



**Politecnico  
di Torino**

Master of Science program in ARCHITECTURE FOR THE  
SUSTAINABILITY DESIGN

Thesis of Master's Degree

# **From Building Information Models to Building Performance Simulation:**

**Parametric workflow for high performance building envelope  
design for daylight and glare analysis**

**Tutors:**

Prof. Valerio Roberto Maria Lo Verso

Prof. Fabio Favoino

**Co-tutor:**

Luigi Giovannini

**Candidate:**

Mila Shoshev

A.Y. 2021/2022



*I would like to dedicate this thesis to my loving family and friends. Special thanks to my parents Kiril and Vesna whose words of encouragement and generous love supported me throughout this whole process. My brother Lazar and his wife Ljupka who stood by me and enlighten my days with their amazing son Kiril, my nephew, who always brought a smile to my face, even when it was the hardest. Also I would like to dedicate this to my loving grandmother Mileva, who have been my biggest cheerleader. Finally I dedicate this work and give special thanks to all my friends who selflessly inspired me to aim high and who have always been there for me whilst pursuing my dreams.*



## **ACKNOWLEDGMENTS**

First and foremost, I would like to express my gratitude and appreciation to my supervisors prof. Valerio Roberto Maria Lo Verso and prof. Fabio Favoino for the thoughtful comments and recommendations on this thesis. Additionally, I thank Lugi Giovannini for the extraordinary help and support, but also for his generous availability throughout these past months.

Furthermore my sincerest thanks goes to my collaborator Iva Lazareska, before all being an amazing friend and supporter without whom during these years of studying architecture, wouldn't be the same. Moreover, I am thankful for her sincere dedication to the collaborated research, which resulted in an amazing experience.

Lastly, I would like to acknowledge the University of Politecnico di Torino, which allowed me to pursue my dreams in studying what I love and make my graduate experience truly amazing.



## **ABSTRACT**

Balancing the need for thermal comfort and sufficient daylight in buildings with regulatory provisions for energy savings is a major design problem that must be taken into account at the early design stages of decision-making. Given society's growing reliance on energy efficiency and user comfort, the parametric design community is today faced with a rising demand to incorporate sustainability-related performance parameters, most notably energy and daylight.

By integrating and organizing the design components simultaneously using Visual Programming Language (VPL) software tools and developing a seamless interoperability between Building Information Modeling (BIM) software programs and Building Performance Simulation Tools (BPSTs), parametric design effectively improves the design process. A broad range of BPSTs are available to designers, but they can be very complex and hence difficult to incorporate at an early point of the design process. But the usage of parametric modeling has greatly increased the designers' access to this optimization approach.

Following that, the purpose of this study is to determine the significance of BPSTs in the analysis of daylight and glare. Next, using a third-party plugin called Rhino Inside Revit, this thesis investigates what information is transferred from a 3D model created in BIM software (Revit) to a VPL software program (Grasshopper) in order to run building performance simulation analysis, specifically daylight and glare analysis. Additionally, it emphasizes how to utilize Rhino Inside Revit's capabilities and include all of its interface's features on Grasshopper. It focuses on performing the proper actions that will enable the user to complete the simulations smoothly and with less time-consuming work.



## LIST OF FIGURES

Figure 2.1 Roles of the building envelope.....	29
Figure 2.2 Ranking criteria concerning usability and graphical visualization of BPS interfaces .....	33
Figure 2.3 Interoperability sub criteria .....	34
Figure 2.4 Ranking the most important features of a simulation tool .....	35
Figure 2.5 Ranking the ten tools according to architects and engineers .....	35
Figure 2.6 Confidence parameters of architects with professional experience below 5 years .	36
Figure 2.7 Fields in which simulation tools are useful.....	36
Figure 2.8 DesignBuilder interface, daylight analysis results .....	40
Figure 2.9 Open Studio as SketchUp plug-in .....	42
Figure 2.10 Introduction to Daylight Simulation in DIVA .....	43
Figure 2.11 Climate Studio Rhino Plug-in .....	44
Figure 2.12 Luminance rendering of Sunlight House shown with photo-realistic and false colour images.....	45
Figure 2.13 Representation of light reception by human’s eye by ALFA .....	46
Figure 2.14 Interoperability and data loss from BIM to BEM using .gbXML and .IFC .....	49
Figure 2.15 Execution of algorithms when working with Dynamo Revit .....	51
Figure 2.16 Screenshot from Revit UI accessing Rhino with RhinoInside.....	53
Figure 2.17 Screenshot from Grasshopper UI accessing Revit-aware components.....	54
Figure 2.18 Revit-aware components icons .....	54
Figure 2.19 Honeybee diagram .....	56
Figure 2.20 Ladybug diagram .....	56
Figure 2.21 Pollination workflow.....	57
Figure 2.22 Capabilities of Pollination to link multiple tools .....	58
Figure 2.23 Daylight Factor calculation .....	60
Figure 2.24 Daylight Autonomy calculation .....	60
Figure 2.25 Continuous Daylight Autonomy calculation.....	61
Figure 2.26 Continuous Daylight Autonomy calculation.....	61
Figure 2.27 Useful Daylight Illuminance calculation, representing the generally considered lower and upper thresholds.....	62
Figure 2.28 Daylight Glare Probability calculation.....	63
Figure 2.29 DGP Ranges .....	63

Figure 3.1 Conceptual workflow of the structure of methodology.....	67
Figure 3.2 TYPE I – Exporting the components and their material properties from Revit to Grasshopper .....	68
Figure 3.3 TYPE I – Exporting the rooms from Revit and then adjusting material properties in Grasshopper .....	69
Figure 3.4 Single-zone model .....	70
Figure 3.5 Screenshot from Revit UI, interception of wall surfaces.....	71
Figure 3.6 Step by step trim correction of perpendicular wall components in Revit .....	72
Figure 3.7 Screenshot from Revit UI, correction of wall trimming.....	72
Figure 3.8 Step by step correction of roof/floor boundaries in Revit .....	73
Figure 3.9 Screenshots from Revit UI, roof adjustments.....	73
Figure 3.10 Screenshots from Revit UI, floor adjustments.....	73
Figure 3.11 Screenshot from Revit UI, Edit Type for components .....	74
Figure 3.12 Screenshot from Revit UI, Edit Assembly .....	74
Figure 3.13 Screenshot from Revit UI, Type Properties.....	74
Figure 3.14 Screenshot from Revit UI, Type Properties.....	75
Figure 3.15 Screenshot from Grasshopper UI, component exportation type I .....	76
Figure 3.16 Screenshot from Grasshopper UI, component exportation type II.....	76
Figure 3.17 Screenshot from Grasshopper UI, component exportation type III.....	77
Figure 3.18 Screenshot from Grasshopper UI, constructing 4Point Surface for surface gaps	77
Figure 3.19 Screenshot from Rhino UI, closing a wall gap.....	78
Figure 3.20 Screenshot from Grasshopper UI, moving a surface to place .....	78
Figure 3.21 Screenshot from Rhino UI, constructing and moving a surface to place .....	79
Figure 3.22 Screenshot from Grasshopper UI, constructing sills and stiles .....	79
Figure 3.23 Screenshot from Rhino UI, splitting the surface for stiles and sills .....	80
Figure 3.24 Screenshot from Rhino UI, constructed indoor and outdoor stiles and sills .....	80
Figure 3.25 Screenshot from Grasshopper UI, importing materials from Revit to Grasshopper .....	81
Figure 3.26 Screenshot from Grasshopper UI, material assets .....	81
Figure 3.27 Screenshot from Grasshopper UI, HB Face and surface properties .....	82
Figure 3.28 Screenshot from Grasshopper UI, material properties .....	83
Figure 3.29 Screenshot from Grasshopper UI, HB Aperture and surface properties .....	84
Figure 3.30 Screenshot from Grasshopper UI, HB Room and Program Type .....	85

Figure 3.31 Screenshot from Grasshopper UI, outdoor stiles amd sills assigned as static shades in Grasshopper.....	86
Figure 3.32 Screenshot from Rhino and Grasshopper UI, shades application .....	86
Figure 3.33 Screenshot from Grasshopper UI, shades .....	87
Figure 3.34 Screenshot from Grasshopper UI, test simulations for dynamic shading .....	87
Figure 3.35 Screenshot from Grasshopper UI, test simulations grid .....	88
Figure 3.36 Screenshot from Grasshopper UI, test simulations translucent modifier.....	88
Figure 3.37 Screenshot from Grasshopper UI, test results DGP .....	89
Figure 3.38 Screenshot from Grasshopper UI, test simulations DGP .....	89
Figure 3.39 Screenshot from Grasshopper UI, final input for HB Model, dynamic shading schedule .....	90
Figure 3.40 Screenshot from Grasshopper UI, final HB Model with dynamic shading .....	90
Figure 3.41 Screenshot from Grasshopper UI, Daylight Factor simulation algorithm .....	92
Figure 3.42 Screenshot from Grasshopper UI, Annual Daylight simulation algorithm.....	92
Figure 3.43 Screenshot from Grasshopper UI, Annual Daylight Metrics occupancy schedule .....	93
Figure 3.44 Screenshot from Grasshopper UI, Visualization of simlulation results for daylight .....	93
Figure 3.45 Screenshot from Grasshopper UI, Spatial Daylight Autonomy (sDA) simulation .....	94
Figure 3.46 Screenshot from Grasshopper UI, Annual Sunlight Exposure simulation, schedule and analysis period .....	94
Figure 3.47 Screenshot from Grasshopper UI, Annual Sunlight Exposure calculation from results.....	95
Figure 3.48 Screenshot from Grasshopper UI, Annual Sunlight Exposure visualization .....	95
Figure 3.49 Screenshot from Grasshopper UI, Annual Sunlight Exposure visualization .....	95
Figure 3.50 Screenshot from Grasshopper UI, Daylight Glare Probability, assigning grids and views.....	96
Figure 3.51 Screenshot from Grasshopper UI, Daylight Glare Probability, assigning sky and parameters.....	97
Figure 3.52 Screenshot from Grasshopper UI, Daylight Glare Probability, producing results	97
Figure 3.53 Screenshot from Grasshopper UI, Glare Autonomy, simulation adjustments and visualizing the results .....	98
Figure 3.54 Screenshot from Grasshopper UI, Spatial Glare Autonomy, simulation adjustments .....	99
Figure 3.55 Multi-zone model .....	100

Figure 3.56 Screenshot from Revit UI, assigning rooms .....	101
Figure 3.57 Screenshot from Revit UI, selecting the rooms .....	101
Figure 3.58 Screenshot from Revit UI, category extraction .....	103
Figure 3.59 Screenshot from Revit UI, specifying room selection.....	103
Figure 3.60 Screenshot from Rhino UI, correctly extracted rooms .....	103
Figure 3.61 Screenshot from Revit UI, Wall System Family exportation type .....	104
Figure 3.62 Screenshot from Rhino UI, incorrectly extraxted room .....	104
Figure 3.63 Screenshot from Grasshopper UI, selecting surfaces's faces .....	105
Figure 3.64 Screenshot from Grasshopper UI, surface from points .....	105
Figure 3.65 Screenshot from Grasshopper UI, surface from faces .....	105
Figure 3.66 Screenshot from Grasshopper UI, surface from edges .....	106
Figure 3.67 Screenshot from Grasshopper UI, HB Face for opaque surfaces .....	107
Figure 3.68 Screenshot from Grasshopper UI, HB Aperture for transperent surfaces .....	108
Figure 3.69 Screenshot from Grasshopper UI, HB Opaque Material construction, Roof .....	109
Figure 3.70 Screenshot from Grasshopper UI, HB Opaque Material construction, Exterior Wall .....	110
Figure 3.71 Screenshot from Grasshopper UI, HB Opaque Material construction, Interior Wall .....	110
Figure 3.72 Screenshot from Grasshopper UI, HB Opaque Material construction, Floor ....	111
Figure 3.73 Screenshot from Grasshopper UI, HB Window Construction .....	111
Figure 3.74 Screenshot from Grasshopper UI, HB Room (Office - 1).....	112
Figure 3.75 Screenshot from Grasshopper and Rhino UI, HB Louver Shades .....	113
Figure 3.76 Screenshot from Grasshopper UI, HB Extruded Border Shades.....	113
Figure 3.77 Screenshot from Grasshopper UI, final HB Model with louver shades .....	114
Figure 3.78 Screenshot from Grasshopper UI, Grid with list of all inputted rooms.....	115
Figure 3.79 Screenshot from Grasshopper UI, Daylight Factor simulation algorithm II .....	115
Figure 3.80 Screenshot from Grasshopper UI, Annual Daylight simulation algorithm II.....	116
Figure 3.81 Screenshot from Grasshopper UI, Annual Daylight Metrics algorithm II .....	116
Figure 3.82 Screenshot from Grasshopper UI, Retrieving each room's values for calculating an average value.....	117
Figure 3.83 Screenshot from Grasshopper UI, ASE analysis II .....	118
Figure 3.84 Screenshot from Grasshopper UI, Daylight Autonomy, simulation adjustments and visualizing the results II .....	119
Figure 3.85 Screenshot from Grasshopper UI, Daylight Glare Probability II – Grid.....	120

Figure 3.86 Screenshot from Grasshopper UI, Daylight Glare Probability II - Simulation...	120
Figure 3.87 Screenshot from Grasshopper UI, Daylight Glare Probability II, producing results .....	121
Figure 3.88 Screenshot from Grasshopper UI, Spatial Glare Autonomy II, simulation adjustments.....	121
Figure 4.1 Daylight Factor (DF) heatmap for the single-zone model .....	123
Figure 4.2 DA for Scenario II – dynamic curtain.....	123
Figure 4.3 DA for Scenario I – no blinds .....	123
Figure 4.4 cDA for Scenario II – dynamic curtain .....	124
Figure 4.5 cDA for Scenario I – no blinds .....	124
Figure 4.6 sDA for Scenario II – dynamic curtain .....	125
Figure 4.7 sDA for Scenario I – no blinds.....	125
Figure 4.8 ASE for Scenario I – no blinds .....	125
Figure 4.9 UDI (0-100 lux) for Scenario II – dynamic curtain .....	126
Figure 4.10 UDI (0-100 lux) for Scenario I – no blinds.....	126
Figure 4.11 UDI (100-500 lux) for Scenario II – dynamic curtain .....	126
Figure 4.12 UDI (100-500 lux) for Scenario I – no blinds.....	126
Figure 4.13 UDI (500-3000 lux) for Scenario II – dynamic curtain .....	127
Figure 4.14 UDI (500-3000 lux) for Scenario I – no blinds.....	127
Figure 4.15 UDI (>3000 lux) for Scenario II – dynamic curtain .....	127
Figure 4.16 UDI (>3000 lux) for Scenario I – no blinds.....	127
Figure 4.17 GA for Scenario I – no blinds .....	128
Figure 4.18 GA for Scenario II – dynamic curtain.....	128
Figure 4.19 DGP results for single-zone model .....	129
Figure 4.20 sGA for Scenario II – dynamic curtain .....	130
Figure 4.21 sGA for Scenario II – no blinds .....	130
Figure 4.22 Single-zone model average annual UDI .....	132
Figure 4.23 Single-zone model DGP for June 20th and December 21st, at 9pm, 12pm and 3pm .....	133
Figure 4.24 Daylight Factor (DF) heatmap for the multi-zone model .....	135
Figure 4.25 Daylight Autonomy (DA) heatmap for the multi-zone model, no blinds.....	136
Figure 4.26 Daylight Autonomy (DA) heatmap for the multi-zone model, louver shades....	136
Figure 4.27 Continuous Daylight Autonomy (cDA) heatmap for the multi-zone model, no blinds .....	137

Figure 4.28 Continuous Daylight Autonomy (cDA) heatmap for the multi-zone model, louver shades .....	137
Figure 4.29 Spatial Daylight Autonomy (sDA) heatmap for the multi-zone model, no blinds .....	138
Figure 4.30 Spatial Daylight Autonomy (sDA) heatmap for the multi-zone model, louver shades .....	138
Figure 4.31 Annual Sunlight Exposure (ASE) heatmap for the multi-zone model, no blinds .....	139
Figure 4.32 Useful Daylight Illuminance (UDI) 0-100 lux, heatmap for the multi-zone model, no blinds.....	140
Figure 4.33 Useful Daylight Illuminance (UDI) 0-100 lux, heatmap for the multi-zone model, louver shades.....	140
Figure 4.34 Useful Daylight Illuminance (UDI) 100-500 lux, heatmap for the multi-zone model, no blinds.....	141
Figure 4.35 Useful Daylight Illuminance (UDI) 100-500 lux, heatmap for the multi-zone model, louver shades.....	141
Figure 4.36 Useful Daylight Illuminance (UDI) 500-3000 lux, heatmap for the multi-zone model, no blinds.....	142
Figure 4.37 Useful Daylight Illuminance (UDI) 500-3000 lux, heatmap for the multi-zone model, louver shades.....	142
Figure 4.38 Useful Daylight Illuminance (UDI) >3000 lux, heatmap for the multi-zone model, no blinds.....	143
Figure 4.39 Useful Daylight Illuminance (UDI) >3000 lux, heatmap for the multi-zone model, louver shades.....	143
Figure 4.40 Glare Autonomy (GA) heatmap for the multi-zone model, no blinds.....	144
Figure 4.41 Glare Autonomy (GA) heatmap for the multi-zone model, louver shades .....	144
Figure 4.42 DGP results for multi-zone model, Office 5.....	145
Figure 4.43 DGP results for multi-zone model, Conference Room 9.....	146
Figure 4.44 Spatial Glare Autonomy (sGA) heatmap for the multi-zone model, no blinds..	147
Figure 4.45 Spatial Glare Autonomy (sGA) heatmap for the multi-zone model, louver shades .....	147
Figure 4.46 Multi-zone model average annual UDI .....	150
Figure 4.47 Multi-zone model DGP for Office 5, June 20th and December 21st, at 9pm,12pm and 3pm.....	151
Figure 4.48 Multi-zone model DGP for Conference room 9, June 20th and December 21st, at 9pm,12pm and 3pm.....	151
Figure 5.1 Screenshot from Revit UI, window measurement changes.....	154

Figure 5.2 Screenshot from Rhino UI, 3D model with the window measurement changes ..	155
Figure 5.3 Daylight Factor (DF) heatmap for the single-zone model compared with the updated version .....	155
Figure 7.1 Vertical flowchart following the comparison of the two studies .....	159
Figure 8.1 Before practice implication of case studies.....	162
Figure 8.2 During practice implication of case studies .....	163
Figure 8.3 After practice implication of case studies .....	164

## LIST OF TABLES

Table 2.1 Radiance tool abilities.....	41
Table 2.2 Key Performance Indicators KPIs .....	59
Table 2.3 KPIs to Grasshopper tools .....	64
Table 3.1 Excel results from test simulations of DGP .....	89
Table 3.2 Excel results from test simulations of DGP (prepared schedule) .....	90
Table 3.3 Multi-zone model WFR calculation.....	102
Table 4.1 KPIs Results for the Single-zone model, both without shading element and dynamic curtain.....	131
Table 4.2 KPIs Results for the Multi-zone model, both without shading element and louver blinds.....	148
Table 4.3 KPIs Results for the Multi-zone model, both without shading element and louver blinds, overall building .....	149

## LIST OF ACRONYMS

AADCC	Accuracy of tools and Ability to simulate Detailed and Complex and building Components
ADP	Abiotic Depletion Potential
AEC	architecture, engineering, and construction
AEC	Atomic Energy Commission
AP	Acidification Potential
API	Application Program Interface
ASE	Annual Sunlight Exposure
ASHRAE	American Society of Heating, Refrigerating and Air-Conditioning Engineers
ASTM	American Society for Testing and Materials
AVF	Analysis Visualization Framework
BIM	Building Information Modeling
BPSTs	Building Performance Simulation Tools
CAD	Computer-Aided Design
CBDM	Climate-based daylight modeling
CBPD	Centre for Building Performance and Diagnostics
CDA	Continuous Daylight Autonomy
CEN	Comite European de Norme
CFD	Computational Fluid Dynamic
CPU	Central Processing Unit

CSV	Comma-Separated Values
DA	Daylight Autonomy
DB	Design Builder
DF	Daylight Factor
DGP	Daylight Glare Probability
DOE	Department Of Energy
DXF	Drawing Exchange Format
EN	European Standards
EP	Energy Plus
EP	Eutrophication Potential
EPBD	Energy Performance of Buildings Directive
EPW	Energy Plus Weather
GA	Glare Autonomy
GBXML	Green Building XML schema
GH	Grasshopper
GUI	Graphical User Interface
GWP	Global Warming Potential
HB	Honeybee
HDR	High Dynamic Range
HVAC	Heating, Ventilation, Airconditioning and Cooling
IAQ	Indoor Air Quality

IBDP	Integration with Building Design Process
IBM	Interoperability of Building Modelling
IES	International Education Standard
IFC	International Finance Corporation.
IGDB	Institute of Genetics and Developmental Biology
IIKB	Integration of Intelligent design Knowledge-Base
ISO	International Organization for Standardization
KPI	Key Performance Indicator
LB	Ladybug
LCA	Life Cycle Assessment
LEED	Leadership in Energy and Environmental Design
LGPL	GNU Lesser General Public License
LISP	LISt Processing
MEP	Mechanical, Electrical and Plumbing
NE	North East
NW	North West
NZEB	Nearly Zero-Energy Buildings
ODP	Ozone Depletion Potential
PMV	Predicted Mean Vote
POCP	Photochemical Ozone Creation Potential
PPD	Predicted Percentage of Dissatisfied

RGB	Read, Green and Blue
RIR	Rhino Inside Revit
RQ	Research Question
RT	Reverberation time
SDA	Spatial Daylight Autonomy
SDK	Software Development Kit
SE	South East
SGA	Spatial Glare Autonomy
SKP	Sketch Up
SW	South West
UDI	Useful Daylight Illuminance
UI	User Interface
UIM	Usability and Information Management (UIM) of interfaces
UNI	Italian National Unification
UTCI	Universal Thermal Climate Index
VOC	Volatile Organic Compounds
VPL	Visual Programming Language
WDP	Water Deprivation Potential
WFR	Window to Floor Ratio
ZEB	Zero-Energy Buildings

# INDEX

<b>1. INTRODUCTION .....</b>	<b>25</b>
1.1. RESEARCH MOTIVATION.....	26
1.2. RESEARCH BACKGROUND.....	26
1.3. RESEARCH GOALS .....	27
1.4. AIM AND OBJECTIVES .....	27
1.5. STRUCTURE OF THE THESIS .....	28
<b>2. LITERATURE REVIEW .....</b>	<b>29</b>
2.1. IMPORTANCE OF BUILDING ENVELOPE FOR BUILDING PERFORMANCE.....	29
2.2. NZEB FROM A VISUAL COMFORT POINT OF VIEW .....	30
2.3. BUILDING PERFORMANCE SIMULATION TOOLS (BPSTs).....	31
2.3.1. IMPORTANCE OF BPSTs.....	32
2.3.2. FIVE MAJOR CHALLENGES .....	32
2.3.3. USER SURVEY RESULTS .....	33
2.4. INTRODUCTION TO BUILDING INFORMATION MODELING SOFTWARE PROGRAMS.....	37
2.4.1. USE OF REVIT AS BIM SOFTWARE FOR DAYLIGHT ANALYSIS ..	37
2.5. DAYLIGHT ANALYSIS SIMULATION TOOLS .....	39
2.5.1. DESIGN BUILDER.....	39
2.5.2. RADIANCE .....	40
2.5.3. DAYSIM.....	41
2.5.4. OPEN - STUDIO .....	42
2.5.3. DIVA FOR RHINO .....	42
2.5.4. CLIMATE STUDIO .....	44
2.5.5. VELUX DAYLIGHT VISUALIZER .....	45
2.5.6. DIALUX EVO .....	46
2.5.7. ALFA .....	46
2.6. INTEROPERABILITY BETWEEN BIM AND DAYLIGHT ANALYSIS SIMULATION TOOLS .....	47
2.6.1. BIM DATA CONVERSION MODE AND DATA LOSS .....	47
2.7. VISUAL PROGRAMING LANGUAGE SOFTWARE PROGRAMS.....	50
2.7.1. POSSIBLE WORKFLOWS FROM BIM TO VPL.....	50

2.7.2.	APPLICATION OF VPL THROUGH DYNAMO .....	51
2.7.3.	APPLICATION OF VPL THROUGH GRASSHOPPER.....	52
2.7.4.	RHINO INSIDE REVIT .....	53
2.7.5.	GRASSHOPPER INSIDE REVIT .....	53
2.7.6.	GRASSHOPPER PLUGINS LADYBUG TOOLS.....	55
2.8.	POLLINATION FOR FUTURE ANALYSIS .....	57
2.9.	KEY PERFORMANCE INDICATORS (KPIS) .....	58
2.10.	DAYLIGHT .....	60
2.10.1.	DAYLIGHT FACTOR (DF) .....	60
2.10.2.	DAYLIGHT AUTONOMY (DA).....	60
2.10.3.	CONTINUOUS DAYLIGHT AUTONOMY (CDA) .....	61
2.10.4.	SPATIAL DAYLIGHT AUTONOMY (SDA) .....	61
2.10.5.	ANNUAL SUNLIGHT EXPOSURE (ASE).....	61
2.10.6.	USEFUL DAYLIGHT ILLUMINANCE (UDI).....	62
2.11.	GLARE .....	63
2.11.1.	DAYLIGHT GLARE PROBABILITY (DGP) .....	63
2.11.2.	GLARE AUTONOMY (GA) .....	63
2.12.	VIEW-OUT.....	64
2.13.	USING GRASSHOPPER PLUGINS FOR KPIS ANALYSIS .....	64
<b>3.</b>	<b>METHODOLOGY .....</b>	<b>65</b>
3.1.	DESCRIPTION OF RESEARCH METHODS .....	66
3.2.	CONCEPTUAL WORKFLOW .....	66
3.4.	WORKFLOW TYPES USING RHINO INSIDE REVIT AS INTEROPERABILITY TOOL .....	67
3.4.1.	TYPE I – EXPORTING THE COMPONENTS AND THEIR MATERIAL PROPERTIES FROM REVIT TO GRASSHOPPER .....	68
3.4.2.	TYPE II – EXPORTING THE ROOMS FROM REVIT AND THEN ADJUSTING MATERIAL PROPERTIES IN GRASSHOPPER .....	69
3.5.	CASE STUDY I – SINGLE-ZONE MODEL .....	70
3.5.1.	DESCRIPTION OF THE MODEL .....	70
3.5.2.	SIMULATION MODEL INPUTS IN REVIT .....	71
3.5.3.	TYPES OF COMPONENT IMPORTATION IN GRASSHOPPER .....	75
3.5.4.	GEOMETRY MODIFICATION IN GRASSHOPPER .....	77

3.5.5.	EXTRACTING THE MATERIAL OF REVIT MODEL COMPONENTS	80
3.5.6.	CREATING THE HONEYBEE MODEL .....	82
3.5.7.	DAYLIGHT SIMULATIONS .....	91
3.5.8.	GLARE SIMULATIONS .....	96
3.6.	CASE STUDY II – MULTI-ZONE MODEL .....	100
3.6.1.	DECRPTION OF MODEL.....	100
3.6.2.	ASSIGNING ROOMS IN REVIT .....	101
3.6.3.	IMPORTING ROOMS WITH RIR IN GRASSHOPPER.....	102
3.6.4.	GEOMETRY MODIFICATIONS ON GRASSHOPPER.....	104
3.6.5.	CREATING HONEYBEE MODEL.....	106
3.6.6.	DAYLIGHT SIMULATIONS .....	114
3.6.7.	GLARE SIMULATIONS .....	118
<b>4.</b>	<b>RESULTS .....</b>	<b>121</b>
4.1.	CASE STUDY I – SINGLE ZONE MODEL.....	122
4.1.1.	DAYLIGHT ANAYLISIS RESULTS.....	122
4.1.2.	GLARE ANALYSIS RESULTS .....	128
4.1.3.	COMMENTS ON THE RESULTS .....	131
4.2.	CASE STUDY II – MULTI-ZONE MODEL.....	134
4.2.1.	DAYLIGHT ANALYSIS RESULTS .....	134
4.2.2.	GLARE ANALYSIS RESULTS .....	144
4.2.3.	COMMENTS ON THE RESULTS .....	148
<b>5.</b>	<b>DISCUSSION .....</b>	<b>152</b>
5.1.	STRENGHTS AND LIMITATIONS OF THE RESEARCH.....	152
5.2.	POSIBILITY OF BACKWARDS INTEROPERABILITY .....	154
5.3.	PRACTICE IMPLICATIONS AND FUTURE RESEARCH .....	156
<b>6.</b>	<b>CONCLUSION .....</b>	<b>157</b>
<b>7.</b>	<b>COMMON DISCUSSION WITH COLABORATOR IVA LAZARESKA ....</b>	<b>158</b>
7.1.	COMPARISON OF STUDIES .....	160
7.2.	RECOMMENDATIONS AND IMPLICATION FOR FUTURE RESEARCH	161
<b>8.</b>	<b>COMMON CONCLUSION WITH COLLABORATOR IVA LAZARESKA</b>	<b>162</b>
<b>9.</b>	<b>REFERENCES.....</b>	<b>166</b>



# 1. INTRODUCTION

In recent years, Building Information Modeling (BIM), which is based on digital modeling and information management, has had a significant impact on the traditional construction process and has helped the industry transition to more industrial construction methods, which has reduced time and costs (Eastman, 2011).

