet_node_r)=1;
f(inlet_node_r)=Tut_out(inlet_node_r);
        %BC outlet_node
        ramo_in=find(A_r(outlet_node_r,:)==-1);
```

```
        nodo_prec=find(A_r(:,ramo_in)==1);
        T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
        k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
        %risoluzione
        T_r=k\f;
        ultimo_ramo_r=find(A_r(outlet_node_r,:)==-1);

T_r(outlet_node_r)=(G_r(ultimo_ramo_r)*T_r(outlet_node_r)+G_f*Tut_out(outlet
_node_r))/(G_r(ultimo_ramo_r)+G_f);
        T_prec_r=T_r;
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;

Q_comp_trans(i,j)=cp_w*(G_f+abs(Gext_m(inlet_node_m)))*(T_m(inlet_node_m)-
T_r(outlet_node_r));
        disp_comp_trans(i,j)=Q_comp_trans(i,j)-sum(carichi);
        Trit_centr_comp(i,j)=T_r(outlet_node_r);
        scost_Q_comp(i)=scost_Q_comp(i)+abs(Q_comp_trans(i,j)-
Q_nod_trans(j));
     end

end

figure(3)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,Q_solo_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(Q_comp_trans,1)
    plot([t_iniz:dt:t_fin]./3600,Q_comp_trans(i,t_iniz/dt:1:t_fin/dt));
end
plot([t_iniz:dt:t_fin]./3600,Q_nod_trans(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[W]');
set(gca,'xtick',[1:1:24]);
title('Potenze da fornire alla rete');

figure(4)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,disp_solo_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(disp_comp_trans,1)
    plot([t_iniz:dt:t_fin]./3600,disp_comp_trans(i,t_iniz/dt:1:t_fin/dt));
end
plot([t_iniz:dt:t_fin]./3600,disp_nod_trans(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[W]');
set(gca,'xtick',[1:1:24]);
```

```
title('Dispersioni delle reti');

load Trit_centr_nod
figure(5)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,T_rit_centr_der(t_iniz/dt:1:t_fin/dt));
for i=1:size(Trit_centr_comp,1)
    plot(t_iniz:dt:t_fin,Trit_centr_comp(i,t_iniz/dt:1:t_fin/dt));
end
plot(t_iniz:dt:t_fin,Trit_centr_nod(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
orignale','location','neo');
xlabel('tempo [h]'); ylabel('[K]');
title('Temperatura baricentro ( rete ritorno )');

scost_Q_comp_perc=[scost_Q_der; scost_Q_comp]./sum(Q_nod_trans)*100;

n_rami_plot_comp=[154 n_rami_plot_comp];
figure(6)
hold on
grid on
box on
plot(n_rami_plot_comp,scost_Q_comp_perc,'bo-','markersize',3);
xlabel('numero rami'); ylabel('%');
set(gca,'xtick',sort(n_rami_plot_comp),'xticklabel',{'2','','','5','14','34'
,'54','74','94','114','134','','154'});
title('Errore');

n_operazioni_comp=[n_operazioni_deram; n_operazioni_comp];
figure(7)
hold on
grid on
box on
plot(n_rami_plot_comp,n_operazioni_comp,'bo-','markersize',3);
xlabel('numero rami'); ylabel('numero di operazioni');
set(gca,'xtick',sort(n_rami_plot_comp),'xticklabel',{'2','','','5','14','34'
,'54','74','94','114','134','','154'});
title('Costo computazionale');
```

CALCOLO CONDIZIONI INIZIALI PER LA RETE ORIGINALE

```
clear all
close all
clc

load DatiPerNicolaBCT408
load A_modificata
load t_Gmax;
load G_staz

L=NetData.L;
d_int=NetData.D; sez_int=0.25*pi*d_int.^2;
h=1.2; H=h*L;
Gext_m=BC.b(t_Gmax,:)'; %prendo quella corrispondente all'istante in cui ho
carico massimo
carico=BC.Phi(65,:);
X=NetData.X; Y=NetData.Y;
dt=300; % [s]
rho_w_m=951.2; rho_w_r=983.3;
cp_w=4186;
TG=10+273.15;
TBC_in=110+273.15;

%tolgo i rami con portata sempre nulla ed i nodi uscenti da questi rami
flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[];      Gext_m(nodo_out)=[];      carico(nodo_out)=[];
X(nodo_out)=[]; Y(nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[];     L(ramo)=[];     d_int(ramo)=[];     G_staz(ramo)=[];
sez_int(ramo)=[]; H(ramo)=[];
   else
       flag=1;
   end
end
A_m=A; A_r=-A; n_nodi=size(A,1); n_rami=size(A,2); clear A;

%individuo i nodi di uscita e di ingresso della rete
w=0;
for i=1:n_nodi
    if isempty(find(A_m(i,:)==-1))
       inlet_node_m=i;
    elseif isempty(find(A_m(i,:)==1))
        w=w+1;
        outlet_node_m(w)=i;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;
Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;

%modifico i carichi assegnando un valore alle utenze con carico nullo
```

```
for i=1:length(outlet_node_m)
    w=outlet_node_m(i);
    if carico(w)==0
       carico(w)=5e4;
    end
end