Moreover, in modern architectural design practice and education, parametric design has emerged as a significant trend. The term describes a technique for digitally modeling a collection of design variations whose relationships to one another are defined by one or more mathematical parameters, which together create a parametric space that may contain a large number of distinct but related shapes. Floor area ratios or other constraints are typical performance-based design elements that need to be taken into account. The parametric design community currently faces a rising demand to incorporate sustainability-related performance characteristics, most notably energy and daylight, given society's growing reliance on energy efficiency and user comfort (Day, 2010).

The location, orientation, form, and lighting are only a few examples of the design parameters that can be applied to every architectural design. If a designer wants to change a parameter after the design is finished, they must follow the standard architectural process, which requires them to start over. In contrast, parametric design enhances the design process effectively by integrating and organizing the design elements at the same time using Visual Programming Language (VPL) software tools and developing a smooth interoperability between BIM software programs and Building Performance Simulation Tools (BPSTs) (Lagios, 2010).

On the other hand a crucial design problem that must be integrated into the early design stages decision-making, is balancing the demand for thermal comfort and adequate daylight conditions in buildings with regulatory provisions for energy savings. Lighting contributes to a building's self-efficiency and consumes 15% of the energy used by buildings globally. The use of daylight effectively can cut overall energy consumption and is a crucial part of the illumination of buildings. The most important decisions about daylight building design are made during the conceptual stage and the major approaches. It's essential to provide architects and designers a set of instructions in order to obtain optimal daylight analysis results because of the rapid development of cities and the requirement for more residential construction (Choi, 2016).

Of course, daylight simulations have been used for years, and an expanding variety of design analysis methods are available to translate scene geometries and material descriptions from nearly any BIM modeler into different daylight simulation software. Designers have access to a wide variety of BPSTs, but they can be extremely complicated and hence challenging to integrate at an early stage of the design process. This optimization technique has been made incredibly accessible to the designers with the use of parametric modelling (Leslie, 2012).

This thesis examines what kind of information is transferred from a 3D model, constructed in BIM software program (Revit), to a VPL software program (Grasshopper), in order to run building performance simulation analysis, more precisely daylight and glare analysis, while using a third-party plugin called Rhino Inside Revit.

## 1.1. RESEARCH MOTIVATION

Starting with a knowledge which was gained previously, while having a course of Parametric modeling of a Building Envelope during the last semester, truly was an incentive to try and explore the abilities of simulation tools with a focus on several aspects of building performance. Personally, the complexity and diversity of software programs, which are more and more used by professionals, immersed into exploring how to deepen the understanding of doing all of the analysis smoothly with not much time consuming. The initial information gathered about building performance simulation tools, was from doing a small project for the course and evaluating how the components of its building envelope performed with various specifications. Back then, Revit was introduced as a BIM tool and Grasshopper as a VPL software program. The curiosity grew much more when multiple studies came across on why this type of analysis is the future of assessing multiple scenarios at the same time.

## 1.2. RESEARCH BACKGROUND

The starting point of the research was clearly finding the common research regarding the general thesis topic. This means that at the beginning the research was done in partnership with another colleague (Iva Lazareska\*), where we tried to accumulate all the information and knowledge we had on this subject and then decided to divide the research into two parts, where their results meet again at the end as part of the discussion and conclusion chapter, comparing two types of workflows assessing visual comfort on one side and energy and thermal comfort on the other. The part on which this research continued was finding the right software for smoother interoperability between BIM and VPL software program, while doing the Visual comfort analysis. What was really interesting was finding why the building envelope plays a big role in the entire building performance, when doing visual comfort analysis. Facades have significantly changed during the history of building design and construction, taking on autonomous, conspicuous architectural responsibilities in relation to the structural component of the building fabric. Initially, the facade offered structural resistance and shelter because it was constructed from a single product, a single product, or a product combination, such stone or bricks and mortar. Movable features, like windows, doors, or shutters, kept rain and wind out of openings. Infill walls replaced stacked walls as a result of the requirement for resource efficiency, which laid the groundwork for the separation of the facade's primary load-bearing role. The facade system evolved into a more complicated system with numerous layers to ensure the achievement of improved performance and climate control levels along with unique indoor environment comfort criteria and energy consumption issues (Knaack, 2014). On another topic there was a big hesitation whether there are people that are ready to use BPSTs, but they simply don't have the experience due to the lack of knowledge or findings that stopped them from exploring another unknown side. Studies of a student's capacity to use

---

\* Iva Lazareska (student) Master of Science Program in Architecture for the Sustainability Design at "Politecnico di Torino" 2019-2022

or interpret simulations are conducted in Germany. These findings show that there is an immediate need to do research in Europe to incorporate BPSTs technologies into the architectural field (Souza, 2019). On the other hand, since experts struggle with both the interfaces and the analysis parameters, there needs to be a change in architectural education. Better communication between architecture and other disciplines will result from the incorporation of building performance simulation technologies in university curricula (Thomas, 2009). Continuing with the second part, is introducing the potential of such tools to compute daylight and glare analysis. Research, as well as observations of human behavior and workplace design, provide evidence that daylight is advantageous. Buildings' daylight-admitting windows are crucial for the perspective and connection they offer to the outside world. In addition, the quality, spectral makeup, and fluctuation of daylight are significant. According to an assessment of how individuals respond to indoor spaces, sunshine is wanted because it satisfies two very fundamental human needs: being able to see a task and the area clearly and having some environmental stimulation. Long-term use of electric illumination is seen to be unhealthy; working outside in the daylight is thought to be less stressful and uncomfortable (Boyce, 2003). However a gap exists within the interoperability between BIM and VPL software programs, due to the data loss and not always smooth workflow, strictly for analyzing visual comfort key performance indicators of building envelope.

### **1.3. RESEARCH GOALS**

Under research goals the main points to achieve during the research and the results to have at the end of this research are included. The four leading goals are:

- To understand and address the importance of Building Performance Simulation Tools;
- To explore and later explain step by step workflow between Revit and Grasshopper, using RhinoInside, especially focusing on simulating the daylight and glare analysis;
- To find and address the errors and data loss within the interoperability;
- To analyze challenges, possibilities and future developments.

### **1.4. AIM AND OBJECTIVES**

Firstly, the study aims to investigate how important are BPSTs into daylight and glare analysis. Moreover, to explore the capability of BIM softwares to produce a 3D model corresponding to the needs of further evaluation in VPL software program and RhinoInside as a plugin used for the interoperability, because it shows its positive and negative sides of errors and data loss.

Therefore, the four main research questions are:

RQ1 What is the importance of BPSTs for Visual Comfort analysis?

RQ2 What are the steps of achieving smooth interoperability between BIM and VPL software programs?

RQ3 Is smooth interoperability a key factor for assessing building envelope performance studies?

RQ4 What are the pros and cons of Rhino Inside Revit for Daylight and Glare analysis?

This study investigates how to transmit data from a 3D model, a building with its components specifications, to a VPL software program for later visual comfort analysis. Moreover, how to utilize the capabilities of Rhino Inside Revit and incorporate all the tools provided within its interface on Grasshopper. It focuses on taking the right steps which will provide the user with not much time consuming work and ability to perform the simulations smoothly. When speaking of the objectives of this study, there are few that were followed in order to solve the previously given problems, such as:

- Constructing the 3D model in Revit with making allowances for the correct modeling of its components;
- Exportation of the components in Grasshopper using Rhino Inside Revit;
- Making use of the components specifications, meaning observe their material properties and use the ones needed for visual comfort analysis;
- Building an algorithm within Grasshopper, using the components provided from Ladybug and Honeybee for daylight and glare analysis;
- Comparing the results from the case studies which have different technologies based on shading devices.

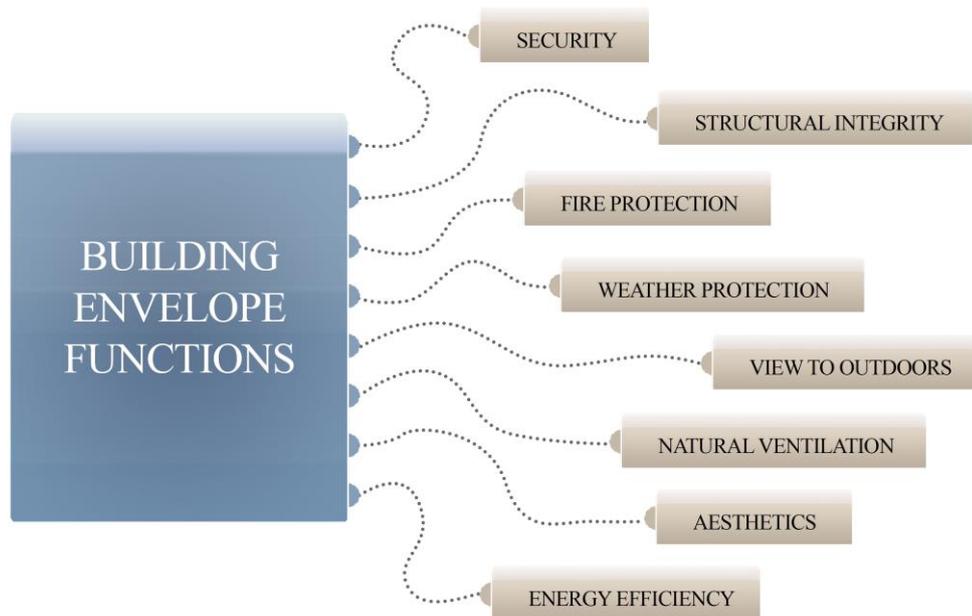
## **1.5. STRUCTURE OF THE THESIS**

The thesis is structured in multiple chapters according to what type of research was done. Following that, Chapter 2 focuses on a quick review about importance of building envelope while using BPSTs for visual comfort analysis. It continues with an explanation of daylight analysis simulation tools, considering BIM software programs as modeling tool and VPL as analyzing software programs. The interoperability between them, errors and data loss is also taking part into this chapter. Lastly, design impact factors and physical variables of visual comfort KPIs and briefly going through their meaning in the analysis of building envelope performance. Next, Chapter 3 is focused on two case studies (single-zone and multi-zone model), starting from making the models in Revit, importing the components in Grasshopper using Rhino Inside Revit, exploring the steps of developing an algorithm for correctly constructed Honeybee model for later analysis mainly using Ladybug and Honeybee as simulation tools and lastly running the simulations. Chapter 4 consists the results of the simulations, where they are presented with heatmaps, charts, tables and images. Finally in Chapter 5 and 6 there is a discussion and conclusion that explains the strengths, challenges, limitations and suggestions for future studies. Last addition to the structure are two Chapters 7 and 8 which are focusing on the common discussion and common conclusion with the collaborator. Here the main point is to explain how the two thesis went were explored together, tested separately and then at the end their pros and cons are mentioned.

## 2. LITERATURE REVIEW

The literature review initially focuses on the starting point addressed as importance of Building Envelope for Building Performance. Then the Building Performance Simulation Tools (BPSTs) and their use is elaborated. Moreover it goes deeply into the research based on the introduction into Building Information Modeling (BIM), Daylight Analysis Simulation Tools and Visual Programming Language (VPL) software programs. It explains the fundamental meaning of multiple software programs, which are taking part of the above mentioned groups, going into deeper clarification of the interoperability between them. Additionally it is briefly mentioned about conversion mode of the files, exchanged among these software programs, plus the data loss. Attention is brought to the software programs and their tools, which were used in the case studies examined in this thesis. On another note this section comprehensively elaborates the functional indicators for building envelope performance assessment. Meaning that it presents all of the design impact factors of Key Performance Indicators (KPIs), particularly focusing on visual comfort.

### 2.1. IMPORTANCE OF BUILDING ENVELOPE FOR BUILDING PERFORMANCE



*Figure 2.1 Roles of the building envelope*

A crucial aspect of developing green buildings is choosing the appropriate building envelope. The effectiveness of the internal architectural space is influenced by the design of a building envelope, the nature of its composition, and its integration with design patterns, shape, size, and distribution of the exterior openings. Such integration serves as insulation and a thermal barrier while also providing daylighting, decreasing glare, providing sight and vision, and shielding occupants from direct solar radiation (Rashwan, 2019). In comparison to conventional structures, green buildings' building envelopes attempt to accomplish a wider variety of goals and frequently operate in a different manner.

In addition to serving as crucial climatic barriers, possible platforms for the collecting of renewable energy, providing indoor comfort with natural ventilation, view to outdoors and structural integrity, building envelopes (*Figure 2.1*) take part as important architectural elements. Historically, the majority of the time and attention in the design was concentrated on integrating passive design methods to minimize energy usage in accordance with an energy conservation strategy due to the poor performance of building envelopes and their restricted technical improvement (Wetter, 2004). Nevertheless, depending on the climatic setting, these measurements may not be representative of the final building performance and might offer a number of restrictions. In the most extreme circumstances, it may even result in circumstances where the functions of several building exterior components are in direct conflict with one another (Echenagucia, 2015).

## **2.2. NZEB FROM A VISUAL COMFORT POINT OF VIEW**

Access to natural illumination increases productivity and creates a healthier and more comfortable interior environment for building inhabitants. When constructed properly, daylighting may deliver high-quality light while greatly lowering the amount of energy needed for lighting and cooling. It can also limit solar heat absorption. In order to decrease building energy use and maximize sunshine dispersion, an efficient façade shading design should help to create this atmosphere (Lee, 2016). New high-performing building concepts, such zero-emission buildings (ZEB), have evolved towards holistic approaches to building design. In this regard, research on ZEBs supported the conclusion that improvements to the building envelope might greatly enhance building performance and help achieve greater sustainability goals (Boyce, 2003). Therefore, regarding the visual comfort leading to nZEB, it's important to balance the environmental advantages of increasing the area of external openings with the unfavorable effects based on the fact that every increase in the area of openings results in increased thermal gains inside the space and thermal discomfort, which then affects energy consumption and cooling loads, especially in a dry, hot climate (Mahdavinejad, 2011). Architectural designers frequently create models using software to assist them visualize various concepts and styles while planning the building envelope; altering one feature of the model normally necessitates low-level comprehensive alterations on many other areas (Reinhart, 2006). The exponential development in population throughout the world over the past 50 years has coincided with an increase in high density built environments, resulting in ever-diminishing natural, landscaped spaces in cities and ever-restricted lighting and views in buildings (Grimm, 2008). Financial constraints on real estate development frequently lead to taller, deeper floor plans with less access to daylight and views per square meter (Seto, 2017), while developers who have access to prime view properties frequently build up or out to maximize the view potential of their properties to the detriment of their neighbors. Energy efficiency norms and standards have become stricter as a result of concern over the energy and carbon implications of buildings (Dogrusoy, 2007). The regulation of lighting energy is influenced by a number of variables, but the building skin has a significant impact. Previously, architects frequently used their instincts and prior knowledge to create

the building's exterior. Environmental engineers will occasionally use mathematical analysis to justify their low-energy designs (Güneralp, 2017). However, later in the design process, their participation is frequently solicited. Consideration of an active design, which has the ability to increase a building's energy efficiency significantly more efficiently, can help passive design techniques be maximized during the early design phases (Ochoa, 2012). The key to sustainable building design in the city is reducing energy use for lighting reasons by employing daylight where there is a high demand for electricity. The energy used for electric lighting can be cut by 25–50% with the use of new light sources, design ideas, and controls, and by 75% with daylighting. Pressure will be put on the entire national energy demand if daylight penetration is not properly taken into account by the designer when determining the location and size of windows, interior surfaces, lighting fixtures, etc. In such a situation, it is essential to increase the use of natural light in urban office buildings in order to reduce reliance on artificial lighting and conserve energy, which is exceedingly scarce. Twelve international case studies have been discovered by the Centre for Building Performance and Diagnostics (CBPD) that show enhanced lighting design enhances individual productivity by 0.7 to 23% while lowering annual energy loads by 27-88%. As a result, one of the major problems that must be solved for sustainable office architecture is lighting (Loftness, 2006).

### **2.3. BUILDING PERFORMANCE SIMULATION TOOLS (BPSTs)**

Design professionals have increasingly relied on Building Performance Simulation (BPS) technologies to help them make judgments about how to create energy-efficient structures. Given the variety of BPS tools available, architects and engineers have a dilemma when deciding which ones to use in routine building design (Attia, 2012).

Programs for simulating building performance are effective instruments for researching visual, acoustic, thermal comfort, energy performance, etc, over the course of a building's life. There are several such tools available today, and they vary greatly in terms of their thermodynamic models, graphical user interfaces, purposes of usage, relevance to life cycles, and the capacity to share data with other software programs. The majority of simulation softwares include an "engine" that permits thorough several simulations based on straightforward text-based input and output files. According to the engine's underlying concept, these engines feature mathematical and thermodynamic methods that are utilized to determine their buildings' performance. The graphical user interface, which makes it easier to create input and analyze output and shows the user the capabilities of the engine, is crucial for the practical use of these tools (Maile, 2007). Building performance analysis are not, however, accessible to everyone due to user-friendly interfaces; in order to generate and comprehend realistic and trustworthy simulation results, one must be aware of the programs' limits and have a basic grasp of thermal processes. Additionally, the graphical user interfaces serve different purposes and typically do not make use of all the features of the associated engine. These tools are typically created to be utilized throughout the design stage of the life cycle of a structure (Bazjanac, 2007).

### **2.3.1. IMPORTANCE OF BPSTs**

As it incorporates human and computer interaction, simulation is a psychological and social science. Even though a thorough understanding of it enables advancement in the creation of simulations, the human dimension represents one of the most important performance indicators. The US government began programs in 1960 to assess the atmosphere in fallout shelters, which gave rise to this field of study (Fernandez-Antolin, 2020). New educational strategies that might equip professionals to create energy-efficient buildings are being included into curriculum with significant interest throughout Europe. Students believe that a crucial component of their architecture education should focus on sustainability. However, presently most courses on sustainability are lecture-based, with minimal emphasis placed on design studios. German researchers have looked into how well students can use and comprehend simulations. These results show that research in Europe is urgently required to apply BPSTs technology into the architectural sector (Vehmaa, 2018). Finally, Spain has an environmental component that is less than 10%. Due to the rigidity and existing structure of these institutions, it is rather challenging to incorporate environmental topics into the curricula of the Spanish Schools of Architecture (Gusc, 2018).

However, some methods and technologies can support a long-term paradigm shift in architectural idea and a teaching change through progressive progression. The new energy modeling tools are now being used by architects, but their usage is still quite limited. On the other hand, since experts struggle with both the interfaces and the analysis parameters, there has to be a shift in architectural education (Boarin, 2020). Moreover the importance of BPSTs regarding visual comfort is paving its way up to the top, as different outcomes of the results can impact the design. So the use of these tools provides an opportunity to simulate daylight and glare analysis and try multiple variations of the chosen design elements. This means that these tools allow architects and engineers to do parallel analysis while exploring the design process of a building (Hetherington, 2011).

### **2.3.2. FIVE MAJOR CHALLENGES**

There is currently no impartial assessment or classification of tool usability and functionality in respect to various user types' requirements. Rarely do tool creators disclose the tool's capabilities and restrictions. The choice of an appropriate tool from the expanding pool of BPS tools is a challenge to the potential user. Therefore, in order to simplify the selection process, we must clearly define the demands of various users in comparison to the capabilities and limits of the tool (Reinhart C. &, 2006).

Building simulation trends from few years ago are reviewed by Attia S. in a survey (Attia, 2012) where he presents key criteria for BPS tool selection and assessment based on an analysis of user requirements for tool capabilities. A literature study and two online questionnaires are used to conduct the research. The conclusions are based on a comparison between engineers and architects. To compare BPS theories and practices, five criteria are created. The recommended criteria are evaluated rigorously and put to the test based on the knowledge gathered from the survey. This research provides a set of

thorough selection criteria for BPS tools in an effort to solve this issue. A review of the body of literature revealed that the simulation community as a whole is debating at least five key subjects. These five criteria are categorized as follows:

- 1) Usability and Information Management (UIM) of interfaces;
- 2) Integration of Intelligent design Knowledge-Base (IIKB);
- 3) Accuracy of tools and Ability to simulate Detailed and Complex and building Components (AADCC);
- 4) Interoperability of Building Modelling (IBM);
- 5) Integration with Building Design Process (IBDP).

The five selection criteria are being presented with no intention of performing a thorough study. Instead, the goal is to identify broad yet crucial underlying assumptions and determine whether any are shared. There were two surveys done during the research. The first one was based on UIM (1), IIKB (2) and IBDP (5) criteria, and the second one based on AADCC (3) and IBM (4). Architects and engineers were the interviewed people, so the results are compared in multiple tables representing their answers.

### 2.3.3. USER SURVEY RESULTS

Several of the survey results are described in this section. Each survey had two sets of questions: eight for the introduction and other for the two selection criteria. Representative questions that illustrate the most significant findings were chosen to be mentioned in the user survey results comparison. Regarding the question “How do you describe your current position?” the respondents had a choice among 12 different groups, which included interns, teachers, graduate and undergraduate students, architects, engineers, and designers of either architecture or engineering. Surprisingly, a lot of construction management experts and students answered completely under the “Other” option. However, all categories were divided into the two main groups in order to perform the comparison between the groups of architects and engineers.

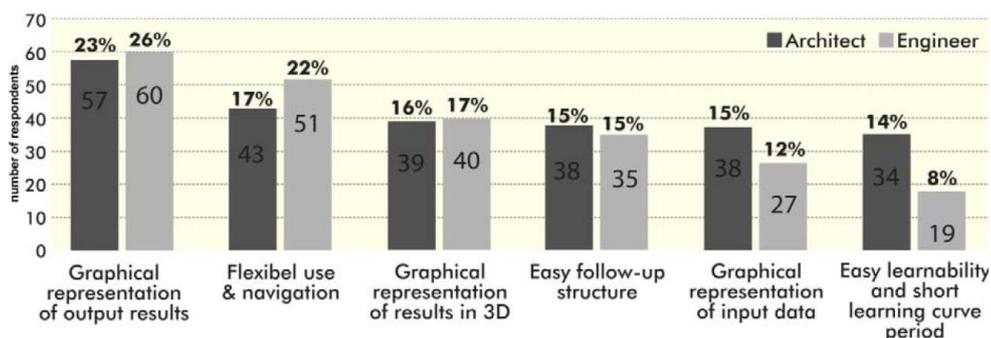


Figure 2.2 Ranking criteria concerning usability and graphical visualization of BPS interfaces

According to *Figure 2.2\**, 23% of architects and 26% of engineers concur that the graphical output results depiction is the most crucial component of the UIM of the interface. Both groups agreed that the

\* Note. Reprinted from “Selection criteria for building performance simulation tools: Contrasting architects’ and engineers’ needs Article in *Journal of Building Performance Simulation*”, by Attia S., Beltran L., LM Hensen J., De Herde A., (2011, January)

graphical depiction of the data in 3D spatial analysis is the most crucial element, followed by the flexible use and navigation. The ease of use and quick learning curve of simulation tools were deemed the least significant features by both groups.