%calcolo Hsum ed SLsum ( uguali per mandata e ritorno e non dipendono dal
tempo)
Hsum=zeros(n_nodi,1); SLsum=Hsum;
for i=1:n_nodi
    if i~=inlet_node_m && isempty(find(outlet_node_m==i))
       rami_coll=find(A_m(i,:));
       for j=1:length(rami_coll)
           Hsum(i)=Hsum(i)+H(rami_coll(j));
           SLsum(i)=SLsum(i)+sez_int(j)*L(j);
       end
    end
end

%risoluzione problema fluidodinamico mandata
G_m=A_m\(-Gext_m);
%risoluzione problema fluidodinamico ritorno
G_r=A_r\(-Gext_r);

%risoluzione problema termico mandata
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for i=1:n_nodi
    if i~=inlet_node_m && isempty(find(outlet_node_m==i))
       ramo_in=find(A_m(i,:)==-1); %solo uno perchè no loop
       rami_out=find(A_m(i,:)==1);
       nodo_prec=find(A_m(:,ramo_in)==1); %solo uno perchè un solo ramo_in
       Gsum_out=0;
       for j=1:length(rami_out)
           Gsum_out=Gsum_out+G_m(rami_out(j));
       end
       k(i,i)=cp_w*Gsum_out+0.5*Hsum(i);
       k(i,nodo_prec)=-cp_w*G_m(ramo_in);
       f(i)=0.5*Hsum(i)*TG;
    end
end
%BC nodo ingresso
w=inlet_node_m;
k(w,w)=1; f(w)=TBC_in;
%BC nodi uscita
for i=1:length(outlet_node_m)
    w=outlet_node_m(i);
    ramo_in=find(A_m(w,:)==-1);
    nodo_prec=find(A_m(:,ramo_in)==1);
    k(w,w)=-cp_w*G_m(ramo_in)-H(ramo_in)/2;
    k(w,nodo_prec)=cp_w*G_m(ramo_in)-H(ramo_in)/2;
    f(w)=-H(ramo_in)*TG;
end

T_m=k\f;

%cerco la T di uscita dalle utenze
Tut_out=zeros(n_nodi,n_nodi);
for i=1:length(outlet_node_m)
```

```
    w=outlet_node_m(i);
    Tut_out(w)=T_m(w)-carico(w)/cp_w/Gext_m(w);
end

%risoluzione problema termico ritorno
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for i=1:n_nodi
    if i~=outlet_node_r && isempty(find(inlet_node_r==i))
       rami_in=find(A_r(i,:)==-1);
       ramo_out=find(A_r(i,:)==1); %solo uno cke no loop
       k(i,i)=cp_w*G_m(ramo_out)+0.5*Hsum(i);
       for j=1:length(rami_in);
           ramo_in=rami_in(j);
           nodo_prec=find(A_r(:,ramo_in)==1);
           k(i,nodo_prec)=-cp_w*G_m(ramo_in);
       end
       f(i)=0.5*Hsum(i)*TG;
    end
end
% BC nodi ingresso
for i=1:length(inlet_node_r)
    w=inlet_node_r(i);
    k(w,w)=1; f(w)=Tut_out(w);
end
%BC nodo uscita
w=outlet_node_r;
ramo_in=find(A_r(w,:)==-1);
nodo_prec=find(A_r(:,ramo_in)==1);
k(w,w)=-cp_w*G_m(ramo_in)-0.5*H(ramo_in);
k(w,nodo_prec)=cp_w*G_m(ramo_in)-0.5*H(ramo_in);
f(w)=-H(ramo_in)*TG;

T_r=k\f;

TIC_m=T_m; TIC_r=T_r;
t_iniz=65*300;

save('TIC_m','TIC_m');
save('TIC_r','TIC_r');
save('t_iniz','t_iniz');
```

CALCOLO CONDIZIONI INIZIALI PER LE RETI EQUIVALENTI

```
clear all
close all
clc

load A_modificata;
load DatiPerNicolaBCT408
load G_staz
load t_iniz; %[s]
load t_Gmax; %istante di tempo della portata da considerare

%DATI GENERALI
dt=300; %[s]
cp_w=4186;
TG=10+273.15;
TBC_in=110+273.15;

Gext_m=BC.b(t_Gmax,:)';
carico=BC.Phi(t_iniz/dt,:);
comp_plot=[1  20  40  60  80  100  120  140  149  150  151  152];  %sono  le
compattazioni che voglio plottare

%Tolgo i carichi e le portate esterne dei nodi in cui entrano rami con
portata nulla
flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[]; Gext_m(nodo_out)=[]; carico(nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[]; G_staz(ramo)=[];
   else
       flag=1;
   end
end
clear A; clear G_staz;

%CARICO I DATI DELLE RETI COMPATTATE
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\ML
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_m_3d
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MM

%||||RETE DERAMIFICATA|||||
A_m=A_m_3d(:,:,1); n_nodi=size(A_m,1); n_rami=size(A_m,2);
A_r=-A_m;
L_m=ML(1,:); L_r=L_m;
H_m=MH_m(1,:); H_r=MH_r(1,:);