Under “Integration of Intelligent Knowledge-base” engineers prefer the ability to assess the sensitivity and unpredictability of the design parameters, architects prioritize the ability to support decision-making. Three times as many architects as engineers elected to embrace overall design during the majority of design stages. Despite the fact that there are 249 architects overall compared to 232 engineers, the comparison is still fair and a noticeable difference can be seen. Also regarding the “Accuracy of the tools” sustainable design is the first preference (40%) of architects. The least essential factors were the capacity to calibrate uncertainty (8%) and the high resolution of the simulation model (6%).

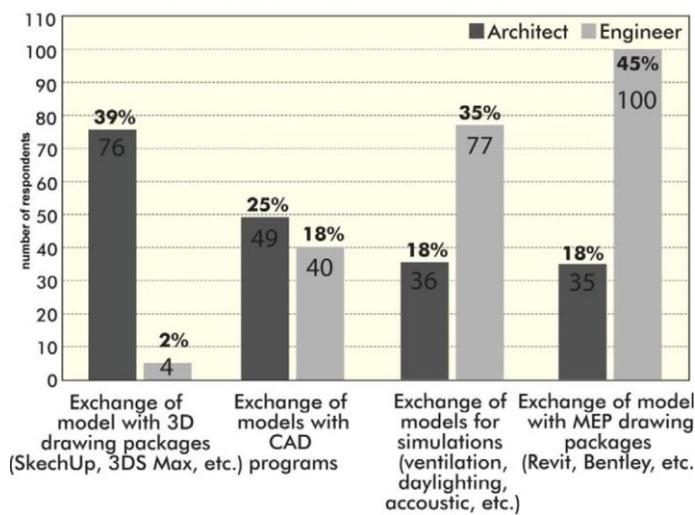


Figure 2.3 Interoperability sub criteria

Contrarily, the majority of engineers (31%) concurred that accurate and realistic findings are the most crucial aspect of tool accuracy. The least essential factors are the real-world sustainability of BPS tools (10%) and the high resolution of the simulation model (9%). “Interoperability of Building Model” was another evaluated topic, which is shown in *Figure 2.3\**. This shows what the priorities of the questioned architects and engineers were.

Architects ranked the ability to exchange models using 3D drawing programs like SketchUp and 3DS Max as their most crucial goal (39%). The second most popular option (25%) was model interchange with CAD software. With nearly no difference in preference (18.3% and 17.8%), the exchange of models for different simulation domains and the interchange of models with MEP (mechanical, electrical and plumbing) drawing packages came in last. Engineers, on the other hand, prioritized many sub-criteria. The ability to exchange models with MEP drawing tools like Revit and Bentley products was the most crucial sub-criteria (45%). The capacity to exchange models for various

Retrieved from: <https://www.researchgate.net/publication/254220572>

\* Note. Reprinted from “Selection criteria for building performance simulation tools: Contrasting architects’ and engineers’ needs Article in *Journal of Building Performance Simulation*”, by Attia S., Beltran L., LM Hensen J., De Herde A., (2011, January)

Retrieved from: <https://www.researchgate.net/publication/254220572>

simulation domains came in second (35%). Engineers chose the option to exchange models with CAD software in third place (18%). Less than 2% of respondents rated the capacity to model with 3D drawing software as the least critical characteristic.

Ranking the most important features of a simulation tool was placed at the end of the survey on purpose to ensure that respondents grasped the meaning of the four contrasted criteria with the least amount of ambiguity. The following was discovered after combining the responses from the two samples into the graph depicted in *Figure 2.4\**.

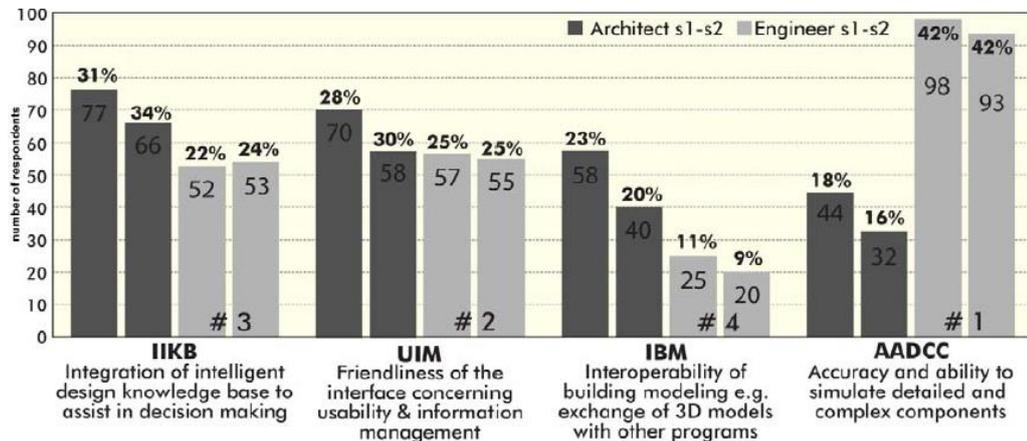


Figure 2.4 Ranking the most important features of a simulation tool

The IIKB was given precedence above the UIM of the interface and even the AADCC by architects in both surveys. The IBM was discovered to be the third crucial criterion for selection, with the AADCC coming in last. Engineers, on the other hand, decided to give AADCC priority. The IIKB finished third, followed by the UIM in second. The IBM was the least significant factor to consider while choosing a BPS tool.

The total number of votes each tool received is shown in *Figure 2.5\**. There is acceptance among architects that the top three are IES VE plug-in (85), ECOTECH (82) and DB (72). The 10 tools were ranked differently by engineers where DB (85) and EP (82) placed first in the group.

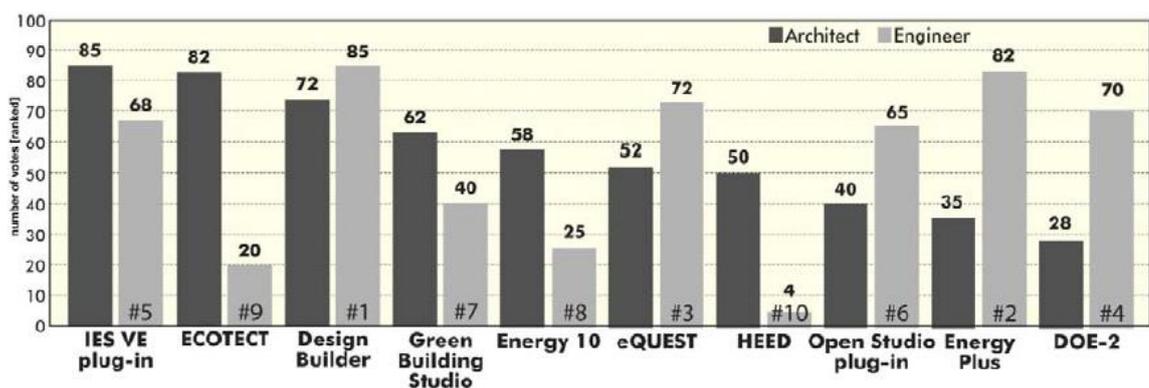


Figure 2.5 Ranking the ten tools according to architects and engineers

\* Note. Reprinted from "Selection criteria for building performance simulation tools: Contrasting architects' and engineers' needs Article in Journal of Building Performance Simulation", by Attia S., Beltran L., LM Hensen J., De Herde A., (2011, January) Retrieved from: <https://www.researchgate.net/publication/254220572>

Every section of the questionnaire was followed with an open question that invited feedback from the respondents. The feedback that was given was according to improvements in the future into BPSTs taking into consideration the five criteria mentioned in the beginning.

Fernandez-Antolin M. M. in another survey (Fernandez-Antolin M. M.-L., 2022) seeks to understand how architects feel about BPSTs. In order to do this, an online survey was carried out to ascertain the selection criteria of these BPSTs and non-users, as well as to look into the causes of their non-use.

There were 157 replies altogether, of which 61% were provided by architects, 26% by architecture students, and 13% by representatives of other disciplines. 54.8% of the workforce was employed in the residential sector, and 41.3% in the field of rehabilitation. 46.2% of those who worked on the typical architectural practice stages of design and construction. 53.7% of all respondents worked in interdisciplinary teams with other professionals, while 43.6% were hired by others. Among them, 29.5% were familiar with the design through computer simulation; this low number of users demonstrates that the majority of BPSTs are incompatible with the demands of architects [41, 42]. It was also clear that these tools were complicated for the profession. Young architects made up 76% of the respondents who were architects and had less than five years of professional experience. *Figure 2.6\** illustrates how confident this group of architects is in various BPST-related factors, which provides a number of justifications for closing the gap between the usage of BPSTs and architectural design.

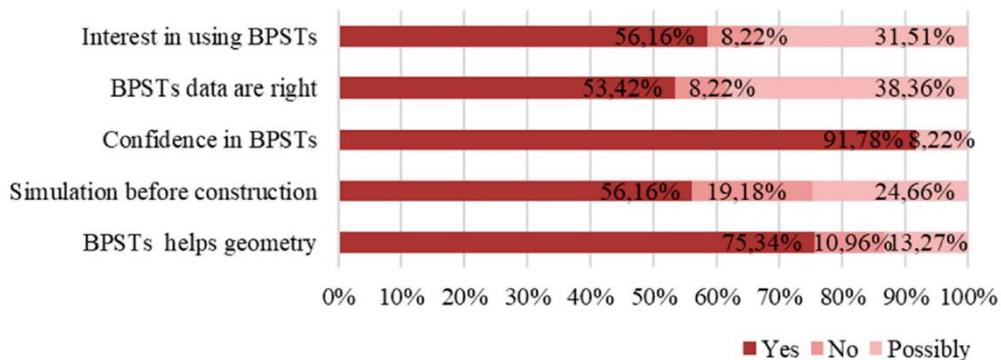


Figure 2.6 Confidence parameters of architects with professional experience below 5 years

Regarding their knowledge about BPSTs until that moment, the questioned people gave their opinion on “Fields in which simulation tools are useful” *Figure 2.7\**.

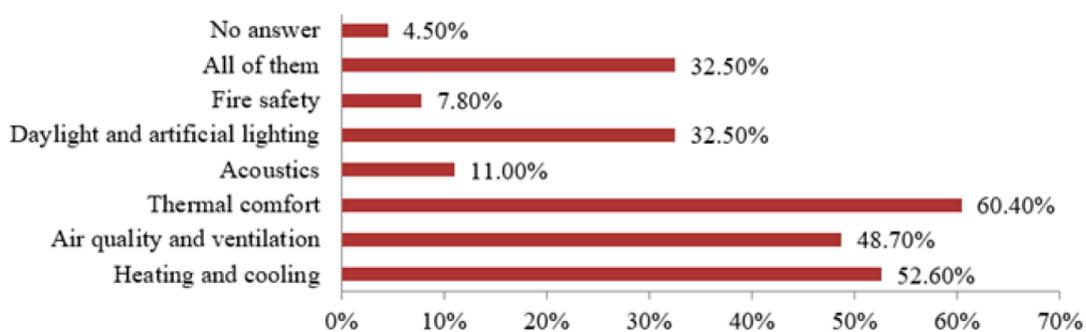


Figure 2.7 Fields in which simulation tools are useful

\* Note. Reprinted from “Building Performance Simulations and Architects against Climate Change and Energy Resource Scarcity”, by Fernandez-Antolin, M.-M., del Río, J.M., González-Lezcano, R.A., (2021, December) Retrieved from: <https://doi.org/10.3390/earth3010003>

Even though this research was made few years ago and it is accurate till today, few of the tools are not used anymore and there is an upgrade made to the ones operated by architects and engineers. The reason behind mentioning this research paper was to introduce the importance of BPSTs, their use since years ago and the development they went through in order to be adapted and adjusted to the needs of the users. Moreover it opens a door to the future where buildings will be designed by architects and engineers utilizing BPSTs, which are very accurate, adaptable, and capable of predicting some performances of the buildings at all design phases. To guide the design process and assess the effects of design choices, BPSTs can be at the center of creating high performance buildings. BPSTs provide a platform and a common ground for architects and engineers to work together. Most essential, promise to create sustainable structures.

## **2.4. INTRODUCTION TO BUILDING INFORMATION MODELING SOFTWARE PROGRAMS**

Building Information Modeling (BIM) is a current example of a significant paradigm shift in the construction industry. The key elements of this change are the representation of building components as data-rich 3D objects rather than as a combination of 2D orthogonal views and written documents, and the use of an interdisciplinary, comprehensive building model as the source for derived “views” rather than a collection of “drawings” that infers a 3D design. A BIM offers a centralized database of details about a structure, including its features and geometry. It is an integrated, comprehensive building model that includes extra 3D data and metadata in a distributed or centralized database together with information from traditional building papers including plans, specifications, and construction details (Ullah, 2019). BIM has been utilized in the construction industry over the past 20 years with the goal of enhancing stakeholder engagement, speeding up the development of new designs, and reducing inefficiencies on-site and between project disciplines. Once it is possible to increase the quality and performance of the construction life-cycle under the industry 4.0 paradigm, it has also evolved into the heart of the Cyber-physical system and a fantastic instrument to improve the energy efficiency concept in building design. BIM is now essential for bridging the chasm between digitization and the building industry. From the design phase to the administration of the building’s life cycle and maintenance, the digital information management capabilities provided by BIM enable process optimization and improved control (de Lange, 2017).

### **2.4.1. USE OF REVIT AS BIM SOFTWARE FOR DAYLIGHT ANALYSIS**

Autodesk offers free, renewable one-year educational access to its products and services for students and educators who qualify. One of them is Revit, used as BIM software program in this research. It is one of the most effective 3D modeling tools available today. The program, which has been developing

over the years, is able to use any kinds of complicated geometries because to its expanded side-tools. Revit was created exclusively for architectural use, that is, by architects for architects. The bulk of those responsible for its creation are either architects or have backgrounds in design and construction. The program enables users to create a building and all of its parts in 3D, add 2D drawing elements to the model, and retrieve building data from the building model's database. Revit is capable of 4D building information modeling and has capabilities for planning and tracking different phases of the building's lifespan, from idea to construction and subsequent maintenance and/or deconstruction (Aryanti, 2020).

Within Revit there is a possibility of performing different types of analysis on an existing buildings. Due to doing the research on daylight analysis, the recommended plug-ins for visual comfort analysis are strictly connected to performing daylight simulations. Hereby, Autodesk® Insight is introduced as most used one and a guide through the steps of using it is later explained (Merwe, 2019).

- *Autodesk® Insight*

Light levels throughout the design with the help of Insight Lighting Analysis with Revit, which offers automatic scheduling, in context daylighting, and lighting analysis results, could be performed. In addition to customized settings, the plugin offers automated configurations for particular research kinds. Based on the fundamental analysis engine, insight is divided into two parts: illumination and solar. Anything under "Lighting" is an illuminance-based study and uses the A360 rendering engine. The amount of light that strikes a surface is measured as illumination, which is in lux in the majority of other countries (based on square meters). Illuminance-based analyses' outcomes are influenced by the supplied sky model, glazing transparency, and how much light reflects off your model's interior surfaces.

Workflow of daylight analysis is elaborated in the following few steps:

1. Installing Insight Revit plugin and setting the project location.
2. Preparing the building elements and specifying floors and rooms as part of the model.
3. Set the reflectivity of opaque materials and valid glazing properties.
4. Click on the "Lighting" command and select "Run New Analysis".
5. Set the analysis type (seven options are available):
  - Illuminance analysis;
  - Daylight factor;
  - Daylight Autonomy (Spatial Daylight Autonomy (sDA) preview) analysis;
  - LEED 2009 IEQc8 opt1 (applicable spaces achieving daylight illuminance levels of a 108-5,400 lux in a clear sky condition);
  - LEED v4 EQc7 opt 1 (sDa+ Annual Sunlight Exposure (ASE));
  - LEED v4 EQc7 opt 2 (illuminance levels 300 lux - 3,000 lux in a clear sky condition);
  - Solar Access.
6. Before committing to a full (all level) study, it is advised to run just one level at a time to ensure that all of the settings are right and to keep costs down.
7. Run the analysis which will be on the cloud.
8. Results are displayed on the Revit model using the Analysis Visualization Framework (AVF), making it possible to precisely examine how lighting interacts with the model.

## **2.5. DAYLIGHT ANALYSIS SIMULATION TOOLS**

Daylight simulations give designers the chance to contrast and refine design options to enhance the occupants' visual and thermal comforts while utilizing energy-efficient methods. Overall, the complexity and scale of daylighting study objectives range widely, from straightforward examination of interior spaces to intricate façade fenestrations to urban environments. Point-In-Time Radiance Simulation is a metric used to assess lighting design and glare by examining precise light levels at a given day and time under localized sky circumstances. On the other hand Climate-based daylight modeling (CBDM) is the process of predicting any luminous quantity (illuminance and/or brightness) by applying realistic sun, sky conditions and climate data (Aryanti, 2020).

Numerous analysis tools have been created over time to investigate how well buildings utilize daylighting. These resources range from straightforward charts to cutting-edge computer applications. Literature can be used to find a brief historical evolution of these instruments (Ward, 1994). Over the past ten years, radiation has been thoroughly examined, also many other visual comfort analysis with a help of daylight simulation tools. It is quite important to mention that these tools are mainly used to study the daylighting performance of buildings (Rubinstein, 1988).

The ten daylight simulation programs used for this report are briefly described in this section. The description is based on data provided by the program developers and data found on relevant articles.

### **2.5.1. DESIGN BUILDER**

In 2011, Design Builder, a 2005 UK invention, incorporated daylighting features through Radiance and a GUI for Energy Plus HVAC systems. Design Builder V-5.2 Beta version of 2017 offered yearly climate-based daylighting using DAYSIM. To construct the geometry, Design Builder offers a built-in tool. Additionally, it can import .gbXML files. After some processing in Revit, models for architectural structures could be used for analyses of energy and daylighting. It takes more time to process the architectural Revit model before it can be exported as .gbXML. Both daylighting and energy simulation in DB might be done using this .gbXML model. Columns, window frame thicknesses, and interior ceilings should all be incorporated to the reduced model in order to simulate daylighting accurately. Additionally, the Energy Plus "Layer Method" might be used to produce unique construction materials. Users should set the Visible Absorptance of the innermost layer to the necessary value for daylighting applications.

The interior surfaces' visible reflectance is automatically calculated as  $1 - \text{visible absorption}$ . Users are unable to view the Visible Reflectance values in the Design Builder GUI, which is one limitation of the program. Both analysis grids and sensor placements are not selectable by users in Design Builder. Users only need to specify the appropriate node height above floor level, grid size, and grid offset from walls because annual daylight studies are automatically done for the entire building and on all grid nodes. In Design Builder, illumination maps for the building's floors are produced. However, neither

average lighting levels in each space nor floor areas below a particular threshold are documented. Illuminance data for analysis nodes cannot be accessed, which is another restriction. Additionally, plotting node values on the floor layout is not possible. For the entire building as well as for each individual space, color-coded work-planes with DA, UDI, sDA, and ASE results can be generated. The inability to plot node values on the work-plane is one restriction. For each building level, Design Builder creates a table with the names of the spaces, their areas, and the percentage of those areas that fall under the sDA and ASE thresholds. Additional research may be done in Excel, and spaces that don't fulfill the requirements could be easily detected. The inaccessibility of node values in Design Builder is one of its drawbacks. Node values would make it easier to locate weak places in spaces (*Figure 2.8\**) (Jakubiec, 2011).

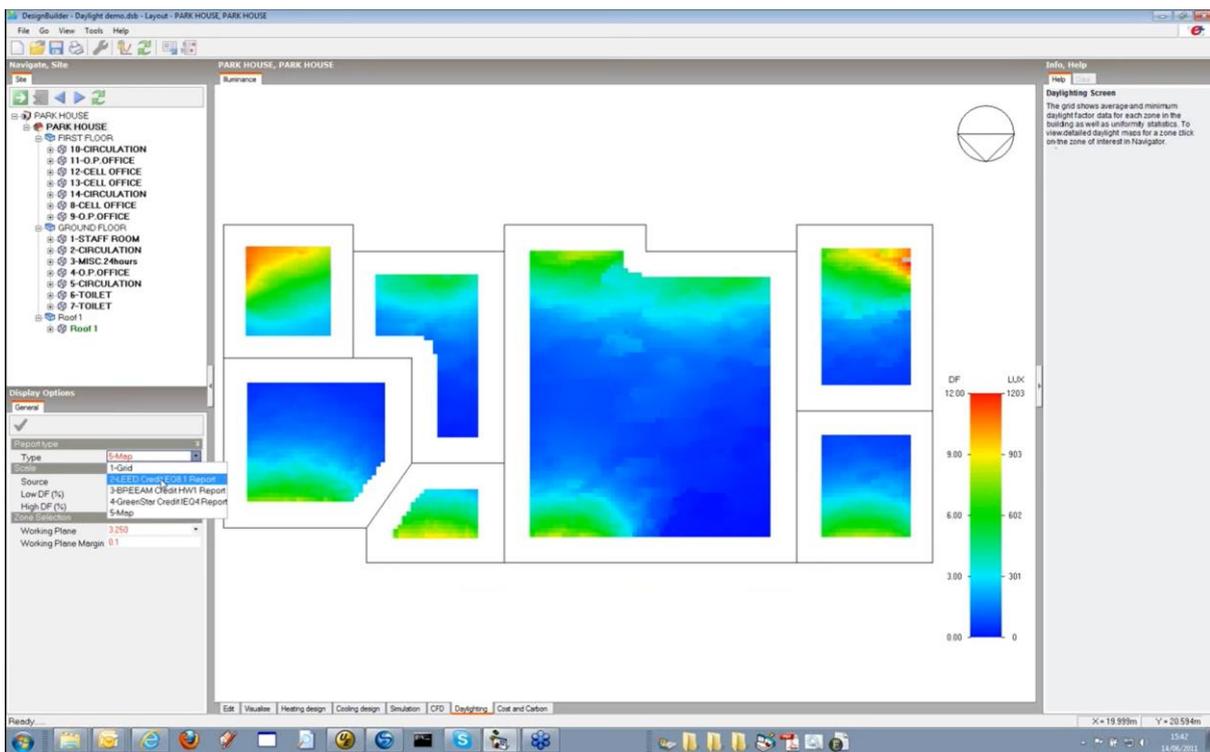


Figure 2.8 DesignBuilder interface, daylight analysis results

## 2.5.2. RADIANCE

Radiance is used to analyze and visualize lighting in designs. The scene geometry, materials, luminaires, time, date, and sky conditions are specified in the input files (for daylight calculations). Calculated values include spectral radiance (i.e., luminance + color), irradiance (illuminance + color), and glare indices. The output of simulations can be seen as colored visuals, numbers, and contour plots. The main benefit of Radiance over more straightforward lighting calculation and rendering tools is that there aren't many restrictions on the geometry or materials that can be simulated (*Table 2.1*). Radiance

\* Note. Reprinted from "DesignBuilder Detailed HVAC and Radiance Daylighting", by Dr. S. Potter and Dr. A. Tindale., (2013, May) Retrieved from: <https://www.youtube.com/watch?v=n6EkJ4kas4E>

is a tool used by researchers to assess new lighting and daylighting technologies as well as by architects and engineers to forecast illumination, visual quality, and look of creative design spaces (Mcneil, 2019).

<b>Modeling and Simulation Capabilities</b>	
Geometry	Planar polygons, planar rings, triangulated meshes, spheres, cylinders, cones. Spherical solid angle (for skies.)
Light model	Inverse square trichromatic with adaptive area subdivision
Color model	Radiometric RGB or (for specialized versions) XYZ. Radiance RGB primaries are close to, but predate, the sRGB primaries
Shaders	Ward empirical shader, procedural and data-based BRDF shader
Reflection and refraction	Mirrors, prismatic glazing, dielectric interfaces
Opaque and translucent materials	Parameterized by the Ward shader, based on measurable material properties
Transport materials	Extensive support for glazing. Thin glass and arbitrarily layered transparent materials are supported
Surface maps	Procedural and data-based visual textures ("patterns") and bump maps ("textures")
Mist, smoke, and clouds	Single-scattering.
Complex light sources	Pre-computed light distributions through large areas of glazing. Luminaire geometry can be "wrapped" with an invisible light-radiating surface to provide luminaires with accurate photometry.
<b>Input/Output</b>	
Geometry input	Textual; usually via conversion from CAD file format, but can be directly written or generated by scripts or provided geometry generators.
Materials input	Textual
Graphical output	Via RGBE HDR image files, the original HDR file format
<b>Simulation Methods</b>	
Rendering method	Hybrid deterministic/stochastic (Monte Carlo) ray tracing. Direct illumination and specular reflections are calculated deterministically. Indirect diffuse contributions and their gradients are calculated at a carefully-chosen subset of surface points, used to estimate neighboring values, and cached. Various other heuristics are used to accelerate the calculation.
Ray intersection finding algorithm	Octree spatial subdivision. Intersection-finding time has been measured to grow at slightly less than the fourth root of the number of surfaces in a scene.
Parallelization	Indirect diffuse contributions are accumulated and shared among local or distributed ray-tracing processes.

*Table 2.1 Radiance tool abilities*

### 2.5.3. DAYSIM

Modeling the annual amount of daylight in and around buildings is done using the proven, RADIANCE-based daylighting analysis software called DAYSIM. DAYSIM enables users to simulate dynamic facade systems, including switchable glazing, modern light redirection devices, and basic venetian blinds. The results of the simulations span from annual glare and electric lighting energy consumption to climate-based daylighting metrics like daylight autonomy and usable daylight illuminance. EnergyPlus, eQuest, and TRNSYS are just a few examples of thermal simulation engines that can be directly integrated with DAYSIM's hourly schedules for occupancy, electric lighting loads, and shading device status (Marsh, 2006).