%trovo i nodi di ingresso e di uscita dalla rete
```

```
for j=1:n_nodi
    if isempty(find(A_m(j,:)==-1))
        inlet_node_m=j;
    elseif isempty(find(A_m(j,:)==1))
        outlet_node_m=j;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;
Gext_m_rif=Gext_m;
Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;
%Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete (
quindi compattazione )e non dal tempo
Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;

for j=1:n_nodi
    if j~= inlet_node_m && j~=outlet_node_m
       rami_coll=find(A_m(j,:));
       Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
       Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
    end
end

% modifico il carico
for j=1:n_nodi
    if Gext_m(j)~=0 && carico(j)==0
       carico(j)=5e4;
    end
end
carico_rif=carico;

%inizializzazioni TIC
MTIC_m_comp=zeros(length(comp_plot)+1,n_nodi);
MTIC_r_comp=MTIC_m_comp;

%--risoluzione problema fluidodinamico mandata--
G_m=A_m\(-Gext_m);
%--risoluzione problema fluidodinamico ritorno
G_r=A_r\(-Gext_r);
%--risoluzione del problema termico mandata--
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for j=1:n_nodi
    if j~=inlet_node_m && j~=outlet_node_m;
       ramo_in=find(A_m(j,:)==-1);
       ramo_out=find(A_m(j,:)==1);
       nodo_prec=find(A_m(:,ramo_in)==1);
       k(j,j)=cp_w*(G_m(ramo_out)+Gext_m(j))+Hsum_m(j)/2;
       k(j,nodo_prec)=-cp_w*G_m(ramo_in);
       f(j)=Hsum_m(j)/2*TG;
    end
end
%BC inlet node
k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
%BC outlet_node
w=outlet_node_m;
ramo_in=find(A_m(w,:)==-1);
nodo_prec=find(A_m(:,ramo_in)==1);
k(w,w)=-cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
k(w,nodo_prec)=cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
```

```
f(w)=-H_m(ramo_in)*TG;

%risoluzione
T_m=k\f;
MTIC_m_comp(1,:)=T_m;

%calcolo la temperature di uscita dalle utenze
Tut_out=zeros(n_nodi,1);
for j=1:n_nodi
    if Gext_m(j)>0
       Tut_out(j)=T_m(j)-carico(j)/cp_w/Gext_m(j);
    end
end
%--risoluzione del problema termico ritorno--
k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
for j=1:n_nodi
    if j~=inlet_node_r && j~=outlet_node_r
       ramo_in=find(A_r(j,:)==-1);
       ramo_out=find(A_r(j,:)==1);
       nodo_prec=find(A_r(:,ramo_in)==1);
       k(j,j)=cp_w*G_r(ramo_out)+0.5*Hsum_r(j);
       k(j,nodo_prec)=-cp_w*G_r(ramo_in);
       f(j)=0.5*Hsum_r(j)*TG-cp_w*Gext_r(j)*Tut_out(j);
    end
end
%BC inlet node
k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
%BC outlet_node
w=outlet_node_r;
ramo_in=find(A_r(w,:)==-1);
nodo_prec=find(A_r(:,ramo_in)==1);
k(w,w)=-cp_w*G_r(ramo_in)-H_r(ramo_in)/2;
k(w,nodo_prec)=cp_w*G_r(ramo_in)-H_r(ramo_in)/2;
f(w)=-H_r(ramo_in)*TG;

%risoluzione
T_r=k\f;
MTIC_r_comp(1,:)=T_r;

%||| MATRICE COMPATTATA ||||
n_rami_iniz=n_rami;
for i=1:length(comp_plot)
    step=comp_plot(i);
    %dati rete
    n_rami=n_rami_iniz-step; n_nodi=n_rami+1;
    A_m=A_m_3d(1:1:n_nodi,1:1:n_rami,step+1);
    A_r=-A_m;
    L_m=ML(step+1,1:1:n_rami); L_r=L_m;
    H_m=MH_m(step+1,1:1:n_rami); H_r=MH_r(step+1,1:1:n_rami);

    %trovo i nodi di ingresso e di uscita dalla rete
    for j=1:n_nodi
        if isempty(find(A_m(j,:)==-1))
            inlet_node_m=j;
        elseif isempty(find(A_m(j,:)==1))
            outlet_node_m=j;
        end
    end
    inlet_node_r=outlet_node_m;
```

```
    outlet_node_r=inlet_node_m;

    %Trovo Hsum per mandata e ritorno
    Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
    for j=1:n_nodi
        if j~= inlet_node_m && j~=outlet_node_m
           rami_coll=find(A_m(j,:));
           Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
           Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
        end
    end
    %settiamo la portata uscente relativa a 155 nodi (senza il valore
negativo al nodo di ingresso)
    Gext_m=Gext_m_rif;
    %vario le portate uscenti mediante i coeff della matrice MM
    for z=1:step
        portata_A=Gext_m(MM(z,1))+MM(z,5)*Gext_m(MM(z,2));
        portata_B=(1-MM(z,5))*Gext_m(MM(z,2))+(1-MM(z,6))*Gext_m(MM(z,3));
        portata_D=Gext_m(MM(z,4))+MM(z,6)*Gext_m(MM(z,3));
        Gext_m(MM(z,1))=portata_A;
        Gext_m(MM(z,2))=portata_B;
        Gext_m(MM(z,4))=portata_D;
        Gext_m(MM(z,3))=[];
    end
    G_f=Gext_m(inlet_node_m);
    Gext_m(inlet_node_m)=-sum(Gext_m)+G_f; Gext_r=-Gext_m;