## 2.5.4. OPEN - STUDIO

In order to allow advanced daylight analysis using Radiance and entire building energy modeling using Energy Plus, Open Studio® is a cross-platform (Windows, Mac, and Linux) set of software applications. To encourage community development, extension, and private sector acceptance, Open Studio is an open source (LGPL) project. The Open Studio Sketch Up Plug-in, Open Studio Application, and the Parametric Analysis Tool are graphical programs. The Open Studio Coalition, which was established to maintain and develop these graphical apps for the community of building energy modeling, is responsible for maintaining the Open Studio Application and the Sketch Up Plug-in. With

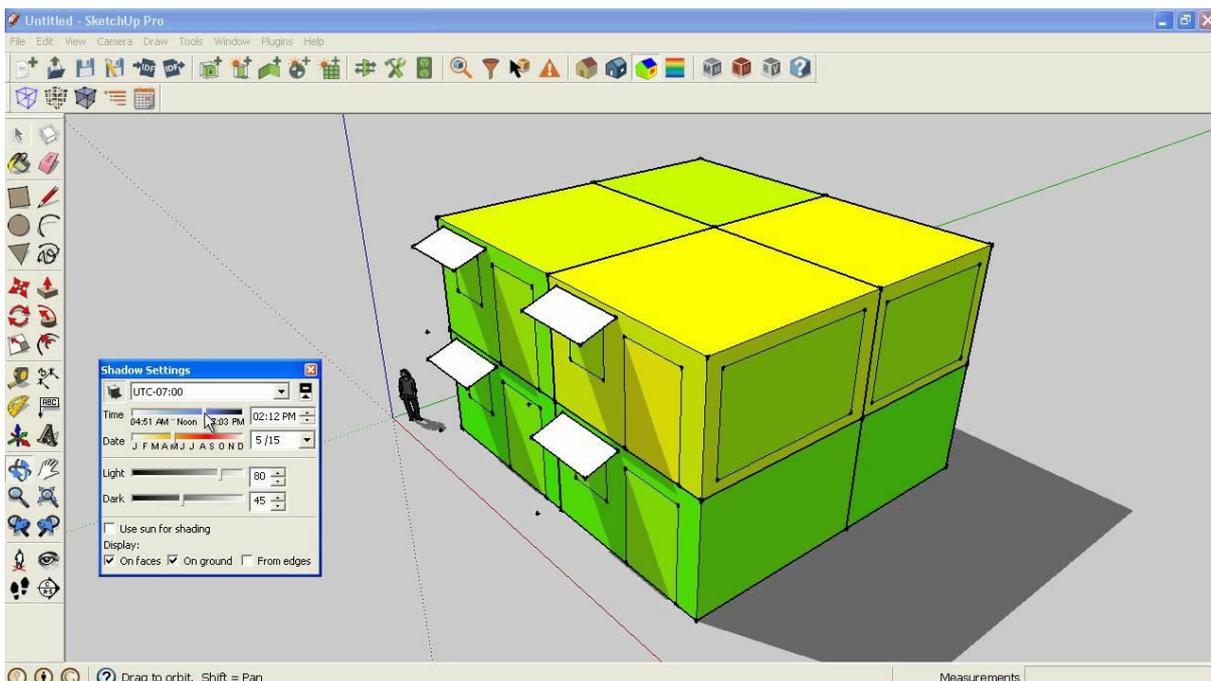


Figure 2.9 Open Studio as SketchUp plug-in

the help of the Sketch Up Plug-in (Figure 2.9\*), users of Trimble's well-liked Sketch Up 3D modeling program may rapidly produce the geometry required for Energy Plus. For the generation of geometry, Open Studio also allows the import of .gbXML and .IFC (Guglielmetti, 2011).

## 2.5.3. DIVA FOR RHINO

The Graduate School of Design at Harvard University initially created DIVA for Rhino between 2009 and 2011 for daylighting study. With the usage of the Radiance engine, DIVA was the first to incorporate DAYSIM for climate-based daylighting. For DIVA, Rhino should be used to construct the geometry. GbXML files cannot be loaded into DIVA, although Revit architectural models can be

---

\* Note. Reprinted from "Viewing Simulation Results in the OpenStudio SketchUp Plug-in and ResultsViewer", by NRELOpenStudio, (2010, December)  
Retrieved from: [https://www.youtube.com/watch?v=TI\\_uygHShL0](https://www.youtube.com/watch?v=TI_uygHShL0)

imported into Rhino. The only issue is that, in order to analyze glazing surfaces in DIVA, 3D surfaces must be flattened to 2D surfaces. Utilizing a script that flattens 3D windows into 2D surfaces, provided by the DIVA development team, solves this issue. Having two distinct models has the advantage that the energy model can be simplified and contain fewer features than the daylight model.

The second model's additional time to draw and the fact that the energy and daylight models are separate drawbacks. When architectural changes are made to one model, the other model is not immediately updated. The second model must therefore be created and updated more slowly as a result [95]. As graphical outputs after the daytime simulation is finished, DIVA automatically populates valuable results in addition to illuminance maps. For numerical outputs in DIVA, node values can be plotted on floor plans. Node values and simulation results are automatically saved in the project folder.

These data are presented as RGB values, which may be converted using a conversion formula into illuminance values at each node of the analysis grid. With DAYSIM, DIVA can simulate annual daylighting. In DIVA, LEED V.4- Option 1 metrics can also be produced. With DA, UDI, sDA, and ASE data, DIVA can produce color-coded floor plans and 3D renderings of spaces. Additionally, DIVA automatically generates a report following annual daylighting that contains hourly graphs of illumination schedules in a space with and without daylighting control. Mean Daylight Autonomy, Mean Annual Daylight Factor, sDA300/50%, Annual Occupancy Hours, and other numerical outputs are produced by DIVA (*Figure 2.10\**). Additionally, DIVA records DAYSIM results in .CSV file format automatically. The occupancy rate, illumination level, and percentage of electric lighting that is turned on are generated for each hour of the year (Jakubiec, 2011).

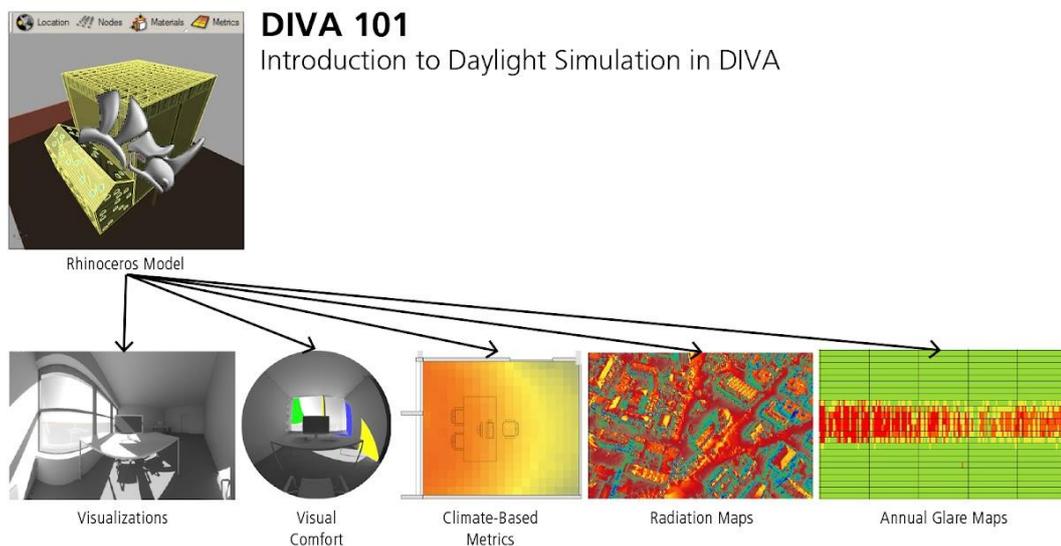


Figure 2.10 Introduction to Daylight Simulation in DIVA

Licenses for DIVA-for-Rhino are no longer available. The most recent update for DIVA, version 4, supports Rhino 6. A new and enhanced option is now available and is named ClimateStudio. The flagship program, Climate Studio, is compatible with Rhino versions 6 and 7, as well as upcoming Rhino/Grasshopper versions.

\* Note. Reprinted from "DIVA and HDR photography workshops at PLEA 2016 in LA", by S. McNeel, (2016, June)  
Retrieved from: <https://blog.rhino3d.com/2016/06/diva-and-hdr-photography-workshops-at.html>

## 2.5.4. CLIMATE STUDIO

The fastest and most precise software for analyzing environmental performance in the architecture, engineering, and construction (AEC) industry is called Climate Studio. Designers and consultants may optimize buildings using its simulation workflows for energy efficiency, daylight access, electric lighting performance, visual and thermal comfort, and other occupant health metrics. Thousands of components, structures, and templates from verified sources and actual measurements are included in Climate Studio. The projects created by Climate Studio are always grounded in reality, whether they use DOE benchmarks, ASHRAE standards, or glazing components from the IGDB. Climate Studio offers interactive representations of local temperature, humidity, wind, radiation, and UTCI variables in addition to a searchable library of over 30,000 weather files. To emphasize passive design potential and site impact, visualize sun angles and shadows. sDA is calculated correctly by Climate Studio utilizing dynamic shading and the most recent LM-83 criteria. With cutting-edge speeds and automated reporting, you may quickly transition from noncompliance to compliance. In a matter of seconds, you may quickly evaluate glare from hundreds of viewing angles for every hour of the year. With automatic reporting, adjustable view factors, and LEED v4 Quality Views calculations, Climate Studio can calculate any combination of scene elements. Apply real-world shades and dynamic glazing products to your design (Figure 2.11\*).

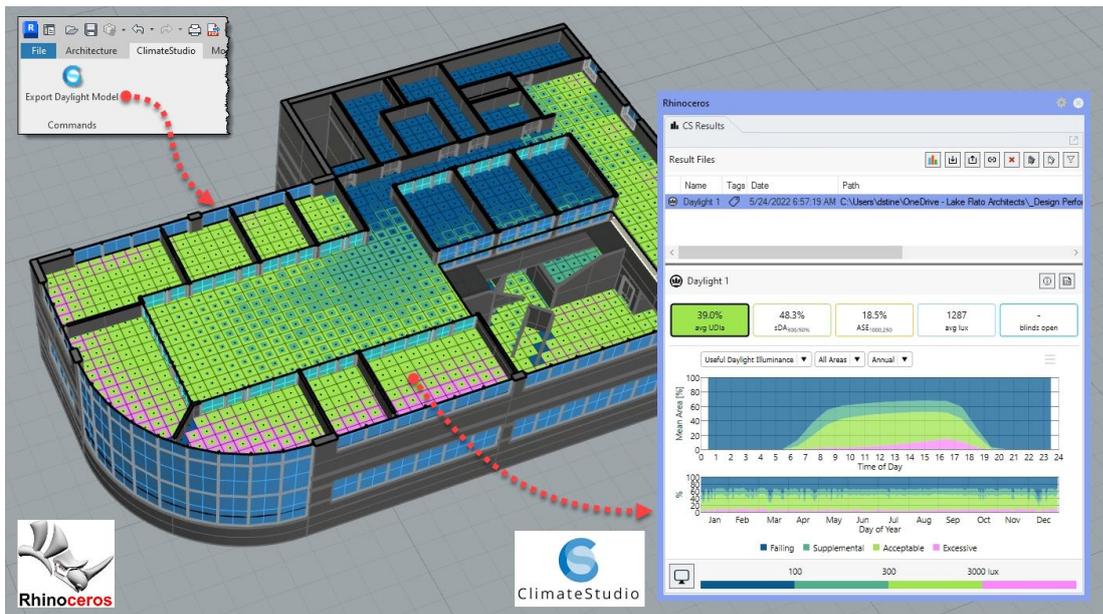


Figure 2.11 Climate Studio Rhino Plug-in

Climate Studio computes and simulates operation schedules automatically, based on LM-83 or custom schema. Placing and editing luminaires, calculating illuminance levels, making visualizations, and figuring out lighting power loads are all made simple. Grasshopper-based fully parametric daylight and energy modeling is available in Climate Studio. There has never been a simpler time for optimizations, sensitivity analysis, and automated model building (ClimateStudio, 2022).

\* Note. Reprinted from "Climate Studio Revit Plug-in", by D. Stine, (2022, May)  
Retrieved from: <https://bimchapters.blogspot.com/2022/05/climate-studio-revit-plug-in.html>

## 2.5.5. VELUX DAYLIGHT VISUALIZER

For the analysis of daylight conditions in buildings, VELUX Daylight Visualizer is a qualified and verified simulation tool. Its goals are to encourage the use of daylight in structures and to help professionals by foreseeing and recording daylight levels and the appearance of a space before the building design is implemented. The program's straightforward user interface makes it accessible, efficient, and simple to use (Ghofrani, 2020).

Most important key features of VELUX Daylight Visualizer are:

- Any type of project, including small, medium and large scale residential, commercial, and industrial projects, can utilize Daylight Visualizer to assess the daylight conditions.
- Using false color and photorealistic photos, see and calculate the distribution and amount of daylight (luminance, illuminance, and daylight factor) in structures.
- Daylight factor calculations
- The output of the simulation can be a report that displays the daylight performance by zone for each space/room in the building. For each zone, the results contain the average, median, lowest, maximum, and uniformity values.
- Create 3D models using the embedded modeling tool so that windows on the facade and roof can be added at will. Alternatively, you can directly import 3D models from your CAD program (such as AutoCAD, Revit, Sketch Up, Archicad, and others) using one of the 3D file types DWG, DXF, SKP, or OBJ.
- A fast and accurate daylighting simulation tool, Daylight Visualizer is built on cutting-edge rendering technology and is capable of replicating the intricate character of daylight in interior building spaces.

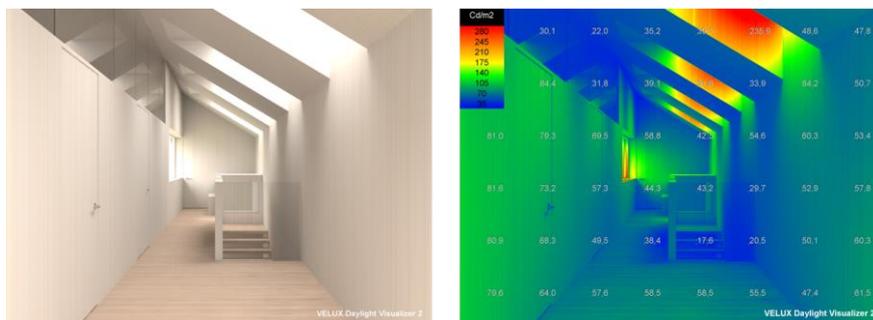


Figure 2.12 Luminance rendering of Sunlight House shown with photo-realistic and false colour images.

In Daylight Visualizer, bidirectional Monte Carlo ray tracing with photon mapping is used to run simulations. This biased two pass global illumination simulation technique. Photons are tracked from light sources in the initial pass, and the hit points they produce on non-specular surfaces are saved as a photon map (Figure 2.12\*). Rays are followed from the eye or camera in the second pass until they encounter a non-specular surface. Then, using density estimation, the radiance value at the camera ray hit site is determined (Jensen, 2001).

\* Note. Reprinted from "1.8 Daylight simulation tools", by VELUX Group, (2022)

Retrieved from: <https://www.velux.com/what-we-do/research-and-knowledge/deic-basic-book/daylight/daylight-simulation-tools>

## 2.5.6. DIALUX EVO

DIALux Evo is using three different sky models to analyze daylight: overcast, average and clear sky. Only when the sky is cloudy can one calculate the daylight quotient necessary by some standards. When a daylight calculation is made, DIALux Evo automatically provides a matched light scenario. Only a clear or average sky allows for the calculation of direct sunlight. The illuminance calculation is only performed inside the building and only through windows that let natural light into the interior. The daylight calculation does not account for other building openings. Inside-the-building windows are not useful for determining daylight. DIALux typically recognizes this on its own. This parameter can be adjusted if necessary at the apertures tool's window. There are no specific settings needed for the documentation of daylight situations. It simply includes information about location, date, time, sky model, and zenith brightness in addition to the photometric data. In the summary, a daylight quotient is always provided (Knowledge Base DIALux Evo, 2021).

## 2.5.7. ALFA

In order to design settings that are safer, healthier, and more productive, architects, lighting designers, and health experts may now forecast and regulate these non-visual impacts using a new software called ALFA, or Adaptive Lighting for Alertness. The most recent service release of version 6 or 7 is required in order to use the ALFA Rhinoceros 3D plugin. The non-visual system's activity cannot be characterized by conventional Red/Green/Blue color channels since it is sensitive to certain blue light wavelengths. ALFA has expanded the Radiance lighting engine to depict the world in high-resolution, 81-color spectrum in order to get around this restriction. ALFA offers an interactive, 360-degree depiction for every view position (*Figure 2.13\**). The camera can be turned to look at the shift in the light's spectrum and the arrival of melanopic lux. To observe the spectral power distribution emanating from any area of the scene, over individual pixels can be hovered (LLC, 2022).

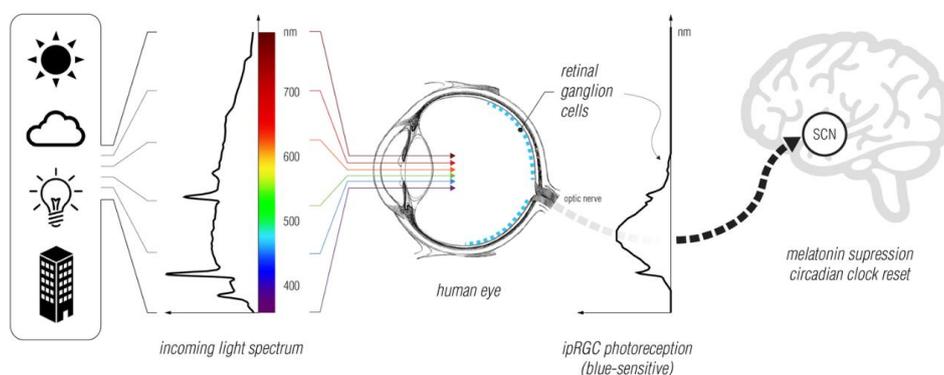


Figure 2.13 Representation of light reception by human's eye by ALFA

\* Note. Reprinted from "Adaptive Lighting for Alertness - A new circadian lighting design software", by Solemma LLC., (2022)  
Retrieved from: <https://www.solemma.com/alfa>

## **2.6. INTEROPERABILITY BETWEEN BIM AND DAYLIGHT ANALYSIS SIMULATION TOOLS**

The fundamental term to better comprehend the mechanisms underlying BIM, such as design and modeling methodology, is undoubtedly “interoperability,” which is understood as the potential for information and data to be exchanged between the various applications in a completely automatic and lossless manner to allow each professional to continue using a specific piece of software, provided that it shares a common exchange format that is accessible to all from a single database.

It is important to remember that this sharing opportunity applies to the whole life cycle of the project, including subsequent maintenance and final disposal, and not just the construction period. In addition, it is important to stress the role that interoperability plays in connecting BIM as both a model and a modeling approach. It is clear from this that BIM provides this option for managing digital information, which has positive effects on design-related costs and efficiency.

Finally, it can be said that even though different design tools have been adopted by individuals for the development of every stage of the construction process, it is still possible to exchange useful information today using a variety of standard formats that can automatically preserve the original data with little risk of loss (Akbarieh, 2018).

However, one of the current digital gaps at the design stage is BIM and Daylight Analysis Simulation Tools interoperability. Although interoperability problems still preclude many BIM applications in pertinent areas, BIM is still regarded as a multidisciplinary tool. Data loss and misinterpretation are frequently the result of incompatibility between the BIM model and the daylight simulation applications, repetitive manual tasks needed to create a model prepared for Daylight Analysis Simulation Tools, and the non-standardized and subjective process, especially when this was taken into account in the first stage of the design step (Pineiro, 2018).

### **2.6.1. BIM DATA CONVERSION MODE AND DATA LOSS**

IFC and gbXML are two open exchange formats that are currently used for information sharing between BIM authoring tools and Daylight Analysis Simulation Tools. Both offer ways to store geometry with associated data, however the BIM tool frequently cannot export this data accurately, while the Daylight Analysis Simulation Tools frequently cannot comprehend it.

In BIM technology, data conversion often takes one of the following three forms (Hijazi, 2015):

- *Direct conversion*

Most performance analysis software programs have modeling capabilities and then do simulation analysis right away. But complex modeling cannot be established, nor is its modeling function simple to use. However, heterogeneous or spatially complicated sectors are typically those that require performance analysis.

- *Intermediate format conversion*

Although gbXML format is already the default data input format for the majority of performance-based analysis software programs, BIM models frequently export gbXML format before importing it into analysis software. The sent data cannot be successfully filtered and simplified due to the jumble of model data, which causes errors in model identification, wall fragmentation, and other issues that need to be fixed. The dilemma gets increasingly serious as the BIM model's complexity rises. Consequently, although being connected to the simulation software by this data interface, it has not been utilized to its full potential. Traditionally, models have been created and imported using the simulation software's built-in model or other techniques, which adds to the labor of designers and inefficiently utilizes the refined BIM model data.

- *Common standard conversion*

IFC is indeed an open building engineering data standard. According to theory, the generated IFC data can be exchanged and communicated regardless of which BIM program was used to create the model. According to the literature, IFC standard development and related assessment in the field of BIM performance analysis believe that, in theory, IFC standard essentially satisfies the data conversion needs of design. However, in actual use, all major software programs use their own databases to align with their platforms when they exchange IFC files with one another. It is inevitable that the information is true or false when IFC files are input and produced because the majority of databases are not created strictly in line with the IFC standard format. Therefore, more work needs to be done before the IFC-based data sharing can be applied to engineering practice.

Bastos Porsani, G. sought to determine whether there is an automatic workflow between the software used for daylight simulation analysis and building information modeling (BIM), as well as how it operates (Bastos Porsani, 2021). The plan was to test three distinct BIM to Daylight Analysis Simulation Tools flows using the same Energy Plus calculation engine in the Daylight Analysis Simulation software and the same BIM authoring tool. Design Builder, Open Studio, and CYPETHERM HE were the three BEM tools chosen for this study, and 3D models were created in Autodesk Revit 2020.

The conclusion mentioned in the same paper was stated mainly on data loss and interoperability errors between the multiple software programs. One of the current barriers between the construction industry and digitalization is the absence of BIM-Daylight Analysis Simulation Tools interoperability.

A semi-automated workflow between BIM and Daylight Analysis Simulation Tools has been seen to exist. Prior to exporting the gbXML and IFC files, Revit requires that a few parameters be turned on. Secondly, the Daylight Analysis Simulation tool needs to be properly configured for the model schema data input, taking into account the building typology, ideal loads, occupation schedule, and weather information. Their configuration, however, was insufficient to guarantee good compatibility, as many of the issues with the Daylight Analysis Simulation software programs were the result of mistakes made when transferring the data from the BIM authoring tool to the gbXML and IFC files. As they supplied thermal properties and geometry values that were different from the baseline model, the energy models

developed with gbXML and IFC cannot be relied upon as a reference for the BIM. This led to diverse and incorrect simulation results (Figure 2.13\*).

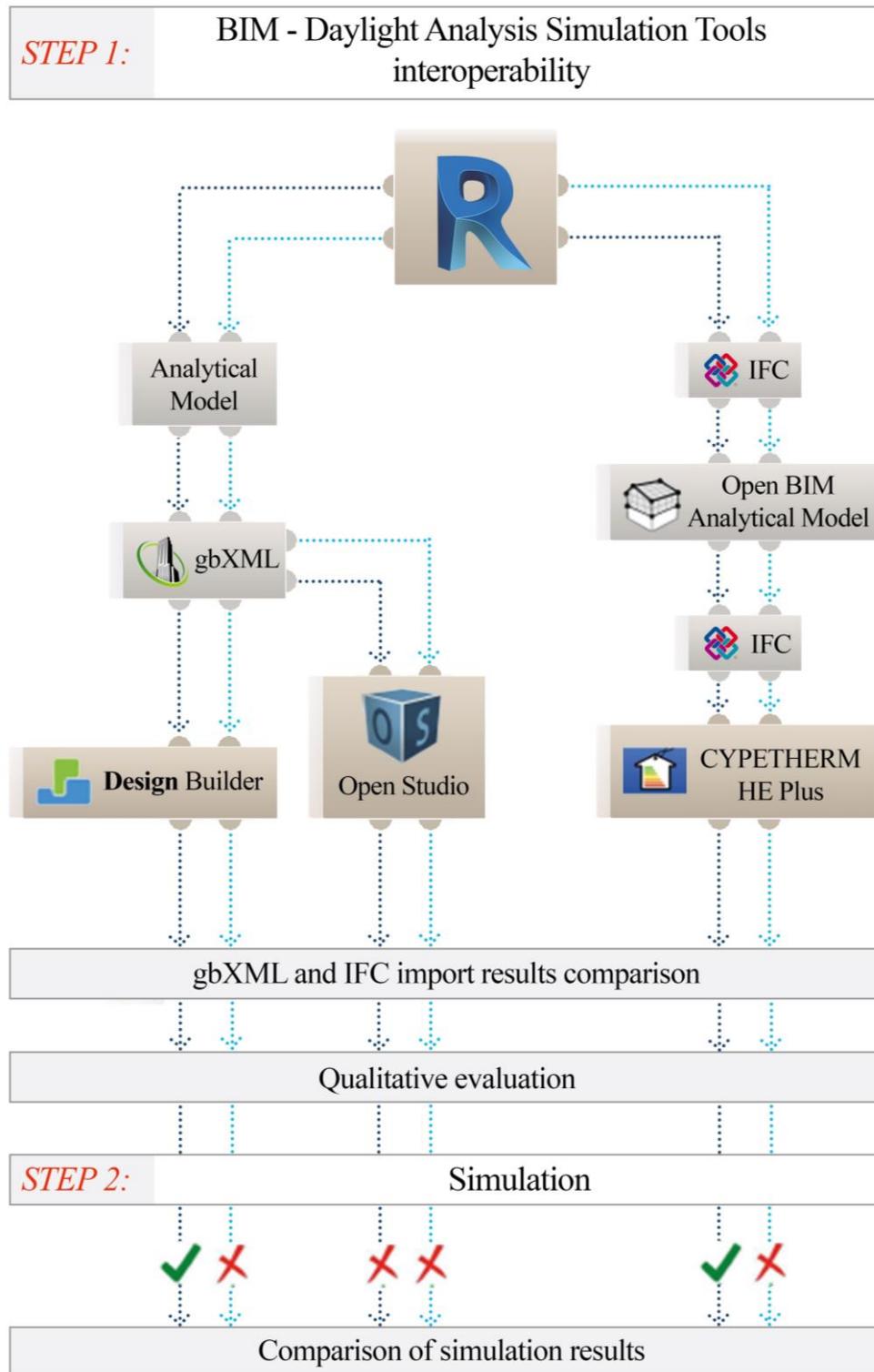


Figure 2.14 Interoperability and data loss from BIM to BEM using .gbXML and .IFC

\* Note. Adopted from "Interoperability between Building Information Modelling (BIM) and Building Energy Model (BEM)", by 60. Bastos Porsani, G.; Del Valle de Lersundi, K.; Sánchez-Ostiz Gutiérrez, A.; Fernández Bandera, C., (2021, November) Retrieved from: <https://doi.org/10.3390/app11052167>

## 2.7. VISUAL PROGRAMING LANGUAGE SOFTWARE PROGRAMS

The Visual Programing Language (VPL) is an approach used by designers to alter project parameters in order to optimize component and design selections targeted at particular goals. When using visual programming, users interact with program parts graphically rather than by typing lines of text code. Grasshopper was used in the current work. Numbers, sliders, operators, and functions, list manipulators, graphic designers, scripts, notes, customizable nodes, and nodes for other developers (such optimization components), among other constructs, are built in a visual programming environment (Asl, 2014).