    %carico di riferimento
    carico=carico_rif;
    %li vario mediante la matrice dei coefficienti MM
    for j=1:step
        carico_A=carico(MM(j,1))+MM(j,5)*carico(MM(j,2));
        carico_B=(1-MM(j,5))*carico(MM(j,2))+(1-MM(j,6))*carico(MM(j,3));
        carico_D=carico(MM(j,4))+MM(j,6)*carico(MM(j,3));
        carico(MM(j,1))=carico_A;
        carico(MM(j,2))=carico_B;
        carico(MM(j,4))=carico_D;
        carico(MM(j,3))=[];
    end
    %--risoluzione problema fluidodinamico mandata--
    G_m=A_m\(-Gext_m);
    %--risoluzione problema fluidodinamico ritorno
    G_r=A_r\(-Gext_r);

    %--risoluzione del problema termico mandata--
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for j=1:n_nodi
        if j~=inlet_node_m && j~=outlet_node_m;
           ramo_in=find(A_m(j,:)==-1);
           ramo_out=find(A_m(j,:)==1);
           nodo_prec=find(A_m(:,ramo_in)==1);
           k(j,j)=cp_w*(G_m(ramo_out)+Gext_m(j))+Hsum_m(j)/2;
           k(j,nodo_prec)=-cp_w*G_m(ramo_in);
           f(j)=Hsum_m(j)/2*TG;
        end
    end
    %BC inlet node
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in;
    %BC outlet_node
```

```
    w=outlet_node_m;
    ramo_in=find(A_m(w,:)==-1);
    nodo_prec=find(A_m(:,ramo_in)==1);
    k(w,w)=-cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
    k(w,nodo_prec)=cp_w*G_m(ramo_in)-H_m(ramo_in)/2;
    f(w)=-H_m(ramo_in)*TG;

    %risoluzione
    T_m=k\f;
    MTIC_m_comp(i+1,1:1:length(T_m))=T_m;

    %calcolo la temperature di uscita dalle utenze
    Tut_out=zeros(n_nodi,1);
    for j=1:n_nodi
        if Gext_m(j)~=0
           Tut_out(j)=T_m(j)-carico(j)/cp_w/Gext_m(j);
        end
    end
    %--risoluzione del problema termico ritorno--
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for j=1:n_nodi
        if j~=inlet_node_r && j~=outlet_node_r
           ramo_in=find(A_r(j,:)==-1);
           ramo_out=find(A_r(j,:)==1);
           nodo_prec=find(A_r(:,ramo_in)==1);
           k(j,j)=cp_w*G_r(ramo_out)+0.5*Hsum_r(j);
           k(j,nodo_prec)=-cp_w*G_r(ramo_in);
           f(j)=0.5*Hsum_r(j)*TG-cp_w*Gext_r(j)*Tut_out(j);
        end
    end
    %BC inlet node
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %BC outlet_node
    w=outlet_node_r;
    ramo_in=find(A_r(w,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    k(w,w)=-cp_w*G_r(ramo_in)-H_r(ramo_in)/2;
    k(w,nodo_prec)=cp_w*G_r(ramo_in)-H_r(ramo_in)/2;
    f(w)=-H_r(ramo_in)*TG;

    %risoluzione
    T_r=k\f;
    %considero G_f

T_r(outlet_node_r)=(G_r(ramo_in)*T_r(outlet_node_r)+G_f*Tut_out(outlet_node_
r))/(G_r(ramo_in)+G_f);
    MTIC_r_comp(i+1,1:1:length(T_r))=T_r;
end

save('MTIC_m_comp','MTIC_m_comp')
save('MTIC_r_comp','MTIC_r_comp')
```

SIMULAZIONE IN TRANSITORIO 2 SULLE RETI EQUIVALENTI

```
clear all
close all
clc

%carico i dati della rete compattata
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\ML
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MH_r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
m
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\Md_int_
r
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_m_3d
load C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\MM
load t_iniz; load t_fin; %[s]
load t_Gmax; %istante in cui ho la portata massima
load MTIC_m_comp; load MTIC_r_comp;
load Q_nod

%carico la portata stazionaria in condizioni di carico massimo e la matrice
%raddrizzata (A_modificata) per modificare i dati dei carichi e delle
portate Gext
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\A_modif
icata
load
C:\Users\Nicola\Desktop\tesi\programmi\BCT408\condizioni_stazionarie\G_staz

%carico i dati delle portate esterne e dei carichi e li modifico
load DatiPerNicolaBCT408
Gext_m=BC.b(t_Gmax,:)'; Mcarichi=BC.Phi;
flag=0;
while flag==0;
   rami_G0=find(G_staz==0);
   if isempty(rami_G0)==0
       ramo=rami_G0(1); %prendo sempre il primo, uno vale l'arltro
       nodo_out=find(A(:,ramo)==-1);
       %tolgo il nodo
       A(nodo_out,:)=[]; Gext_m(nodo_out)=[]; Mcarichi(:,nodo_out)=[];
       %tolgo il ramo
       A(:,ramo)=[]; G_staz(ramo)=[];
   else
       flag=1;
   end
end
clear A
clear G_staz