Graphics (such blocks and wires) are used in visual programming languages to build software. Essentially, the program is the flow diagram. Visual programming languages include Simulink, Grasshopper, Dynamo, Marionette, and others. Their user interfaces differ significantly from those of text-based programming languages, such as BASIC, LISP, C#, Python, Java, and Processing, where the code is entered into a text editor or a more complex code editing program (Celani, 2012).

### 2.7.1. POSSIBLE WORKFLOWS FROM BIM TO VPL

There are multiple workflows that can be followed while transmitting the data from BIM software to a VPL software. As mentioned previously for the purpose of this research Revit was used as a BIM software. Exploring the choices and which workflow is best suiting, few workflows came across, which allow almost smooth interoperability. Unfortunately not all of them are tested and used on a daily basis. Some of them, due to their complexity, are still less known or under development.

- Revit ↔\* Dynamo (Ladybug, Honeybee, Energy Analysis for Dynamo);
- Revit →\*\* Export as .DXF/.DWG → Import to Rhino ↔ Grasshopper (to analyze all components need to be specified);
- Revit ↔ Rhino Inside Revit (RIR) ↔ Grasshopper (Honeybee, Ladybug, Dragonfly, Bombyx, Pachiderm, OneClickLCA);
- Revit ↔ Revit Connector → Import to Speckle Stream → Open Speckle Web Account Copy the link → Import as link connected to Grasshopper “Receive” node from Speckle Connector → Define objects attributes → further analysis on Grasshopper (Honeybee, Ladybug, Dragonfly, Bombyx, Pachiderm, OneClickLCA).

The above mentioned are the most used and most accurate workflows. Looking through their pros and cons and following few researches and articles, where the interoperability was tested, using Rhino Inside Revit was introduced as most smoothest workflow. Of course this is evaluated in Chapter 3, where the capabilities of Rhino Inside Revit are tested on the chosen case studies.

---

\* Use “↔” for both ways interoperability, transmitting data from one to another software and backward

\*\* Use “→” for one way interoperability, transmitting data from the first software to the following software

## 2.7.2. APPLICATION OF VPL THROUGH DYNAMO

For software or other applications that operate in a virtual world, such as Revit or Maya, Dynamo is software that can be utilized as a plugin. Programmer and non-programmer alike can use the software. Through the use of various text-based programming languages, users can design logic and behavioral scenarios visually. After the software has been installed, Dynamo enables the user to operate within a programming process that combines elements to decide the order and relationships of the activities that make up the algorithms. The algorithms work perfectly in real time for a variety of tasks, including data transfer and geometry construction (Alisherbek, 2021).

Although Dynamo is an adaptable environment made for a variety of applications, it is actually made to use Revit. For the Building Information Model (BIM), the visual application generates reliable options. Along with third-party libraries created by the advanced AEC team, Dynamo offers a choice of websites created especially for Revit. The algorithmic data model is extended by the Dynamo graphical algorithm editor for Revit employing data environment and logic.

BIM has new possibilities thanks to Revit's adaptability and trustworthy database. For each project, Revit builds a trustworthy database, but it might be challenging for the typical user to get that data via the user interface. A thorough API (Application Program Interface) is provided by Revit so that outside developers can make their own tools. Data democratization is a goal of Dynamo Revit's graphical algorithm editor (*Figure 2.15\**) (Yerjanovich, 2021).

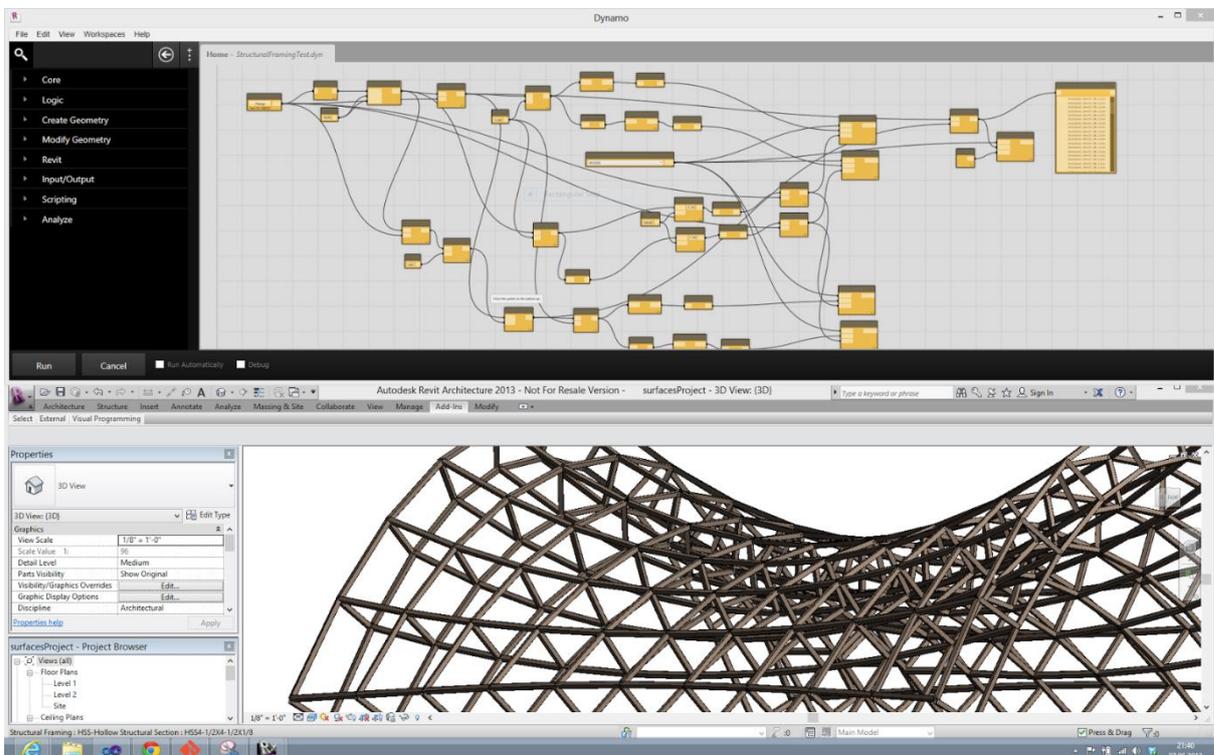


Figure 2.15 Execution of algorithms when working with Dynamo Revit

\* Note. Adopted from "Improving BIM workflow with dynamo revit plugin", (2022, May)

Retrieved from: <https://constructionreviewonline.com/installations-materials/improving-bim-workflow-with-dynamo-revit-plugin/>

Dynamo is a crucial component of many entertaining applications, and the project is set up to be used in the creation of unique workflow software solutions. Software platform, flexible and extensive design tool: Dynamo Visualization, which can be utilized to create a variety of creative workflows whether it functions as a stand-alone software or in addition to other design tools. The idea of separating lexical data type from list data type is provided by Dynamo 2.0. The way you create data for your process and how you work may alter significantly as a result of this shift. Dictionary and list data types were unified in version 2.0. The lists, in essence, were dictionaries with keys. An example of a form of data that consists of key-value pairs with a unique key for each set is a dictionary. With a key instead of an index value like the one in the list, you it can be essentially "searched for things" as there are no orders in the dictionary. Switches in Dynamo 2.0 can only be strings.

The views of the model can be divided into several sheets either directly from the designed work or from an Excel file. With Dynamo, a huge variety of jobs may be automated. On the plus side, any task that must be finished more than five times per day and takes more than a minute can be automated using Dynamo. Renaming the order of pages, inserting a group of families, editing multiple sheets, and modifying the text box are additional capabilities offered by Dynamo.

In addition to task automation, Dynamo is a potent design tool for data exporting and importing. Designers can make the design developments they produce into an actual, precisely defined process using Dynamo. However, altering structures or intricate geometries is not the only thing that can be done using Dynamo as a design tool. It makes sense to automate the process of working with dull items so that users can focus on developing the project's many components to make it more fascinating while avoiding having to work with repeated building components.

At every level of the design process, dynamo assists in modeling the performance of structures. Real data cannot be replaced by simulation data, but it does enable for the evaluation of projects using objective standards. You can find the best answer by figuring out which classification works the best. This choice can be made not just after construction but also throughout the design phase thanks to computational design tools like Dynamo. With Dynamo, the user must approach the task methodically and logically at each stage (Qudratovich, 2021).

### **2.7.3. APPLICATION OF VPL THROUGH GRASSHOPPER**

Visual user interfaces for programming, which enable users to construct personalized, adaptable, and potent form-generating algorithms without having to first learn how to write code, have become increasingly popular recently. Instead of inputting lines of text code, users interact with program elements graphically when using visual programming. With Grasshopper which is part of Rhino, nodes are generated; depending on the tool, they can be of many types, including numbers, sliders for modifying values, operators and functions, list manipulation tools, graphic producers, scripts, notes, "watch" nodes, and customized nodes. The program is resolved from left to right, and they are practically hooked together (Jabi, 2014).

## 2.7.4. RHINO INSIDE REVIT

Rhino may now be integrated into other 64bit Windows apps thanks to a new technology called Rhino Inside. Based on this technology, Rhino Inside Revit offers a platform for an unheard of degree of connectivity between Rhino and Revit. Technically speaking, it is a Revit add-on that, like other Revit add-ons, loads Rhino and its plugins (such as Grasshopper) into the memory of Revit. In addition to bringing a number of new components for communicating with Revit, Grasshopper also makes the two software APIs (Application Programming Interfaces) accessible through its script components. Practically speaking, Rhino Inside Revit offers a set of tools for linking the Grasshopper ecosystem's extensive ecology to the Revit environment, opening up entirely new design workflows and possibilities.

Rhino Inside Revit extends Grasshopper with more than 300 Revit-aware components that can query, modify, analyze, and produce native Revit objects. With each new iteration, more components are added to enable more native Revit types.

Rhino Inside Revit is a true game-changer since it makes the information transmission between Rhinoceros 3D and Autodesk Revit rapid, simple, and intuitive. The ability to use Rhino and Grasshopper inside of programs like Revit is made possible by this open-source initiative. The geometry may be transferred from Rhino to Revit with the simple touch of a button, and vice versa. Rhino Inside Revit's simplicity is its strength; anyone can learn to use it without needing to be an expert in computational design (Aksamija, 2013).

## 2.7.5. GRASSHOPPER INSIDE REVIT

Grasshopper is generally used to create generative algorithms, such as those for generative art. A large number of Grasshopper's components produce 3D geometry. Other sorts of algorithms, such as those for text, audio, video, and haptic applications, may also be present. Parametric modeling for structural engineering, parametric modeling for design and fabrication, lighting performance analysis for environmentally friendly architecture, and building energy consumption are some of the most sophisticated uses of Grasshopper (Tedeschi, 2011).

Grasshopper is possibly one of Rhino Inside Revit's most significant features. Firstly, Rhino Inside Revit will start loading once you click the Start button, which is under the Rhino Inside Revit tab in Revit UI (User Interface). The add-on makes an effort to load Rhino 7 into Revit's memory and verify that it is authorized. New Rhinoceros and Grasshopper panels will be added to the toolbar after the loading process is finished. To access it there is a button on the Rhinoceros tab in Revit (*Figure 2.16*).

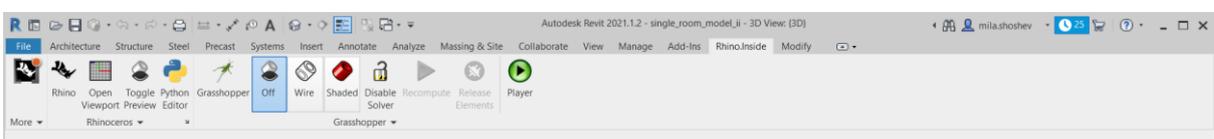


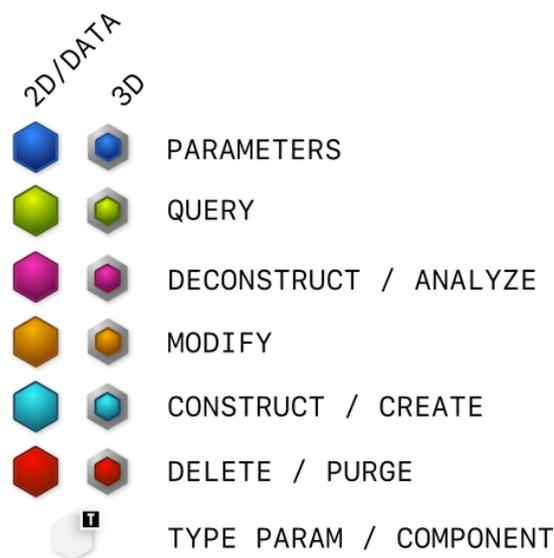
Figure 2.16 Screenshot from Revit UI accessing Rhino with RhinoInside

After Grasshopper is launched through Revit, its UI is opened and there is an extra Revit tab in Grasshopper window, with a lot of Revit-aware components (*Figure 2.17*).



*Figure 2.17 Screenshot from Grasshopper UI accessing Revit-aware components*

The icons for Revit-aware components aid in recognizing the function of the component. The base color, as seen on *Figure 2.18\**, indicates the form of activity (Query, Analyze, Modify, Create). A number of badges that are attached to icons also indicate the Type, Identity, or other characteristics of the data that the component is intended to operate with.



*Figure 2.18 Revit-aware components icons*

The interactivity of its components is arguably the most significant aspect of a visual programming environment like Grasshopper. The Rhino Inside Revit project adds this interaction to the Revit environment, which makes it much easier for designers to explore the design area and come up with fresh solutions. All Rhino and Grasshopper add-ons may be able to access the Revit API because Rhino is executing within the memory of Revit. Since the python scripts may access the Rhino API, Grasshopper API, and Revit API simultaneously thanks to this capability, the Python programming in Rhino and Grasshopper becomes considerably more powerful.

\* Note. Retrieved from: <https://www.rhino3d.com/inside/revit/beta/getting-started#grasshopper-inside-revit>

## 2.7.6. GRASSHOPPER PLUGINS LADYBUG TOOLS

Multiple analysis can be done using the tools that can be downloaded and accessed as plugins through Grasshopper. They provide various components that can evaluate one object's states and visualize the results, which later can be compared. All of these analysis are done in order to gather results for visual comfort, thermal comfort, acoustic comfort, air tightness and many more, providing further development of the initial design of, for example, an existing building. These tools allow to perform a smoother interoperability between Revit and Grasshopper, particularly using Rhino Inside Revit. Using the data transmission availability, changes can be made in the design during the design stages, while the Revit UI is opened. All them may be applied as dimensional, geometry, material or technology changes (Ladybug Tools, 2020).

- *Ladybug*

In Grasshopper, Ladybug enables the analysis and visualization of weather data. Diagrams such as the sun route, wind rose, psychometric chart, etc. are included in this, as well as geometric studies like radiation analysis, shadow studies, and view analysis. Grasshopper can import common Energy Plus Weather files (.EPW) through Ladybug. It offers a selection of interactive 2D and 3D climatic visualizations to aid in decision-making during the preliminary design stages. Through solar radiation research, view assessments, sunlight-hours modeling, and other methods, Ladybug additionally aids in the assessment of preliminary design ideas. Integration with visual programming environments enables high levels of customization and immediate feedback on design changes.

- *Honeybee*

Energy Plus/Open Studio (for building energy, HVAC sizing, thermal comfort, etc.) and Radiance are two simulation engines with which Grasshopper3D is connected via Honeybee (for daylighting and glare simulation). In-depth daylighting and thermodynamic modeling, which are frequently most important in the middle and later stages of design, are supported by Honeybee. It specifically develops, runs, and displays the output of energy models using Energy Plus/Open Studio and radiation and daylight simulations using Radiance. It achieves this by integrating these engines with the Grasshopper/Rhino CAD environment. For these engines, it also functions as an object-oriented Software Development Kit (SDK).

- *Dragonfly*

By utilizing an abstracted 2D representation of building geometry, where all rooms are taken into account to be extrusions of floor plates, Dragonfly makes it possible to create and manipulate large-scale EnergyPlus and Radiance models.

- *Butterfly*

Advanced computational fluid dynamic (CFD) simulations can be run by connecting Grasshopper to the OpenFOAM engine through Butterfly. Butterfly is designed to execute various popular types of airflow simulations that are helpful for building design, as well as to easily convert geometry to OpenFOAM.

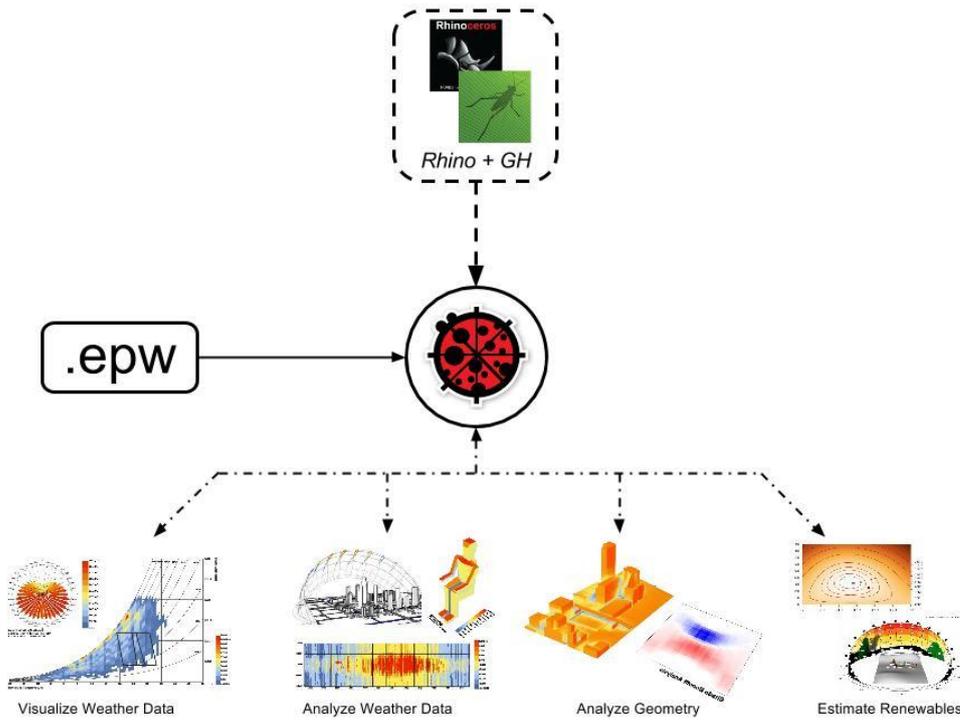


Figure 2.20 Ladybug diagram

For the purposes of the thesis research and the case studies done for daylight and glare analysis, Ladybug and Honeybee were used. Therefore the following *Figure 2.19\** and *Figure 2.20\**, are diagrams that show where exactly the tools can be applied and what engines they use (Ladybug Tools, 2020).

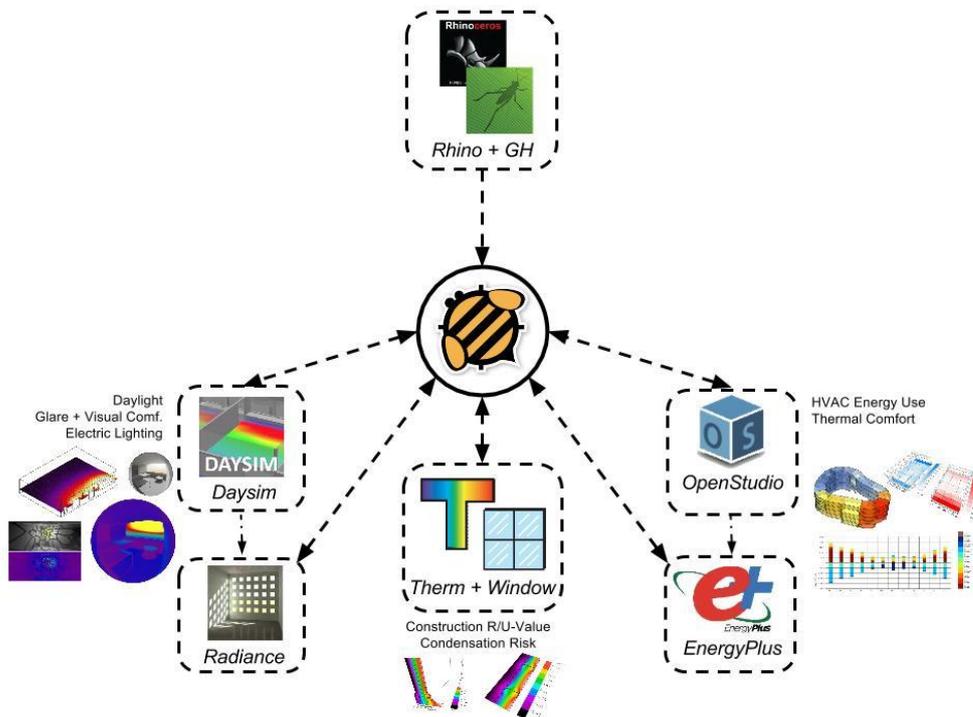


Figure 2.19 Honeybee diagram

\* Note. Retrieved from: <https://www.food4rhino.com/en/app/ladybug-tools>

## 2.8. POLLINATION FOR FUTURE ANALYSIS

During the research period, the newest updates connected to Ladybug tools were followed. While doing the analysis there was a new expansion in this field which will be just mentioned in order to compare the abilities of all the tools mentioned above. After several years of development as a commercial product that adds to the ecosystem of Ladybug Tools, Pollination is now available for usage from July, 2022. Pollination provides the independence and adaptability of Ladybug Tools while being far more user-friendly and available to you and your team. A flexible and linked ecosystem is produced through the interaction of plugins, apps, cloud computing, and package repositories (*Figure 2.21\**).

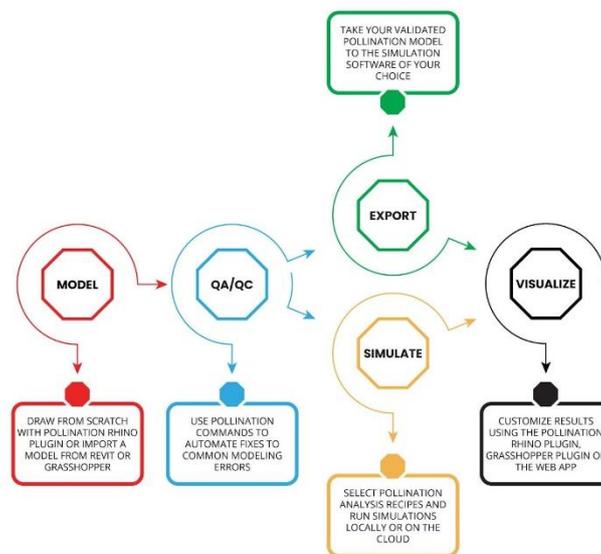


Figure 2.21 Pollination workflow

The Pollination ecosystem excels at addressing some of the most important pain points in the Atomic Energy Commission (AEC) sector. It was created using an anti-black-box methodology, which is exceedingly uncommon among services of a similar nature. For instance, analytical models can be created in Revit and simulated instantly online, or the model can be imported into Rhino or Grasshopper and make use of the corresponding Pollination plugins to validate it and run simulations. The model can be exported to yet another simulation program from a third party. Pollination and Ladybug Tools can perform practically all of the needs, but the models can still be exported to other tools if you wish. Some of the most cherished features of Ladybug Tools are now accessible to non-Grasshopper users thanks to Pollination CAD plugins for Rhino (and Revit). With Ladybug Tools Grasshopper plugins, they make it simple to outsource difficult or time-consuming jobs. Pollination subscribers can now use drop down menus and panels with simple navigation for Rhino's environmental simulation (and Revit) (Roudsari, 2022). Due to the delayed latest upgrade this research was reduced to only the previous inventions. Finding all this information was very important for the research, but only for comparison and possible future upgrading and refining of the thesis research (Figure 2.22\*).

\* Note. Retrieved from: <https://discourse.ladybug.tools/t/have-you-tried-pollination-yet/18784>

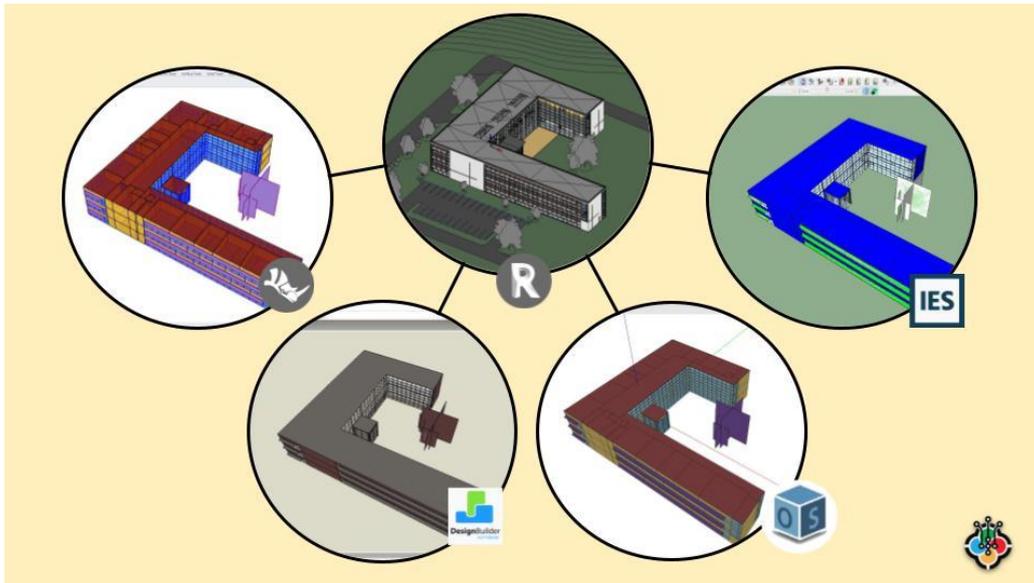


Figure 2.22 Capabilities of Pollination to link multiple tools

## 2.9. KEY PERFORMANCE INDICATORS (KPIs)

Key Performance Indicators (KPIs) are a crucial component in the sustainability of buildings assessment. They arise from the main design impact factors, which take into consideration few aspects of building comfort. In this research all of them and their physical variables were mentioned, but only the ones which represent visual comfort were evaluated within the case studies.

As *Table 2.2* shows, the main aspects of building comfort are: Visual Comfort, Thermal Comfort, Air Quality, Acoustic Comfort, Energy and Environmental Footprint/ LCA (Life Cycle Assessment). The main idea behind this table was to show the division into multiple segments, such as physical variables, performance indicators, target value, reference and connect all of them with the tools used inside Grasshopper to see which specific tool can calculate or asses a defined KPI.

As mentioned previously, the research was done mainly focusing on visual comfort of the building. Under visual comfort, there are three main aspects: Daylight, Glare and View-out. Such evaluation is impacted by some design choices which are made during the design phase and can be later changed due to the analysis results. Ones of the foremost design impacts, while simulating building envelope analysis for its visual comfort, are: Transparent Surface Size, Exterior and interior shading devices, Type of glass, Surrounding environment and Position of reference point. Nevertheless, Incident illuminance ( $E_v$ ), Reflectance ( $\rho$ ) and Refraction ( $\mu$ ) of the referenced surface, are taken into account for the further calculations of the KPIs. It is important to mention that all of the Target values in *Table 1* are referenced by the regulative norms and standards: International Education Standard (IES), International Organization for Standardization (ISO), Italian National Unification (UNI), European Standards (EN), American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE), etc.

Aspects	Performance indicators	Target value	Reference	Design Impact Factors	Physical Variables
Visual Comfort	Daylight	<ul style="list-style-type: none"> <li>Daylight Factor (DF) <ul style="list-style-type: none"> <li>≥ 2% - residential</li> <li>≥ 3% - commercial</li> <li>&gt; 50%</li> </ul> </li> <li>Useful Daylight Illuminance (UDI) <ul style="list-style-type: none"> <li>UDI<sub>100-3000</sub> ≥ 80%</li> </ul> </li> <li>Continuous Daylight Autonomy (cDA) <ul style="list-style-type: none"> <li>&gt; 55%</li> </ul> </li> <li>Spatial Daylight Autonomy (sDA) <ul style="list-style-type: none"> <li>sDA<sub>300,50%</sub> ≥ 40% - sufficient</li> <li>sDA<sub>300,50%</sub> ≥ 55% - preferable</li> <li>sDA<sub>300,50%</sub> ≥ 75% - optimal</li> </ul> </li> <li>Annual Sunlight Exposure (ASE) <ul style="list-style-type: none"> <li>ASE<sub>1000,250</sub> ≤ 10% (otherwise, identify how glare is addressed)</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Italian Technical Standard UNI 10840:2007</li> <li>IES LM-83-12</li> <li>UK Education Funding Agency. Baseline designs and strategies for schools in the Priority School Building Program (PSBP).</li> <li>IES LM-83-12</li> <li>IES LM-83-12</li> <li>LEED v4.1   U.S. Green Building Council</li> </ul>	<ul style="list-style-type: none"> <li>Transparent Surface Size <ul style="list-style-type: none"> <li>Exterior and interior shading devices</li> <li>Type of glass</li> </ul> </li> <li>Surrounding environment <ul style="list-style-type: none"> <li>Position of reference point</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Incident illuminance</li> <li>Reflection and refraction of the referenced surface</li> </ul>
	Glare	<ul style="list-style-type: none"> <li>Glare Autonomy <ul style="list-style-type: none"> <li>GA<sub>40%</sub></li> </ul> </li> <li>Spatial Glare Autonomy <ul style="list-style-type: none"> <li>sGA<sub>40%,5%</sub></li> </ul> </li> <li>Daylight Glare Probability (DGP) <ul style="list-style-type: none"> <li>&lt;20% - not validated</li> <li>20% - 35% - imperceptible</li> <li>35% - 40% - perceptible, but not disturbing</li> <li>40% - 45% - disturbing</li> <li>&gt; 45% - intolerable</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>EN 17037</li> <li>EN 17037</li> <li>EN 17037:2018</li> </ul>		
	View out	<ul style="list-style-type: none"> <li>Maximum view distance <ul style="list-style-type: none"> <li>↑</li> </ul> </li> <li>Number of visual layers <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Aesthetical scene quality <ul style="list-style-type: none"> <li>↑</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>EN 14501:2021</li> </ul>		
Thermal Comfort	<ul style="list-style-type: none"> <li>Predicted Percentage of Dissatisfied (PPD) <ul style="list-style-type: none"> <li>&lt; 20%</li> </ul> </li> <li>Predicted Mean Vote (PMV) <ul style="list-style-type: none"> <li>close to 0</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>ASHRAE 55 and ISO 7730</li> </ul>	<ul style="list-style-type: none"> <li>Building materials <ul style="list-style-type: none"> <li>Thickness of component <ul style="list-style-type: none"> <li>Total area</li> </ul> </li> </ul> </li> <li>HVAC</li> </ul>	<ul style="list-style-type: none"> <li>Mean radiant temperature</li> <li>Operative temperature</li> <li>Relative humidity</li> <li>Activity level</li> <li>Air velocity</li> <li>Clothing</li> </ul>	
Energy	<ul style="list-style-type: none"> <li>Primary Energy Use <ul style="list-style-type: none"> <li>↓ (kWh)</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>EPBD- 2010/31/EU</li> </ul>	<ul style="list-style-type: none"> <li>Features of building envelope <ul style="list-style-type: none"> <li>Type and size of building <ul style="list-style-type: none"> <li>HVAC</li> </ul> </li> </ul> </li> <li>Equipment systems <ul style="list-style-type: none"> <li>Renewable Energy Source</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Opaque and transparent surfaces properties</li> <li>Energy loads and zones</li> <li>Occupancy schedule</li> <li>Equipment schedule <ul style="list-style-type: none"> <li>Location</li> </ul> </li> </ul>	
Acoustic Comfort	<ul style="list-style-type: none"> <li>Reverberation time (RT) <ul style="list-style-type: none"> <li>RT<sub>60</sub> &lt; 1s - good for classrooms</li> <li>RT<sub>60</sub> = 1s - good for speaking; articulation of speech is clear</li> <li>RT<sub>60</sub> = 1.5s - 2.5s - a good compromise if the room is to be used for both speaking and music</li> <li>RT<sub>60</sub> = 3.5s - better for music, but some loss of articulation. Would likely be difficult to understand speech</li> <li>RT<sub>60</sub> = 8 s to 11 s. Large medieval cathedrals will have a very long RT<sub>60</sub></li> </ul> </li> <li>Weighted Sound Reduction index <ul style="list-style-type: none"> <li>↑</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>ISO 3382-1 (performance spaces)</li> <li>ISO 3382-2 (ordinary rooms)</li> <li>ISO 3382-3 (open-plan offices)</li> <li>ASTM E2235 standard</li> <li>ISO 16283-2:2020</li> </ul>	<ul style="list-style-type: none"> <li>Building materials <ul style="list-style-type: none"> <li>Thickness of component</li> </ul> </li> <li>Windows and doors area <ul style="list-style-type: none"> <li>Total area</li> </ul> </li> <li>Surrounding environment</li> </ul>	<ul style="list-style-type: none"> <li>Frequency and Amplitude of the incident sounds</li> <li>Mass and Natural resonances <ul style="list-style-type: none"> <li>Absorption, Transmittance and Reflection of sounds</li> <li>Air tightness</li> </ul> </li> <li>Presence of people</li> </ul>	
Environmental Footprint-LCA	<ul style="list-style-type: none"> <li>Global Warming Potential (GWP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Acidification Potential (AP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Eutrophication Potential (EP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Photochemical Ozone Creation Potential (POCP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Ozone Depletion Potential (ODP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Abiotic Depletion Potential (ADP) minerals, metals and fossil resources <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Water Deprivation Potential (WDP) <ul style="list-style-type: none"> <li>↓</li> </ul> </li> <li>Embodied/ Operational impact <ul style="list-style-type: none"> <li>↓</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>EN 15804</li> <li>EN 15804</li> <li>EN 15804</li> <li>EN 15804</li> <li>EN 15804</li> <li>EN 15804</li> <li>ISO 14046</li> <li>EN 15978</li> </ul>	<ul style="list-style-type: none"> <li>Type of material <ul style="list-style-type: none"> <li>HVAC</li> </ul> </li> <li>Renewable/ Non-renewable Energy Source <ul style="list-style-type: none"> <li>Geometry</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>The absorption of infrared radiation by a given gas</li> <li>The time horizon of interest (integration period) <ul style="list-style-type: none"> <li>The atmospheric lifetime of the gas</li> </ul> </li> <li>Atmospheric pollution from SO<sub>2</sub> or NH<sub>3</sub> <ul style="list-style-type: none"> <li>VOCs</li> </ul> </li> <li>Water demand</li> <li>Embodied/ Operational energy <ul style="list-style-type: none"> <li>Reuse and recycle</li> </ul> </li> </ul>	
Air Quality	<ul style="list-style-type: none"> <li>Ventilation Rate <ul style="list-style-type: none"> <li>&gt; 7.5 L/s<sup>-1</sup> PP</li> </ul> </li> <li>CO<sub>2</sub> <ul style="list-style-type: none"> <li>1080 mg m<sup>-3</sup></li> </ul> </li> <li>PM10 (Particulate matter) <ul style="list-style-type: none"> <li>0.05 mg m<sup>-3</sup></li> </ul> </li> <li>Indoor Air Quality Index (IAQ) <ul style="list-style-type: none"> <li>&lt; 100</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>WHO AQGs</li> </ul>	<ul style="list-style-type: none"> <li>Building materials <ul style="list-style-type: none"> <li>Thickness of component</li> </ul> </li> <li>Windows and doors area <ul style="list-style-type: none"> <li>Total area</li> </ul> </li> <li>HVAC</li> <li>Surrounding environment</li> </ul>	<ul style="list-style-type: none"> <li>Airflow</li> <li>Humidity</li> <li>Presence of people</li> </ul>	

Table 2.2 Key Performance Indicators KPIs

## 2.10. DAYLIGHT

High levels of illumination and great color rendering are provided by daylight. Due to these two characteristics, daylight creates the ideal environment for clear vision. However, daylight can also provide distracting solar glare and extremely bright reflections on displays, both of which impair clear vision. As a result, the way sunshine is provided affects how well tasks are performed. When designing for daylighting in buildings, all of these elements must be taken into account (Ruck, 2000).

### 2.10.1. DAYLIGHT FACTOR (DF)

Daylight Factor (DF) is the ratio of horizontal indoor to outdoor illumination by daylight under continuously overcast sky conditions, expressed as percentages. In the past, determining the average daylight factor by hand or using a computerized version of the manual approach were the only ways to evaluate the brightness of the light in a space. While the output from the majority of computer-based calculations shows the average daylight factor generated from a number of points on the working plane, the term “average daylight factor” is occasionally interpreted to mean the average daylight factor on all surfaces. Commonly applied as a design criterion, given by the CEN Daylight Standard (EN 17037) the average daylight factor is advised to be over 2% (Kubba, 2012).

$$DF = \frac{E_{P,obs}}{E_{P,unobs}}$$

Figure 2.23 Daylight Factor calculation

$E_{P,obs}$  is the horizontal illuminance at a point  $P$  due to the presence of a room that obstructs the view of the sky;  
 $E_{P,unobs}$  is the horizontal illuminance at the same point  $P$  if the view of the sky is unobstructed by the room.

### 2.10.2. DAYLIGHT AUTONOMY (DA)

The percentage of occupied time that daylight meets the specified illuminance at a given position in a space is known as Daylight Autonomy (DA), which is a measure of daylight availability. In general, daylight areas are those where the DA is greater than 50% (Reinhart C. F., 2001).

$$DA = \frac{\sum_i (wf_i \cdot t_i)}{\sum_i t_i} \in [0, 1]$$

with  $wf_i = \begin{cases} 1 & \text{if } E_{Daylight} \geq E_{limit} \\ 0 & \text{if } E_{Daylight} < E_{limit} \end{cases}$

Figure 2.24 Daylight Autonomy calculation

$t_i$  is each occupied hour in a year;  
 $wf_i$  is a weighting factor depending on values of  $E_{Daylight}$  and  $E_{limit}$  that are respectively, the horizontal illuminance at a given point due to the sole daylight and the illuminance limit value.

### 2.10.3. CONTINUOUS DAYLIGHT AUTONOMY (CDA)

Values below the user-specified threshold receive linear partial credit from Continuous Daylight Autonomy. If a certain point consistently exceeded 300 lux on a yearly basis and 300 lux was designated as the DA threshold (DA300), the cDA300 might produce a value of roughly 55–60% or more (Šprah, 2019).

$$DA_{CON} = \frac{\sum_i (wf_i \cdot t_i)}{\sum_i t_i} \in [0, 1]$$

with  $wf_i = \begin{cases} 1 & \text{if } E_{Daylight} \geq E_{limit} \\ \frac{E_{Daylight}}{E_{limit}} & \text{if } E_{Daylight} < E_{limit} \end{cases}$

In contrast to earlier definitions of DA, partial credit is attributed to time-steps when the measured daylight illuminance ( $E_{Daylight}$ ) lies below the limit ( $E_{Limit}$ ).

Figure 2.25 Continuous Daylight Autonomy calculation

### 2.10.4. SPATIAL DAYLIGHT AUTONOMY (SDA)

On a yearly basis, Spatial Daylight Autonomy (sDA) evaluates whether a space receives enough daylight on a work plane during normal business hours. For 50% of the time that an area is occupied, 300 lux is the desired level. Larger interior spaces that have higher sDA values get at least 300 lux of daylight for at least half of the working day. Typically, sDA is computed using a daylight simulation program that determines the amount of daylight in the area for each hour of the year (Council, 2013).

$$sDA_{x/y\%} = \frac{\sum_i (wf_i \cdot DA)}{\sum_i p_i} \in [0, 1]$$

with  $wf_i = \begin{cases} 1 & \text{if } DA \geq DA_{limit} \\ 0 & \text{if } DA < DA_{limit} \end{cases}$

$x$  is the reference illuminance level,  $y$  is the time fraction,  $p_i$  are the points belonging to the calculation grid.

Figure 2.26 Continuous Daylight Autonomy calculation

### 2.10.5. ANNUAL SUNLIGHT EXPOSURE (ASE)

Annual Sunlight Exposure (ASE) is the proportion of a space that is exposed to excessive amounts of direct sunlight (1000 Lux or more for at least 250 hours of occupied time per year), which might result in glare or higher cooling loads. A relative value with smaller sunny regions that receive no more than 1000 lux of direct sunlight for more than 250 hours per year is advised by IES. Although there isn't a set cutoff for this statistic, settings with a visual comfort score of 10% or above are considered visually

uncomfortable, 7% neutral and 3% are unambiguously acceptable. Prior to the deployment of operable shades to prevent direct sunlight, ASE is calculated using the IES simulation approach. As a metric to assist designers in limiting excessive sunlight in a place, ASE is intended to complement sDA. (Heschong, 1999).

## 2.10.6. USEFUL DAYLIGHT ILLUMINANCE (UDI)

The term “Useful Daylight Illuminance” (UDI), which was derived from the term “Daylight Illuminance,” divides the annual illuminance distribution into three bins using lower and upper thresholds. The lower bin (UDI 100 lux) represents the time when there is insufficient daylight, the upper bin (UDI 3000 lux) presents the time when there is an abundance of daylight, and the intermediate bin (UDI 100-3000 lux) presents useful daylight. Generally, 100 lux and 3000 lux are set to be the lower and upper thresholds (Nabil, 2005).

$$UDI = \frac{\sum_i (wf_i \cdot t_i)}{\sum_i t_i} \in [0, 1]$$

$$\left\{ \begin{array}{l} UDI_{Overlit} \quad \text{with } wf_i = \begin{cases} 1 & \text{if } E_{Daylight} > E_{Upper\ limit} \\ 0 & \text{if } E_{Daylight} \leq E_{Upper\ limit} \end{cases} \\ UDI_{useful} \quad \text{with } wf_i = \begin{cases} 1 & \text{if } E_{Lower\ limit} \leq E_{Daylight} \leq E_{Upper\ limit} \\ 0 & \text{if } E_{Daylight} < E_{Lower\ limit} \vee E_{Daylight} > E_{Upper\ limit} \end{cases} \\ UDI_{Underlit} \quad \text{with } wf_i = \begin{cases} 1 & \text{if } E_{Daylight} < E_{Lower\ limit} \\ 0 & \text{if } E_{Daylight} \geq E_{Lower\ limit} \end{cases} \end{array} \right.$$

Figure 2.27 Useful Daylight Illuminance calculation, representing the generally considered lower and upper thresholds

Moreover the UDI can be divided into four groups considering the value of lux:

- UDI *fell short* (0–100 lux) insufficient illumination;
- UDI *supplementary* (100 - 500 lux) integrate with electric light;
- UDI *autonomous* (500 - 3000 lux) no electric light required;
- UDI *exceeded* (>3000 lux) bad situation: glare, overheating, etc.

Therefore, any daylight illumination in the range of 100 lux to 3000 lux should be regarded as "useful" for lighting the space. The full lit area must be taken into account. As a result, anytime all the illuminances fall within the range of 100 to 3000 lux, "useful daylight illuminance" is stated to occur while evaluating time-varying daylight illuminances throughout the work plane. The evaluation of a daylit space must, as was said previously, be based on time-varying daylight illuminances for a whole year at a time-step of an hour or less.

## 2.11. GLARE

Glare refers to the visual situations when discomfort, annoyance, or a decline in visual performance and visibility are brought on by an inappropriate brightness distribution. There are two possible types of glare: disability glare and discomfort glare. Disability glare occurs when there is too much light and the person inside can't see, and discomfort glare occurs when there is a range of luminance in the field of vision and it impairs vision and wears the eyes out (Carlucci, 2015).

### 2.11.1. DAYLIGHT GLARE PROBABILITY (DGP)

Daylight glare probability (DGP) measures the probability that an occupant will perceive glare from daylight at a specific time, location, and viewing angle. It is computed by taking into account the light intensity, size of the glare source, contrast, and its location in the field of view when evaluating the possible occupant's whole visual field.

$$DGP = 5,87 * 10^{-5} * E_v + 9,18 * 10^{-2} \log \left( 1 + \sum \frac{L_{s,i}^2 * \omega_{s,i}}{E_v^{1,87} * P_i^2} \right) + 0,16 \quad (1)$$

$E_v$  represents vertical illuminance at eye [lx];

$L_s$  is a source luminance [cd/m<sup>2</sup>];

$\omega_s$ , a solid angle [steradians];

Figure 2.28 Daylight Glare Probability calculation

Imperceptible Glare (<0.35), Perceptible Glare (0.35-0.4), Disturbing Glare (0.4-0.45), and Intolerable Glare (>0.45) are the four levels of glare that are expressed on a scale from 0 to 1 (Vos, 2003).

DGP Ranges (DGR)			
Imperceptible	Perceptible	Disturbing	Intolerable
DGP<0.35	0.35<DGP<0.40	0.40<DGP<0.45	DGP>0.45

Figure 2.29 DGP Ranges

### 2.11.2. GLARE AUTONOMY (GA)

Glare autonomy can be defined as the percentage of time the view is free of glare given a vector of DGP values for all occupied hours for a specific view. When DGP is less than a certain percentage, like 40%, we consider the view to be glare-free (McNeil, 2016).

### 2.11.3. SPATIAL GLARE AUTONOMY (SGA)

Spatial Glare Autonomy is a metric describing the percentage of the sensor grid that is free glare according to the glare threshold and the target time. The sGA value is expressed as a percentage of the sensors in the analysis grid.

## 2.12. VIEW-OUT

Numerous scientific studies (Heschong, 2021) have shown the value of views of the outdoors to people’s health and well-being in a world where we spend more than 90% of our time indoors. Particularly views of nature have been shown to boost cognitive function and lower stress levels. This feature has developed into a crucial component of visual comfort, nearly as important as the flood of natural light. One might think that having windows would be sufficient to give residents vistas and the advantages that come with them (Hellinga, 2013).

## 2.13. USING GRASSHOPPER PLUGINS FOR KPIS ANALYSIS

The following *Table 2.3* represents the available plug-ins in Grasshopper which could evaluate the provided KPIs, divided into visual, thermal, acoustic comfort, energy, air quality and LCA.

Aspects	Performance indicators	Honeybee	Ladybug	Butterfly	Pachyderm	Bombyx	OneClickLCA	Engines
Visual Comfort	Daylight	• Daylight Factor (DF) • Daylight Autonomy (DA) • Useful Daylight Illuminance (UDI) • Continuous Daylight Autonomy (cDA) • Spatial Daylight Autonomy (sDA) • Annual Sunlight Exposure (ASE)	✓					RADIANCE
	Glare	• Glare Autonomy • Spatial Glare Autonomy • Daylight Glare Probability (DGP)	✓					
	View out	• Maximum view distance • Number of visual layers • Aesthetical scene quality	✓	✓				
Thermal Comfort	• Predicted Percentage of Dissatisfied (PPD) • Predicted Mean Vote (PMV)	✓	✓	✓				THERM ENERGY +
Energy	• Primary Energy Use	✓						ENERGY+
Acoustic Comfort	• Reverberation time (RT) • Weighted Sound Reduction index				✓			PACHYDERM
Environmental Footprint-LCA	• Global Warming Potential (GWP) • Acidification Potential (AP) • Eutrophication Potential (EP) • Photochemical Ozone Creation Potential (POCP) • Ozone Depletion Potential (ODP) • Abiotic Depletion Potential (ADP) minerals, metals and fossil resources • Water Deprivation Potential (WDP) • Embodied/ Operational impact					✓	✓	
Air Quality	• Ventilation Rate • CO2 • PM10 (Particulate matter) • Indoor Air Quality Index (IAQ)	✓						ENERGY+

Table 2.3 KPIs to Grasshopper tools

### 3. METHODOLOGY

To begin with, this chapter follows a detailed explanation of the workflow, designed to construct the 3D model, to import all the information in Grasshopper using RhinoInside, make an algorithm for further visual comfort analysis with Ladybug and Honeybee and run the simulations. Based on the research inquiries that were earlier discussed in chapter one, the research questions were as follow:

RQ1 What is the importance of BPSTs for Visual Comfort analysis?

RQ2 What are the steps of achieving smooth interoperability between BIM and VPL software programs?

RQ3 Is smooth interoperability a key factor for assessing building envelope performance studies?

RQ4 What are the pros and cons of Rhino Inside Revit for Daylight and Glare analysis?

In the interest of answering these questions, the literature review was considered, which more precisely answered the first question about BPSTs. Followed by the gathered information about BIM and VPL it was continued to explore the interoperability between them and tried multiple scenarios in order to achieve smoother workflow. In this chapter all the data previously collected from the literature review, was used in practice. Normally coming across some errors and obstacles brought a realization that in order to get to the end point few algorithm paths were needed to be tried, starting from importing all components to running the simulations.

As previously addressed, this research focuses on two case studies. The first one being a single-zone model consisting one small office and the second one, a more complex building with multiple rooms, such as offices, meeting room, utility and sanitary room, called multi-zone model. Furthermore, as a starting point this chapter explains the details of the building, model inputs in Revit and all the changes to the initial model design that need to be done, to pursue with daylight and glare analysis in Grasshopper. It elaborates the different data transmission while doing divergent steps, made because of the contrast of the buildings due to their complexity. Also, because of multiple types of exportation of the building components into Grasshopper, there is a more detailed description on how to visualize the elements, such as, interior and exterior walls, windows, floors, roofs. etc. and how to use them for the purpose of daylight and glare analysis. For some of the geometries there is also a modification in both Revit and Grasshopper. Then, it describes the possible solutions, taking into consideration errors and data loss which occurred during this forwards interoperability. This research covers the part with specifically constructing the Honeybee model, which is crucial for running the simulations. More errors and modifications were found during this process. Lastly it explains all the simulations that were run, focusing to a large extent on the visual comfort analysis. This means that the above mentioned KPIs in Chapter 2, are calculated for both case studies using the components provided by Ladybug and Honeybee. One thing worth to mention is that view-out is not tested during this analysis.