%|||||||||||| DATI |||||||||||||||
dt=300; t_iniz=t_iniz+dt;
TBC_in= [zeros(1,t_iniz/dt-1) 273.15+[110:0.2:130 130:-0.2:110]];
TG=273.15+10;
```

```
cp_w=4186;
rho_w_m=951.2; rho_w_r=983.3;
%i per i gradi di compattazione
%j per il tempo
%z per gli altri cicli interni al ciclo del tempo

%|||||||||||||||| RETE DERAMIFICATA ||||||||||||||||||||
A_m=A_m_3d(:,:,1); n_nodi=size(A_m,1); n_rami=size(A_m,2);
A_r=-A_m;
L_m=ML(1,:); L_r=L_m;
H_m=MH_m(1,:); H_r=MH_r(1,:);
d_int_m=Md_int_m(1,:); d_int_r=Md_int_r(1,:);
sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
T_prec_m=MTIC_m_comp(1,:); T_prec_r=MTIC_r_comp(1,:); % IC

%trovo il nodo di ingresso e di uscita dalla rete
for j=1:n_nodi
    if isempty(find(A_m(j,:)==-1))
        inlet_node_m=j;
    elseif isempty(find(A_m(j,:)==1))
        outlet_node_m=j;
    end
end
inlet_node_r=outlet_node_m;
outlet_node_r=inlet_node_m;
Gext_m_rif=Gext_m;
Gext_m(inlet_node_m)=-sum(Gext_m); Gext_r=-Gext_m;
%Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete (
quindi compattazione )e non dal tempo
Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
SLsum_m=Hsum_m; SLsum_r=Hsum_m;
for j=1:n_nodi
    if j~= inlet_node_m && j~=outlet_node_m
       rami_coll=find(A_m(j,:));
       Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
       Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
       SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
       SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
    end
end

%modifico la matrice dei carichi
for i=t_iniz/dt:1:t_fin/dt
    for j=1:n_nodi
        if Gext_m(j)~=0 && Mcarichi(i,j)==0
           Mcarichi(i,j)=5e4;
        end
    end
end
n_operazioni_der=0;
%--risoluzione problema fluidodinamico mandata--
tic
G_m=A_m\(-Gext_m);
%--risoluzione problema fluidodinamico ritorno
G_r=A_r\(-Gext_r);
n_operazioni_der=n_operazioni_der+toc;
%--inizializzazione
Q_solo_der=zeros(length(dt:dt:t_fin),1);
```

```
T_r_centr_der=Q_solo_der;
carichi_tempo=Q_solo_der;
somma_scost_Q_der=0;
%--risoluzione problema termico per mandata e ritorno--
for j=t_iniz/dt:1:t_fin/dt
    %leggo i dati dei carichi
    carichi=Mcarichi(j,:);
    %--risoluzione problema termico mandata--
    tic
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_m && z~=outlet_node_m
           ramo_in=find(A_m(z,:)==-1);
           ramo_out=find(A_m(z,:)==1);
           nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
           k(z,nodo_prec)=-cp_w*G_m(ramo_in);
           f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
        end
    end
    %BC inlet node
    k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in(j);
    %BC outlet_node
    ramo_in=find(A_m(outlet_node_m,:)==-1);
    nodo_prec=find(A_m(:,ramo_in)==1);
    T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
    k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
    T_m=k\f;
    T_prec_m=T_m;
    %cerco la temperatura di uscita dalle utenze
    Tut_out=zeros(n_nodi,1);
    for z=1:n_nodi
        if Gext_m(z)>0
           Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
        end
    end

    %--risoluzione problema termico ritorno--
    k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
    for z=1:n_nodi
        if z~=inlet_node_r && z~=outlet_node_r
           ramo_in=find(A_r(z,:)==-1);
           ramo_out=find(A_r(z,:)==1);
           nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
           k(z,nodo_prec)=-cp_w*G_r(ramo_in);
           f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
        end
```

```
    end
    %BC inlet node
    k(inlet_node_r,inlet_node_r)=1; f(inlet_node_r)=Tut_out(inlet_node_r);
    %BC outlet_node
    ramo_in=find(A_r(outlet_node_r,:)==-1);
    nodo_prec=find(A_r(:,ramo_in)==1);
    T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
    k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
    %risoluzione
    T_r=k\f;
    T_prec_r=T_r;
    n_operazioni_der=n_operazioni_der+toc;
    Q_solo_der(j)=abs(Gext_m(inlet_node_m))*cp_w*(T_m(inlet_node_m)-
T_r(outlet_node_r));
    carichi_tempo(j)=sum(carichi);
    T_r_centr_der(j)=T_r(outlet_node_r);
    somma_scost_Q_der=somma_scost_Q_der+abs(Q_nod(j)-Q_solo_der(j));
end
figure(1)
hold on
grid on
box on
plot(dt:dt:t_fin,carichi_tempo,dt:dt:t_fin,Q_solo_der);
plot(dt:dt:t_fin,Q_nod,'k--');
legend('carico','potenza in ingresso nella rete deramificata','potenza in
ingresso nella rete originale','location','neo');
title('potenze con 154 rami (no ramificazioni)');

n_rami_in=n_rami;
%|||||||||||||||| RETE COMPATTATA ||||||||||||||||||||
comp_plot=[1  20  40  60  80  100  120  140  149  150  151  152];  %sono  le
compattazioni che voglio plottare
n_operazioni_comp=zeros(length(comp_plot),1);
%inizializzazione variabili da stampare
Q_comp_trans=zeros(length(comp_plot),length(dt:dt:t_fin));
Trit_centr_comp=Q_comp_trans;
somma_scost_Q=zeros(length(comp_plot),1);
v_n_rami=somma_scost_Q;
for i=1:length(comp_plot)
    gc=comp_plot(i); %grado di compattazine
    n_rami=n_rami_in-gc;
    v_n_rami(i)=n_rami;
    n_nodi=n_rami+1; %no loop
    %negli indicatori metto gc+1 invece che gc perchè i valori 1 si
    %riferivano al grad di compattazione zero
    A_m=A_m_3d(1:1:n_nodi,1:1:n_rami,gc+1);
    A_r=-A_m;
    L_m=ML(gc+1,1:1:n_rami); L_r=L_m;
    H_m=MH_m(gc+1,1:1:n_rami); H_r=MH_r(gc+1,1:1:n_rami);
    d_int_m=Md_int_m(gc+1,1:1:n_rami); d_int_r=Md_int_r(gc+1,1:1:n_rami);
    sez_int_m=0.25*pi*d_int_m.^2; sez_int_r=0.25*pi*d_int_r.^2;
    Vol_int_m=sez_int_m.*L_m; Vol_int_r=sez_int_r.*L_r;
```

```
    T_prec_m=MTIC_m_comp(i+1,1:1:n_nodi);
T_prec_r=MTIC_r_comp(i+1,1:1:n_nodi); % IC

    %trovo i nodi di ingresso e di uscita dalla rete
    for j=1:n_nodi
        if isempty(find(A_m(j,:)==-1))
            inlet_node_m=j;
        elseif isempty(find(A_m(j,:)==1))
            outlet_node_m=j;
        end
    end
    inlet_node_r=outlet_node_m;
    outlet_node_r=inlet_node_m;

    %Trovo Hsum ed SLsum per mandata e ritorno che dipendono solo dalla rete
( quindi compattazione )
    %e non dal tempo
    Hsum_m=zeros(n_nodi,1); Hsum_r=Hsum_m;
    SLsum_m=Hsum_m; SLsum_r=Hsum_m;
    for j=1:n_nodi
        if j~= inlet_node_m && j~=outlet_node_m
           rami_coll=find(A_m(j,:));
           Hsum_m(j)=H_m(rami_coll(1))+H_m(rami_coll(2));
           Hsum_r(j)=H_r(rami_coll(1))+H_r(rami_coll(2));
           SLsum_m(j)=Vol_int_m(rami_coll(1))+Vol_int_m(rami_coll(2));
           SLsum_r(j)=Vol_int_r(rami_coll(1))+Vol_int_r(rami_coll(2));
        end
    end
    %settiamo la portata uscente relativa a 155 nodi (senza il valore
negativo al nodo di ingresso)
    Gext_m=Gext_m_rif;
    %vario le portate uscenti mediante i coeff della matrice MM
    for z=1:gc
        portata_A=Gext_m(MM(z,1))+MM(z,5)*Gext_m(MM(z,2));
        portata_B=(1-MM(z,5))*Gext_m(MM(z,2))+(1-MM(z,6))*Gext_m(MM(z,3));
        portata_D=Gext_m(MM(z,4))+MM(z,6)*Gext_m(MM(z,3));
        Gext_m(MM(z,1))=portata_A;
        Gext_m(MM(z,2))=portata_B;
        Gext_m(MM(z,4))=portata_D;
        Gext_m(MM(z,3))=[];
    end
    G_f=Gext_m(inlet_node_m);
    Gext_m(inlet_node_m)=-sum(Gext_m)+G_f; Gext_r=-Gext_m;
    %--risoluzione problema fluidodinamico mandata--
    tic
    G_m=A_m\(-Gext_m);
    n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
    %--risoluzione problema fluidodinamico ritorno
    tic
    G_r=A_r\(-Gext_r);
    n_operazioni_comp(i)=n_operazioni_comp(i)+toc;
    for j=t_iniz/dt:1:t_fin/dt
        %leggo i dati dei carichi
        carichi=Mcarichi(j,:);
        %li vario mediante la matrice dei coefficienti MM
        for z=1:gc
            carico_A=carichi(MM(z,1))+MM(z,5)*carichi(MM(z,2));
            carico_B=(1-MM(z,5))*carichi(MM(z,2))+(1-
MM(z,6))*carichi(MM(z,3));
```

```
            carico_D=carichi(MM(z,4))+MM(z,6)*carichi(MM(z,3));
            carichi(MM(z,1))=carico_A;
            carichi(MM(z,2))=carico_B;
            carichi(MM(z,4))=carico_D;
            carichi(MM(z,3))=[];
        end