Wrapping up Chapter 3, there is a report on how the results impact the design and overall performance of each case study. Most importantly this part explains all the problems and achievements when performing the simulations according to the given results.

### **3.1. DESCRIPTION OF RESEARCH METHODS**

So far the thesis research follows few research methods. The two case studies were intentionally selected because of their different complexity of the models. So the observed methods pay attention on modeling the object, including the right changes to the model in Revit and it makes the workflow a bit different for both case studies. Another technique is using the least complicated algorithm inside Grasshopper for importing the Revit elements. With the use of simple and few components provided by Revit tools within Grasshopper UI, developing a manageable algorithm was the achieved success.

After this step the Honeybee model was constructed, which is necessary for further analysis of the models. Both for the single-zone and multi-zone model there was almost the same first steps in choosing the right Honeybee face, applying all the faces to Honeybee room and then constructing the Honeybee model. What was really important was applying all apertures and shading devices to the Honeybee model. The difference between each scenario was the type of glass on the window, type of the material layers in the components and mainly shading device.

The research methods not only have their focus on these specifications, but most importantly on the type of shading device control. As the research was developed there is a part where the use of dynamic shading was imported. This helped to understand “what is the difficulty of finding an algorithm which can then be used in the simulations in the same way as static shades work, but with an automatically controlled shading elements”.

Another method of getting the desired outcome of the thesis research was commenting and comparing the final results of the simulation and see what are the pros and cons of the interoperability with Rhino Inside Revit, once the analysis results are previewed.

### **3.2. CONCEPTUAL WORKFLOW**

The conceptual study framework is consisted of few steps (Figure 3.1) following the methodology workflow from beginning to end. It is based on the path followed continuously, starting with constructing the 3D model, ending back to the updated 3D model. Basically it shows the interoperability through both case studies and it makes a circular motion, with possibility to again repeat the process of same steps in order to evaluate once again the building performance of its envelope.

The first step is deciding what type of case studies should be tested. The reason for choosing this type of buildings is, as previously mentioned, their complexity. Next, based on the required model properties and the desired outputs, two models were built. In the following few steps the 3D model was developed in Revit, preparing it correctly for the exportation in Grasshopper. This means that from the literature review and the knowledge gained during the first part of the research, the capabilities were applied and it was understood how to model a correct 3D model for visual comfort analysis. As next steps Rhino Inside Revit is used and the Grasshopper UI is opened through Revit. Then from the

components given by Revit tool bar in Grasshopper, the ones that take the geometry components and their properties by Element Type are chosen.

After this step, the geometry adjustments are made, with choosing the right faces of the wall components and connect them creating a closed boundary from all surfaces, such as, wall, floor and roof surfaces. Windows are applied according to their glass position, laying on the wall surface, chosen previously, and shading devices are constructed whether from Revit or with Honeybee appropriate tools. Next step is adjusting the material properties of each component. Then all of these elements are inputted into one component, creating Honeybee Model. Then this model is ready for evaluation.

First to last step is running the simulations for daylight and glare, with Honeybee components and visualizing them with Ladybug components. Lastly, the comparison between the results is described and used as understanding for a future step, which could be implementation of the backwards interoperability, from Grasshopper to Revit.

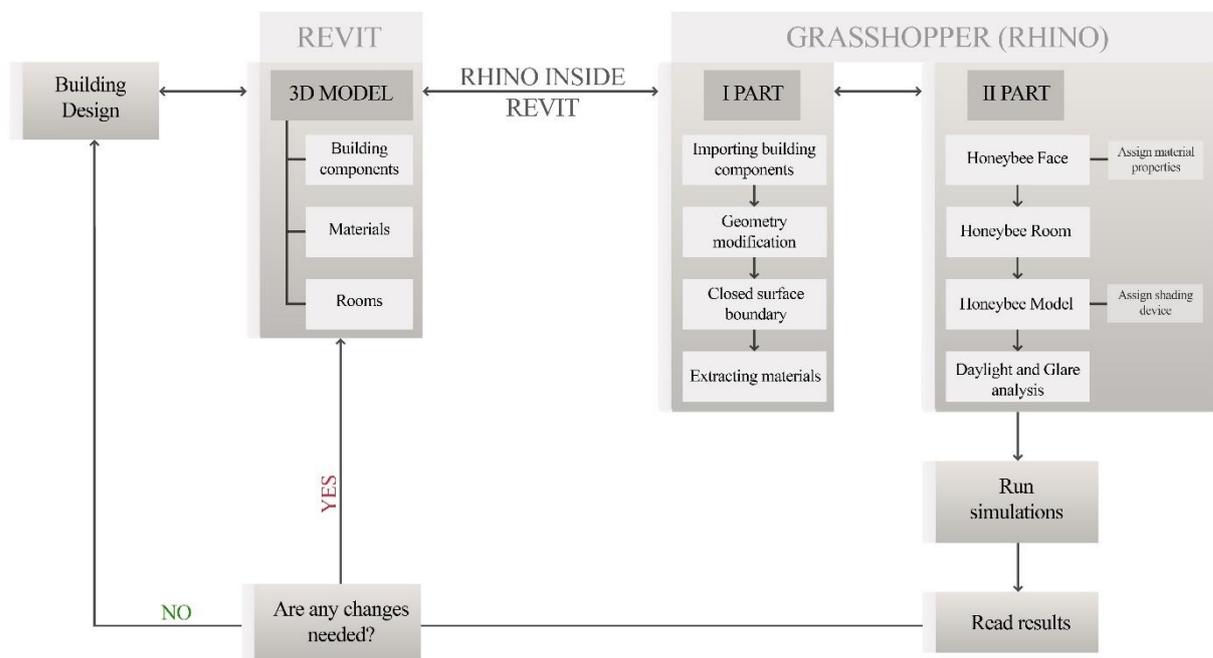


Figure 3.1 Conceptual workflow of the structure of methodology

### 3.4. WORKFLOW TYPES USING RHINO INSIDE REVIT AS INTEROPERABILITY TOOL

As stated above, Revit is used as a BIM software program due to its daily use by professionals. The reason behind this is that Building Information Modeling (BIM), which uses a three-dimensional model of the building built with input from all the disciplines to develop an integrated design, brings the process of sharing the information between multiple software programs.

The following case studies both start from the same starting point, building the model in Revit. Some of the differences are adjusted due to the complexity of the models. They are different because the first one consists one room and the other one multiple rooms. The exchange of data is made with Rhino Inside Revit, because from all of the studies mentioned in the literature review chapter, this plugin have performed the smoothest interoperability, due to its ability to transmit information between different types of software programs, back and forth. The most important thing to mention is that Rhino Inside Revit allows controlling the changes from both Revit and Grasshopper and if there is a modification made within one of the programs, it can be automatically passed on the other program and vice versa. During the research exercises it is shown that when having a more complex building there is another workflow to be followed in order to avoid few unnecessary steps while performing the "forwards" interoperability. In the first part of the exportation of model components from Revit to Grasshopper there is only one direction of transmitting the components' data. Also when doing the analysis on Grasshopper, if a change of a material property is made within Revit, the data will immediately appear on Grasshopper.

The final point to address is that in the most case scenarios interoperability is used for geometry transformations of the building or when an extension is designed to an existing building in the later stages of the design process. In practice, when having a 3D model with all the information about the components in Revit, transferring the data to Grasshopper, evaluating the building performance with Grasshopper tools, if there is a need of a change in the shape of a particular component due to the obtained results, it is possible to do it with Rhino Inside Revit. In the following few pages, the interoperability between Revit and Grasshopper is explained within few steps divided in two workflows for the Case Study I and Case Study II.

### 3.4.1. TYPE I – EXPORTING THE COMPONENTS AND THEIR MATERIAL PROPERTIES FROM REVIT TO GRASSHOPPER

Starting with the first case study, which is a single zone model, the exportation of model elements was tested directly in Grasshopper, using the components from the Revit library in the Grasshopper toolbar. The idea was to try and construct the model in Grasshopper, by selecting the surfaces directly from the elements, such as: roof, floor, walls, apertures, etc.

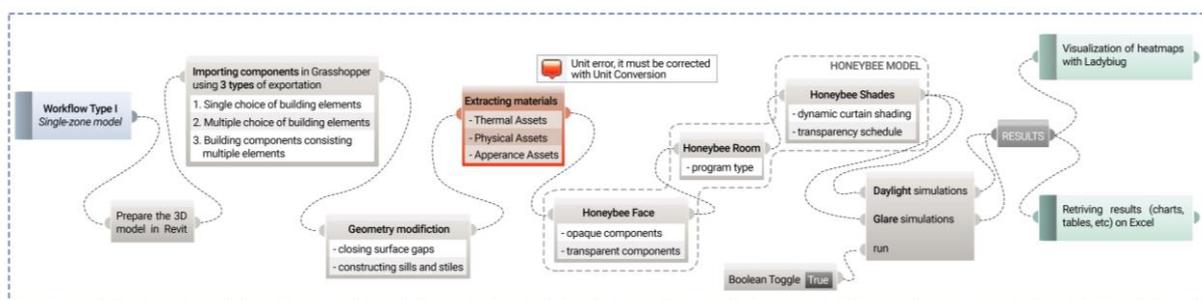


Figure 3.2 TYPE I – Exporting the components and their material properties from Revit to Grasshopper

Figure 3.2 shows the steps of constructing the first type of workflow. Starting from the preparation of the 3D model to exporting the building elements in Grasshopper, the steps follow the modification of the geometry if there are any changes to be made. Furthermore, the material properties provided in Revit were extracted in Grasshopper, where their thermal assets could be used for the purpose of the daylight and glare analysis. After this the algorithm continues with the creation of Honeybee Model and running the simulations.

### 3.4.2. TYPE II – EXPORTING THE ROOMS FROM REVIT AND THEN ADJUSTING MATERIAL PROPERTIES IN GRASSHOPPER

The second type of workflow was according to the room categorization which was assigned previously in Revit. Main reason behind this type of exportation of the 3D model in Grasshopper, was due to its complexity as multi-zone model, with multiple rooms. Most importantly, the second type was tested whether the room exportation will be faster, simpler and more accurate, compared to the elements exportation.

As the second model was more complex the main idea was to find a way of having a smoother interoperability, without using too much time and taking multiple steps over and over again. Also after analyzing the first workflow, as a consequence of errors and complications that got into the way of geometry and material properties exportation, it appeared that a different type of exportation had to be tested.

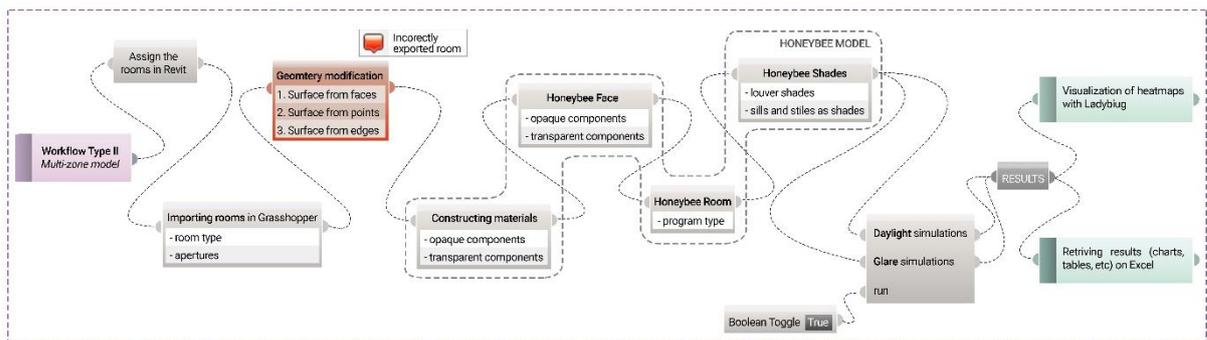


Figure 3.3 TYPE I – Exporting the rooms from Revit and then adjusting material properties in Grasshopper

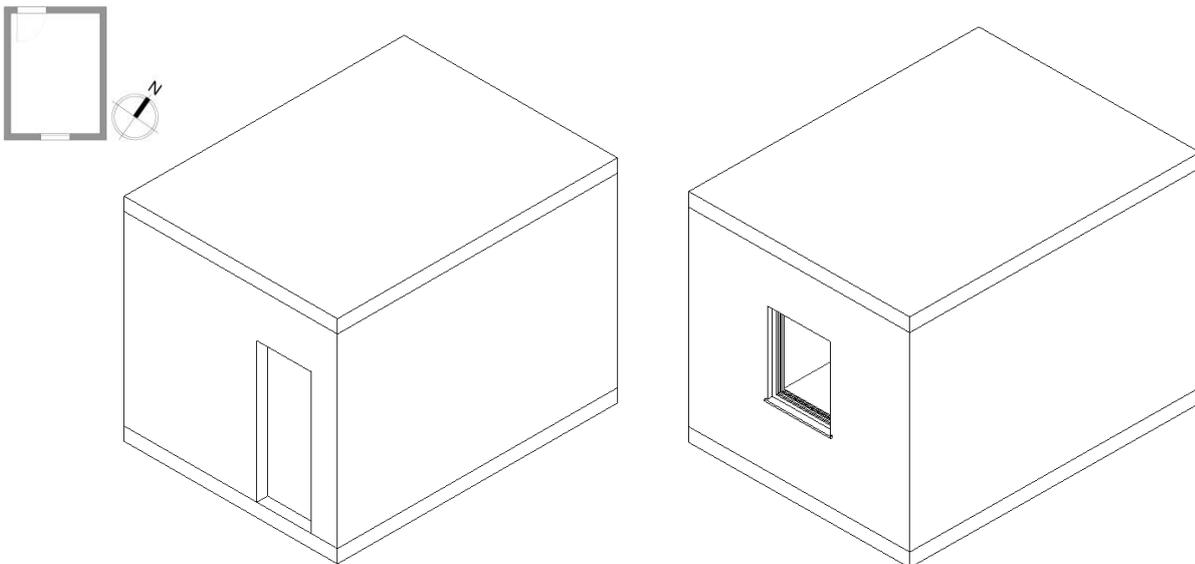
Figure 3.3 shows the full workflow from beginning to end. Step by step is explained every motion made on Revit, from assigning the rooms to exporting them on Grasshopper. It is easier to understand how these steps connect and work together in order to perform the desired simulations and obtained accurate results.

### 3.5. CASE STUDY I – SINGLE-ZONE MODEL

For the purpose of the research the first test model was designed to be a basic model consisting one zone. Because of the unknown expectations from the analysis, the initial idea was to see how Grasshopper accepts simple models of buildings, with minimum complexity, and how the simulations can be performed based on the transmitted information from Revit. From the previous research analysis the knowledge I got was that if you add more to the complexity of the model, there is a chance of having more errors and data loss, when transferring the data from BIM to VPL software programs. By reason of this information, found in prior articles and research papers, I decided to start with a simple "shoe-box" design.

#### 3.5.1. DESCRIPTION OF THE MODEL

Firstly it was started with the geometry, because it was crucial to choose what type is necessary for a simple model. Due to the fact that curved walls would be a challenge to try and use them in the basic model, it was decided to go with a rectangular shaped geometry (*Figure 3.4*).



*Figure 3.4 Single-zone model*

The model represents a small office of 11m<sup>2</sup>, situated in Turin, Italy, with a north rotation of 35°. The single-zone model contains a single window and door. The room itself has measurements of 3.2x4.2x3 meters, the window 0.9x1.2 meters on the SW (South-West) facade and the door 0.8x2.1 meter on the NE (North-East) facade. In the analysis the door was excluded. Window to floor area ratio (WFR) results in 9.9%. The wall components are one layer material (ex: concrete), which was added because later for the analysis all material assets were implemented in Grasshopper with a help of components provided from Revit tool bar within Grasshopper. The same material was applied to the roof and floor components. For the window there was a use of glass type from Revit library.

### 3.5.2. SIMULATION MODEL INPUTS IN REVIT

Specific modifications on both case studies' models needed to be done. This was due to easier control on the components when imported in Grasshopper. In this part of the research there is a representation of steps to follow for solving some issues with the geometry of the single-zone model in Revit, which can apply to all types of models if the same workflow as in this case study is followed. In order to evaluate a model within VPL software program and do analysis on the building envelope, only a closed box is accepted. It means the building elements are not selected as solid components, but their surfaces. For this purposes, it's better to adjust few modifications in Revit before transmitting the data to Grasshopper. The simulation model inputs were changed after both of the buildings were modeled as normally they should be, in order to prepare a 3D model. Then an importation in Grasshopper with Rhino Inside Revit was tried. After considering that it is time consuming to make all the changes in Grasshopper, which are the corrections that are easier to be done in Revit, were found.

#### 3.5.2.1. DESIGN STRATEGIES

Since in Grasshopper, for comfort analysis, Honeybee accepts only closed surface boundary, it is necessary to correctly check the previously constructed 3D model. Furthermore, it is precisely mentioned, within the rules of inputting data in Honeybee components for further evaluation, that from all components only one surface, which corresponds to the preferred analysis, should be selected. For example the walls, roof and floor of a single room, should have their internal or external surfaces selected, which will then complete a closed surface boundary. On the other hand Revit extends or cuts the components under default choice. *Figure 3.5* shows how perpendicular walls are trimmed differently.

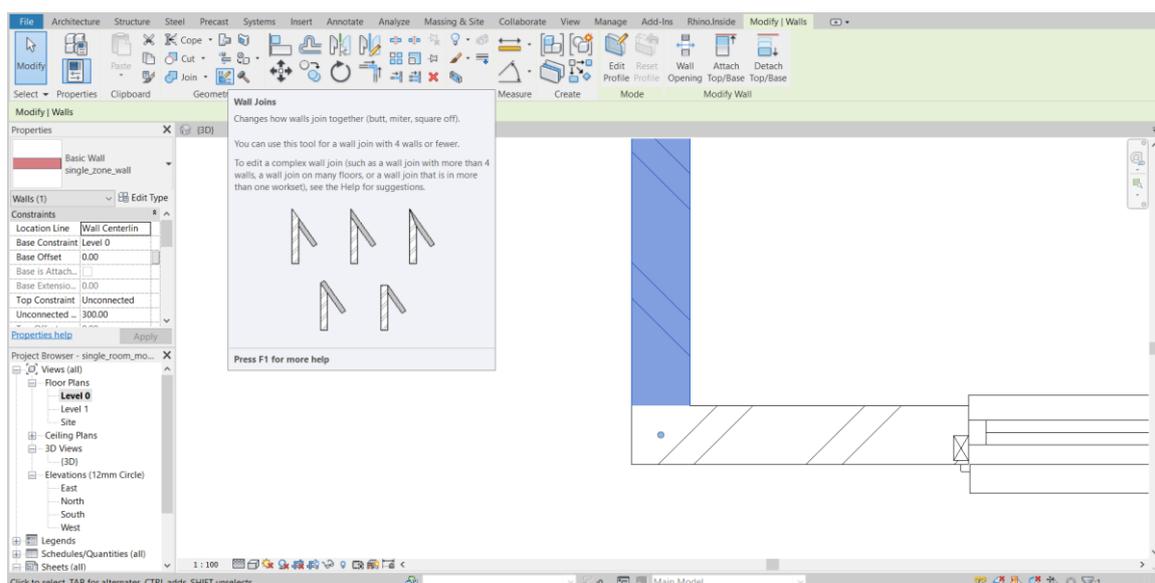
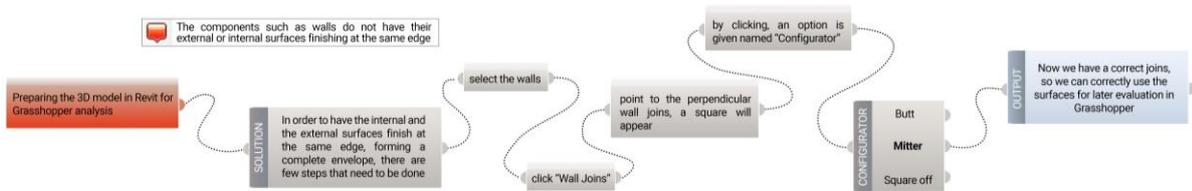


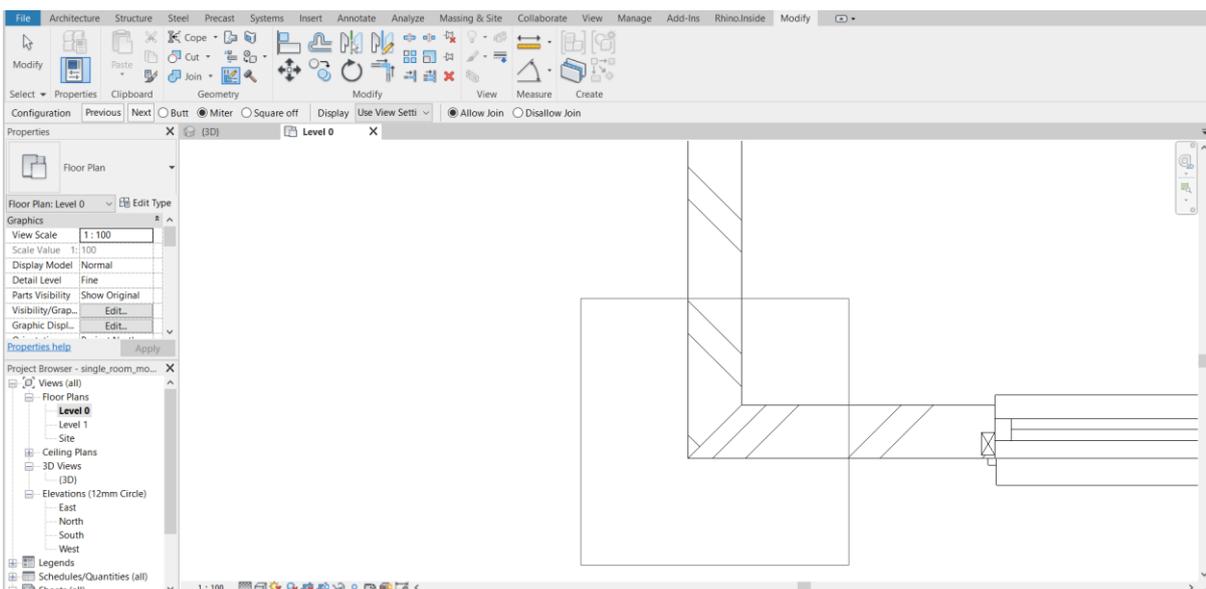
Figure 3.5 Screenshot from Revit UI, interception of wall surfaces

The internal surfaces of the wall components are meeting at a curve where one is intersecting the other internal surface. Opposite of this, the external surfaces are not crossing each other, one is simply intercepted by the internal surface of the other wall component. As a result this will show as a gap in Grasshopper, when choosing the right surface for further construction of the closed surface boundary. The solution to this problem is better to be corrected in Revit, as it doesn't change anything with the building geometry design. There is a tool called "Wall joins" where a different cut for the perpendicular walls can be chosen. All of the steps are elaborated in *Figure 3.6*, using graphical style to explain the path on how to edit this in Revit.



*Figure 3.6 Step by step trim correction of perpendicular wall components in Revit*

*Figure 3.7* shows inside Revit UI, where can be clearly seen how the wall connection changes once different configuration of "Wall joins" is selected. The choice between "Butt", "Mitter" and "Square off" is that, "Butt" is the first choice in Revit which goes under default and is explained above. If "Next" is clicked that means that it changes the wall that has priority for the edge extension. "Mitter" joins the perpendicular wall components, connecting the external surfaces from both walls into one curve and the internal surfaces from both walls in another curve. This means that the walls are diagonally trimmed at the edges. The "Square off" is used for walls that are not perpendicular, but one of them is diagonal or they both do not form an angle of 90 degrees. This button is used to "square off" one wall and avoid angular edge. In this particular case for Honeybee analysis in Grasshopper, "Mitter" wall join was used due to easier selection of wall surfaces, in order to create closed envelope. For the purpose of the research only this type of wall join was explained in detail, with graphical schemes.



*Figure 3.7 Screenshot from Revit UI, correction of wall trimming*

Another important correction to do is connected to floors and roofs. Doing the daylight and glare analysis both roof and floor surfaces need to be adjusted to correspond with the chosen surfaces from the wall components. Even though this can be modified in Grasshopper, an easier way is using Revit for this type of changes. As explained above, it could be either chosen an internal or external surfaces of the components, which would led to closed surfaced boundary.

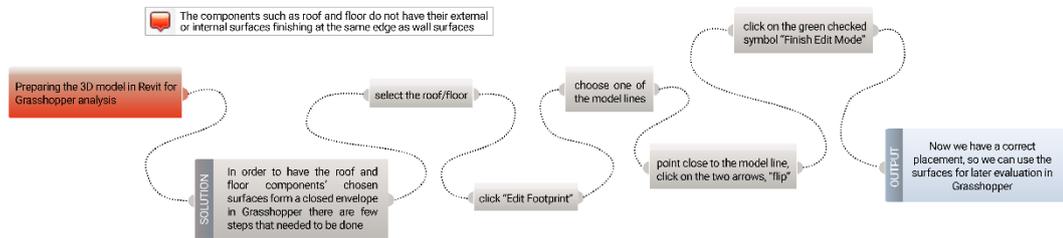


Figure 3.8 Step by step correction of roof/floor boundaries in Revit

When having a simple model in Revit this step can be done just for the aim of the analysis. The roof component (when having a flat roof) needs to be constructed within the internal boundaries of the wall components. It doesn't matter if the roof thickness is going to be extruded inside or outside the building box, but to follow the steps explained in *Figure 3.8*.