        %--risoluzione problema termico mandata
        tic
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_m && z~=outlet_node_m
               ramo_in=find(A_m(z,:)==-1);
               ramo_out=find(A_m(z,:)==1);
               nodo_prec=find(A_m(:,ramo_in)==1);

k(z,z)=0.5*SLsum_m(z)*rho_w_m*cp_w/dt+cp_w*(G_m(ramo_out)+Gext_m(z))+0.5*Hsu
m_m(z);
               k(z,nodo_prec)=-cp_w*G_m(ramo_in);

f(z)=0.5*Hsum_m(z)*TG+0.5*SLsum_m(z)*rho_w_m*cp_w/dt*T_prec_m(z);
            end
        end
        %BC inlet node
        k(inlet_node_m,inlet_node_m)=1; f(inlet_node_m)=TBC_in(j);
        %BC outlet_node
        ramo_in=find(A_m(outlet_node_m,:)==-1);
        nodo_prec=find(A_m(:,ramo_in)==1);
        T_med_prec=(T_prec_m(outlet_node_m)+T_prec_m(nodo_prec))/2;

k(outlet_node_m,outlet_node_m)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt+cp_w*G
_m(ramo_in)+0.5*H_m(ramo_in);
        k(outlet_node_m,nodo_prec)=0.5*Vol_int_m(ramo_in)*rho_w_m*cp_w/dt-
cp_w*G_m(ramo_in)+0.5*H_m(ramo_in);

f(outlet_node_m)=Vol_int_m(ramo_in)*rho_w_m*cp_w/dt*T_med_prec+H_m(ramo_in)*
TG;
        T_m=k\f;
        T_prec_m=T_m;
        %cerco la temperatura di uscita dalle utenze
        Tut_out=zeros(n_nodi,1);
        for z=1:n_nodi
            if Gext_m(z)~=0 && z~=inlet_node_m
               Tut_out(z)=T_m(z)-carichi(z)/Gext_m(z)/cp_w;
            end
            if z==inlet_node_m && G_f~=0
               Tut_out(z)=T_m(z)-carichi(z)/G_f/cp_w;
            end
        end

        %--risoluzione problema termico ritorno--
        k=zeros(n_nodi,n_nodi); f=zeros(n_nodi,1);
        for z=1:n_nodi
            if z~=inlet_node_r && z~=outlet_node_r
               ramo_in=find(A_r(z,:)==-1);
               ramo_out=find(A_r(z,:)==1);
               nodo_prec=find(A_r(:,ramo_in)==1);

k(z,z)=0.5*SLsum_r(z)*rho_w_r*cp_w/dt+cp_w*G_r(ramo_out)+0.5*Hsum_r(z);
```

```
                k(z,nodo_prec)=-cp_w*G_r(ramo_in);

f(z)=0.5*Hsum_r(z)*TG+0.5*SLsum_r(z)*rho_w_r*cp_w/dt*T_prec_r(z)-
cp_w*Gext_r(z)*Tut_out(z);
            end
        end
        %BC inlet node
        k(inlet_node_r,inlet_node_r)=1;
f(inlet_node_r)=Tut_out(inlet_node_r);
        %BC outlet_node
        ramo_in=find(A_r(outlet_node_r,:)==-1);
        nodo_prec=find(A_r(:,ramo_in)==1);
        T_med_prec=(T_prec_r(outlet_node_r)+T_prec_r(nodo_prec))/2;

k(outlet_node_r,outlet_node_r)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt+cp_w*G
_r(ramo_in)+0.5*H_r(ramo_in);
        k(outlet_node_r,nodo_prec)=0.5*Vol_int_r(ramo_in)*rho_w_r*cp_w/dt-
cp_w*G_r(ramo_in)+0.5*H_r(ramo_in);

f(outlet_node_r)=Vol_int_r(ramo_in)*rho_w_r*cp_w/dt*T_med_prec+H_r(ramo_in)*
TG;
        %risoluzione
        T_r=k\f;
        ultimo_ramo_r=find(A_r(outlet_node_r,:)==-1);

T_r(outlet_node_r)=(G_r(ultimo_ramo_r)*T_r(outlet_node_r)+G_f*Tut_out(outlet
_node_r))/(G_r(ultimo_ramo_r)+G_f);
        T_prec_r=T_r;
        n_operazioni_comp(i)=n_operazioni_comp(i)+toc;

Q_comp_trans(i,j)=cp_w*(G_f+abs(Gext_m(inlet_node_m)))*(T_m(inlet_node_m)-
T_r(outlet_node_r));
        Trit_centr_comp(i,j)=T_r(outlet_node_r);
        somma_scost_Q(i)=somma_scost_Q(i)+abs(Q_nod(j)-Q_comp_trans(i,j));
     end

end

load Q_nod
figure(2)
hold on
grid on
box on
plot([t_iniz:dt:t_fin]./3600,carichi_tempo(t_iniz/dt:1:t_fin/dt),[t_iniz:dt:
t_fin]./3600,Q_solo_der(t_iniz/dt:1:t_fin/dt));
for i=1:length(comp_plot)
    plot([t_iniz:dt:t_fin]./3600,Q_comp_trans(i,t_iniz/dt:1:t_fin/dt))
end
plot([t_iniz:dt:t_fin]./3600,Q_nod(t_iniz/dt:1:t_fin/dt),'k--');
legend('carico  totale','154  rami','153  rami','134  rami','114  rami','94
rami','74 rami','54 rami','34 rami','14 rami','5 rami','4 rami','3 rami','2
rami','rete originale','location','neo');
title('Potenza da fornire alla rete');
xlabel('tempo [h]'); ylabel('[W]');

load Trit_centr_nod
figure(3)
hold on
grid on
```

```
box on
plot([t_iniz:dt:t_fin]./3600,T_r_centr_der(t_iniz/dt:1:t_fin/dt));
for i=1:length(comp_plot)
    plot([t_iniz:dt:t_fin]./3600,Trit_centr_comp(i,t_iniz/dt:1:t_fin/dt));
end
plot([t_iniz:dt:t_fin]./3600,Trit_centr_nod(t_iniz/dt:1:t_fin/dt),'k--');
legend('154 rami','153 rami','134 rami','114 rami','94 rami','74 rami','54
rami','34  rami','14  rami','5  rami','4  rami','3  rami','2  rami','rete
originale','location','neo');
title('Temperatura di ritorno in centrale');
xlabel('tempo [h]'); ylabel('[K]');

v_n_rami=[154;v_n_rami];
somma_scost_Q_perc=[somma_scost_Q_der;somma_scost_Q]/sum(Q_nod)*100;
figure(4)
hold on
grid on
box on
plot(v_n_rami,somma_scost_Q_perc,'bo-','markersize',3);
xlabel('numero rami'); ylabel('%');
set(gca,'xtick',sort(v_n_rami),'xticklabel',{'2','','','5','14','34','54','7
4','94','114','134','','154'});
title('Errore');

n_operazioni_comp=[n_operazioni_der; n_operazioni_comp];
figure(5)
hold on
grid on
box on
plot(v_n_rami,n_operazioni_comp,'bo-','markersize',3);
xlabel('numero rami'); ylabel('numero di operazioni');
set(gca,'xtick',sort(v_n_rami),'xticklabel',{'2','','','5','14','34','54','7
4','94','114','134','','154'});
title('Costo computazionale');
```

RIFERIMENTI

[1] Sito web : http://ambiente.comune.forli.fc.it/p/energia/fonti-di-energia-rinnovabile/teleriscaldamento/1-cosa-e-il-teleriscaldamento-e-quali-sono-i-suoi-vantaggi/

[2] Autori : A. Benonysson, I.B. Bohm, H.F. Ravn
Titolo : Optimization in a district heating system
Elsevier

[3] Autori: Elisa Guelpa, Claudia Toro, Adriano Sciacovelli, Roberto Melli, Enrico Sciubba, Vittorio Verda
Titolo : Optimal operation of large district heating networks through fast fluid-dynamic simulation
Elsevier

[4] Autori : Helge V.Larsen, Halldór Pálsson, Benny Bøhm, Hans F.Ravn
Titolo : Aggregated dynamic simulation model of district heating networks
Elsevier

[5] Autori : A. Sciacovelli, V. Verda, R. Borchiellini
Titolo : Numerical design of thermal systems
Clut

BIBLIOGRAFIA

[1] Autori : A. Sciacovelli, V. Verda, R. Borchiellini
Titolo : Numerical design of thermal systems
Clut

[2] Autori : Halldór Pálsson, Helge V. Larsen, Benny Bøhm, Hans F. Ravn, JiJun Zhou
Titolo: Equivalent models of district heating systems for on-line minimization of operational costs of the complete district heating system
Halldór Pálsson (Editor)

[3] Autori : Benny Bøhm (Project Leader), Seung-kyu Ha, Won-tae Kim, Bong-kyun Kim, Tiina Koljonen, Helge V. Larsen, Michael Lucht, Yong-soon Park, Kari Sipilä, Michael Wigbels Magnus Wistbacka
Titolo : Simple Models for Operational Optimisation
Editor : Benny Bøhm

[4] Autori : Benny Bøhm, Helge V. Larsen
Titolo : Simple models of district heating systems for load and demand side management and operational optimisation

ARTICOLI

[1] Autori : A. Benonysson, I.B. Bohm, H.F. Ravn
Titolo : Optimization in a district heating system
Elsevier

[2] Autori: Elisa Guelpa, Claudia Toro, Adriano Sciacovelli, Roberto Melli, Enrico Sciubba, Vittorio Verda.
Titolo : Optimal operation of large district heating networks through fast fluid-dynamic simulation
Elsevier

[3] Autori : Helge V.Larsen, Halldór Pálsson, BennyBøhm, Hans F.Ravn
Titolo: Aggregated dynamic simulation model of district heating networks
Elsevier

[4] Autori: Guelpa Elisa, Sciacovelli Adriano, Verda Vittorio
Titolo: Thermo-fluid dynamic model of large district heating networks for the analysis of primary energy savings
Elsevier

[5] Autori : Helge VLarsen, Benny Bøhm, Michael Wigbels
Titolo : A comparison of aggregated models for simulation and operational optimisation of district heating networks
Elsevier

[6] Autori : Dušan Strušnik, Jurij Avsec
Titolo : Artificial neural networking model of energy and exergy district heating mony flows
Elsevier