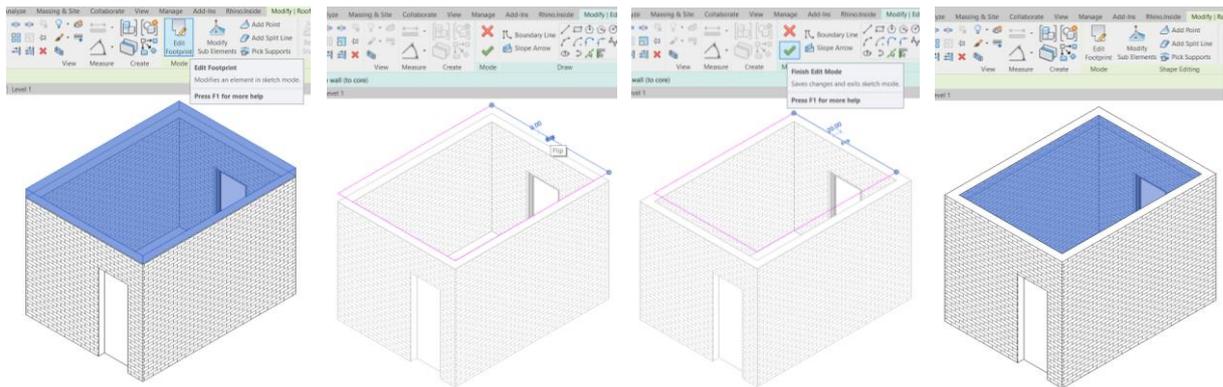


Figure 3.9 Screenshots from Revit UI, roof adjustments

As shown on *Figure 3.9*, the roof was extruded outwards, but then with the "flip" of the model lines it ended with an inwards extrusion, which will help choosing the correct surface in Grasshopper. According to daylight and glare analysis, the surfaces which are analyzed are the internal ones. This is due to getting accurate results, from the simulations done on the software program, correlated with real-time analysis. Same steps are done with the floor component (*Figure 3.10*).

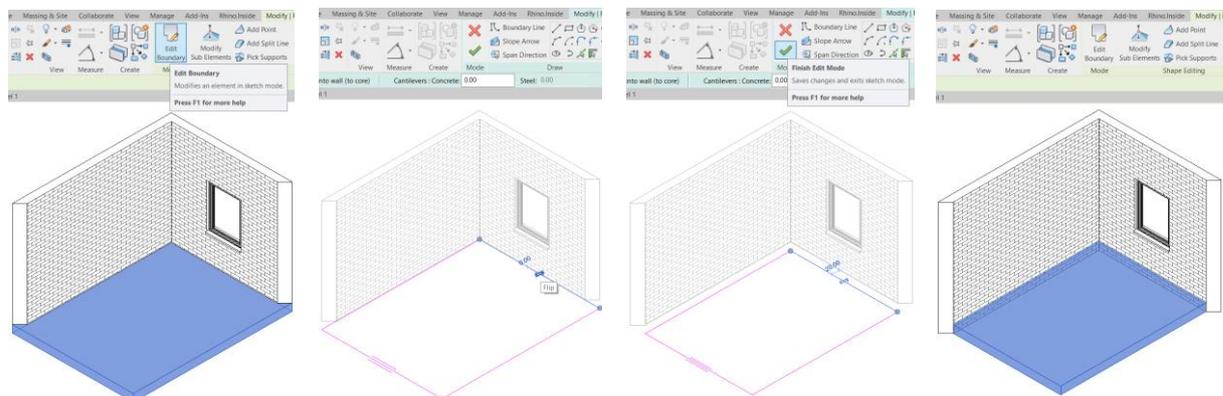


Figure 3.10 Screenshots from Revit UI, floor adjustments

The last step of Revit adjustments was material specification of the components. Using different types of layers in all components was a major demand, especially for the daylight and glare analysis. But for the single-zone model, to keep the simplicity of the model, was used only one material layer for the walls, roof and floor. *Figure 3.11* shows where to find the material editor, after a component is selected.

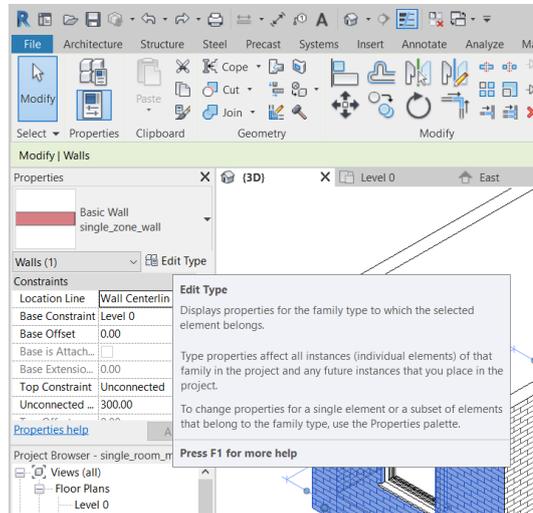


Figure 3.11 Screenshot from Revit UI, Edit Type for components

After a pop-up window is opened (*Figure 3.13*), where the "Type" and "Family" of the selected components can be read, multiple "Type Properties" are shown. Under the section "Type Parameters" there are: Construction, Graphics, Materials and Finishes, Analytical Properties and Identity Data. When clicking on "Construction" and selecting from "Structure" the "Edit..." button, another window pops up (*Figure 3.12*). Here the layers of the material can be adjusted. In this case there is only one layer of concrete and its thickness is 0.2m. It is selected as "Structure" and later more layers can be added to this as insulation, wraps, air gaps, etc.

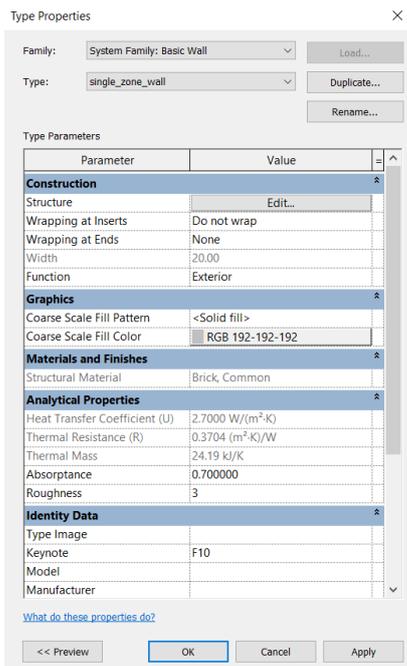


Figure 3.13 Screenshot from Revit UI, Type Properties

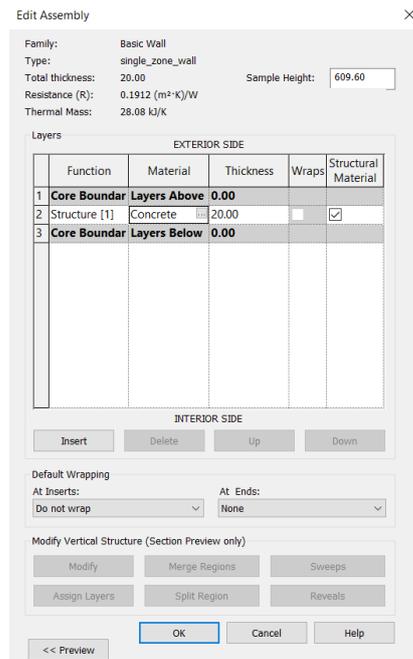
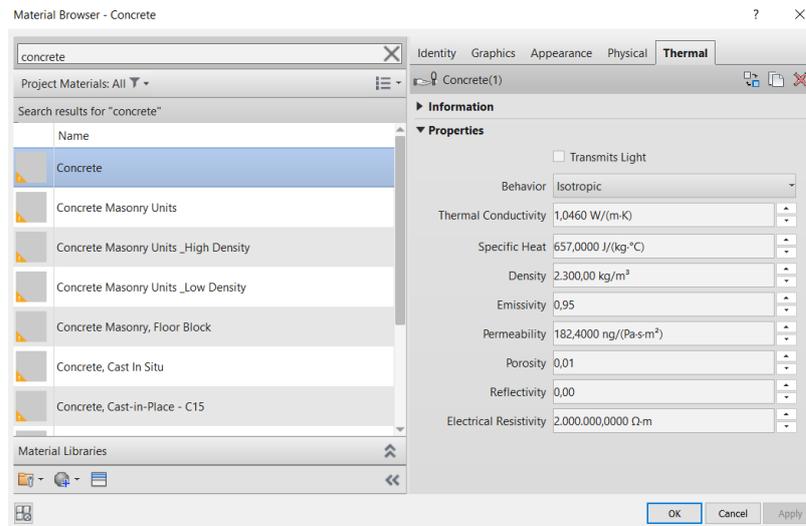


Figure 3.12 Screenshot from Revit UI, Edit Assembly

When the window "Material Browser" is opened (*Figure 3.14*) the properties of the concrete can be seen and also all the "Thermal assets" are later used for the analysis in Grasshopper.



*Figure 3.14 Screenshot from Revit UI, Type Properties*

### **3.5.3. TYPES OF COMPONENT IMPORTATION IN GRASSHOPPER**

Technically speaking, for different types of components there are different types of importation processes. After the construction of the model in Revit, with the help of Rhino Inside few of the Revit components were used within the UI of Grasshopper for model elements importation. The way in which the steps for importing elements differ is that they are selected according to their category or subcategory sorted in Revit. By this it is meant that components which are unique of one type in the model, components of the same type that are several in the model, and components that consist of multiple elements.

#### **3.5.3.1. IMPORTING BUILDING ELEMENTS BASED ON THEIR CATEGORIES (SINGLE CHOICE OF BUILDING ELEMENTS)**

The importing of Revit building elements is quite simple, using the components from the Revit tab in Grasshopper toolbar. In this first example (*Figure 3.15*), the floor was imported based on its category. The component "Built-In Categories" allows to choose between different types of building elements. Then after the elements were filtered, the element geometry was easily extracted. Just because for the further daylight analysis a closed model was needed, consisting only surfaces, the geometry was deconstructed using "Deconstruct Brep" and with "List Item" the external surface of the floor were chosen, using an index number slider. In this case for the single-zone model the same importation type for the flat roof was performed.

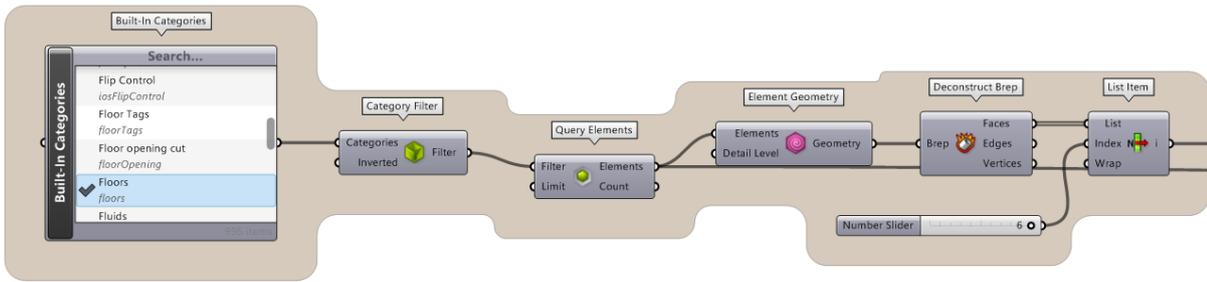


Figure 3.15 Screenshot from Grasshopper UI, component exportation type I

### 3.5.3.2. IMPORTING BUILDING ELEMENTS BASED ON THEIR CATEGORIES (MULTIPLE CHOICE OF BUILDING ELEMENTS)

The difference between this and the previous algorithm is that if there are multiple elements in the same category, the component “Value Picker” (Figure 3.16) allows you to choose which element you want to extract. This component is plugged into the output “Elements” of the component “Query Elements”. After choosing which element was wanted, in this case one of the four existing exterior walls, once again the element geometry was deconstructed and the internal surface was chosen. The “Value Picker” component needed to be copied as many times as the number of exterior walls and the same steps were made, for choosing the interior surface.

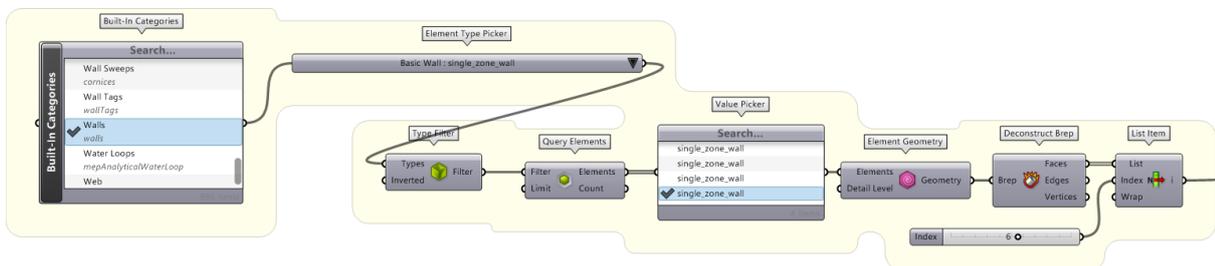


Figure 3.16 Screenshot from Grasshopper UI, component exportation type II

### 3.5.3.3. IMPORTING BUILDING ELEMENTS BASED ON THEIR CATEGORIES (MULTIPLE CHOICE OF BUILDING ELEMENTS)

Finally, the third way (Figure 3.17) of importing Revit building elements is only done if a component (ex: window, door, etc.) consists multiple elements (ex: glass, frame, door panel, etc.). It is based on choosing the building element from its category. In order to use the “Value Picker” again and choose from the subcategory list, additional component connected to the output of “Built-In Categories” is needed called “Category SubCategories”. Then with “Element Geometry By SubCategory”, which is a Python script component, it is simple to choose from the “Value Set Picker” which element you need. For example the glass was chosen and then the needed surface for the analysis was extracted.



Slider" for its Index input. Then into the component "4Point Surface", which allows to construct a surface from 4 points, all 4 points were inputted in Corner A, B, C and D. This path showed as the easiest and fastest one to model a surface within Grasshopper. A better preview of the process is shown on *Figure 3.17*. This represents the geometry preview in Rhino, more precisely the course of the steps taken with the algorithm that was made.

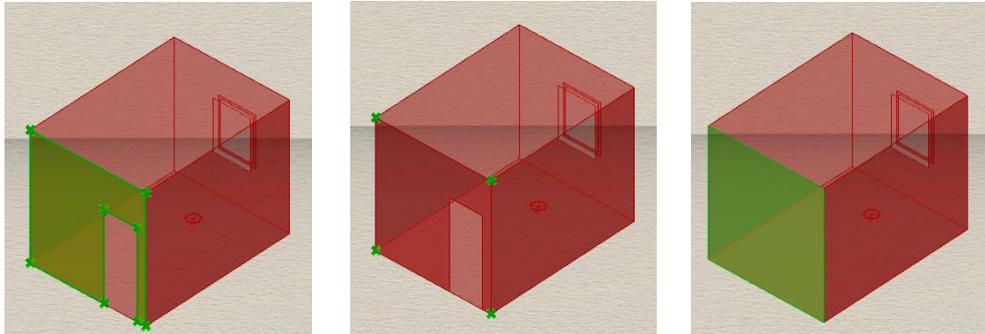


Figure 3.19 Screenshot from Rhino UI, closing a wall gap

The same way of closing the gap in the surface was used for the window aperture, but according to daylight and glare analysis there was one thing that needed to be considered. When performing the visual comfort analysis it is really important to be aware of all surfaces where the sun rays hit. Most of the widows' placement in the walls is occurring in the middle of the wall thickness. This means that the glass surface, most of the time, is not aligned with the wall surface, whether that is an internal or external surface. Although there are buildings that have a type of curtain walls or windows that are aligned with the wall surface, this case is an example of using a centrally located window placement within the thickness of the wall. Since Honeybee Model asks for closed surface boundary, this arrangement of the window elements doesn't meet those requirements and there is a modification that needs to be done within Grasshopper. This time the 4 points from the surface were the ones laying on the edges of the gap. The surface was constructed with the same algorithm used in *Figure 3.20*, just another component was added in order to move the constructed surface to the place where the glass is. This step was done because Honeybee Apertures need to lay on a surface and that surface should be the position where the glass surface is placed (*Figure 3.20*). The component "Move To Plane" with the input "Geometry" allows a chosen geometry to be moved and the input "Plane" is another surface, in this case the glass surface, where exactly it will relocate. Then the resulting surface is connected to other components, following the flow of the algorithm.

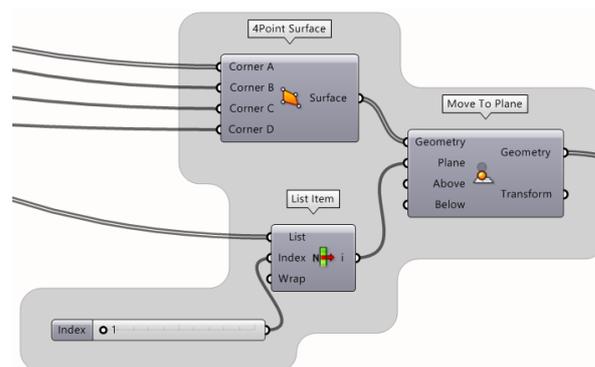
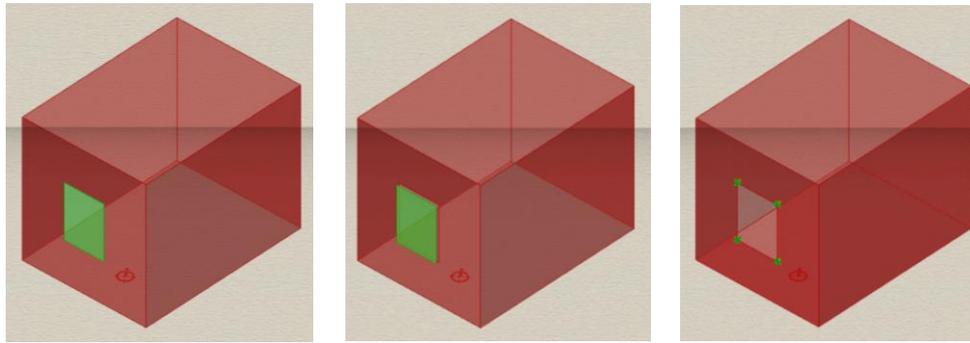


Figure 3.20 Screenshot from Grasshopper UI, moving a surface to place

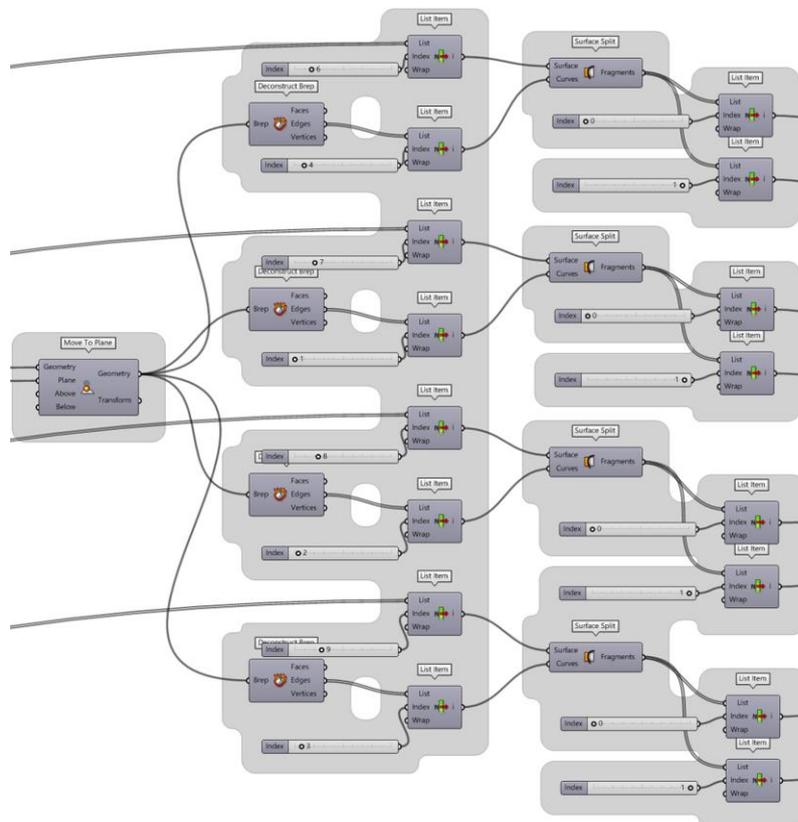
Furthermore there is a preview of the exact steps mentioned in this part of the steps, but not in Grasshopper UI, but Rhino UI. On the following *Figure 3.21* shows step by step on how all of the taken actions with the used algorithm are visualized in Rhino.



*Figure 3.21 Screenshot from Rhino UI, constructing and moving a surface to place*

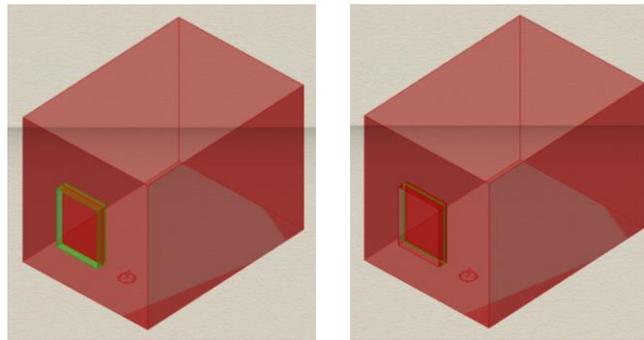
As seen on the last step of *Figure 3.21* the surface was moved, but this created new gaps on the sides. Normally in a real situation when the window is installed in a wall of a building, the constructing elements that are on both sides, bottom and top of the glass, are the sills and stiles. These elements need to be considered while performing visual analysis. This is due to the direction of the incoming sun rays and how the light can be reflected from these surfaces too.

Therefore the next algorithm of components, shown on *Figure 3.22*, was used to construct these additional elements. When importing walls as components there can be seen the sills and stiles, but they are imported as a complete surface and they are not divided according to indoor and outdoor part. Exactly this was the reason to construct them.



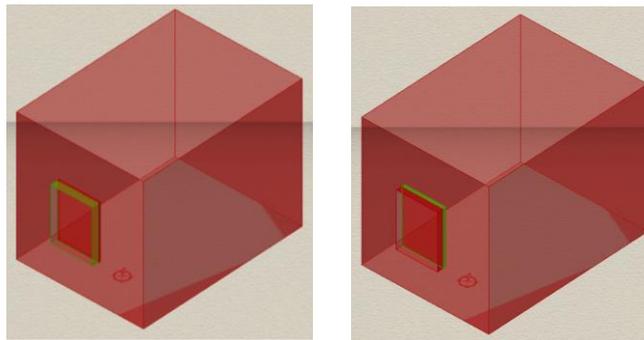
*Figure 3.22 Screenshot from Grasshopper UI, constructing sills and stiles*

The whole process of the components' algorithm used for constructing these surfaces is visualized in *Figure 3.23* within Rhino UI.



*Figure 3.23 Screenshot from Rhino UI, splitting the surface for stiles and sills*

Starting from the already imported wall surfaces, each surface that corresponds as a stile in the model using "List Item" and "Number Slider" was chosen. Then from the previously repositioned surface that closed the gap, its edges were deconstructed with "Deconstruct Brep" and an edge that directly rests on the chosen stile or sill surface was chosen. These are both plugged in a new component called "Surface Split" which allows to split a surface in pieces with a curve. The resulted fragments are two surfaces, each, in this case indoor and outdoor stiles or sills (*Figure 3.24*).



*Figure 3.24 Screenshot from Rhino UI, constructed indoor and outdoor stiles and sills*

Later the indoor stiles and sills are used for the purpose of constructing the closed surface boundary together with the other surfaces. The outdoor stiles and sills are considered as shading elements, just for the analysis.

### **3.5.5. EXTRACTING THE MATERIAL OF REVIT MODEL COMPONENTS**

Materials are an important step when doing different type of comfort analysis. For the daylight and glare analysis the most important parameters are the radiance parameters. These parameters are crucial when modifying them in Honeybee components. The values of the outputs of the simulations, could depend on the type of materials we choose for the building elements. This type of analysis, didn't need all material properties. In the following workflow there is an explanation how to import the materials and extract their properties given in Revit (*Figure 3.25*).

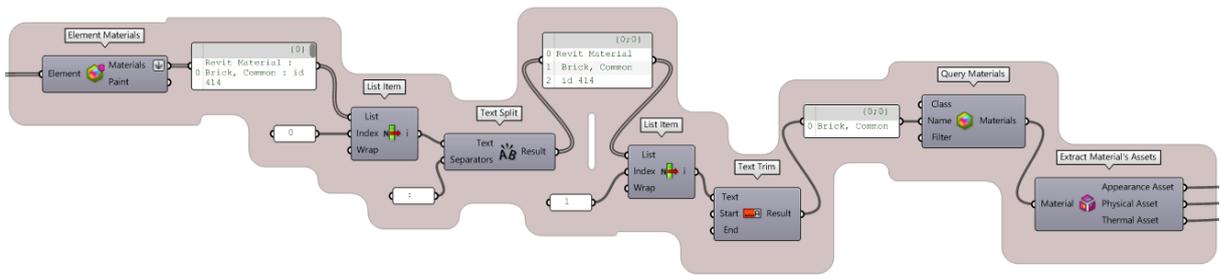


Figure 3.25 Screenshot from Grasshopper UI, importing materials from Revit to Grasshopper

It starts with a component from Revit library in Grasshopper, under Material menu, called "Element Materials". In this component the element is inputted and with a panel the Name, Type and ID can be read. If there are multiple layers in one component then this panel will show all layers that are inside one component. The second part of the workflow is mainly using few components that allow extracting the Type of selected material layer. Then with the component "Query Materials", the material document list was extracted which was connected to another component called "Extract Material's Assets". This final component is able to retrieve three types of assets.

### 3.5.5.1. ANALYZING MATERIAL ASSETS

As shown in *Figure 3.26* there are Appearance, Physical and Thermal Assets. They can be extracted and later used, with the provided components from Revit that are: Analyze Appearance Asset,

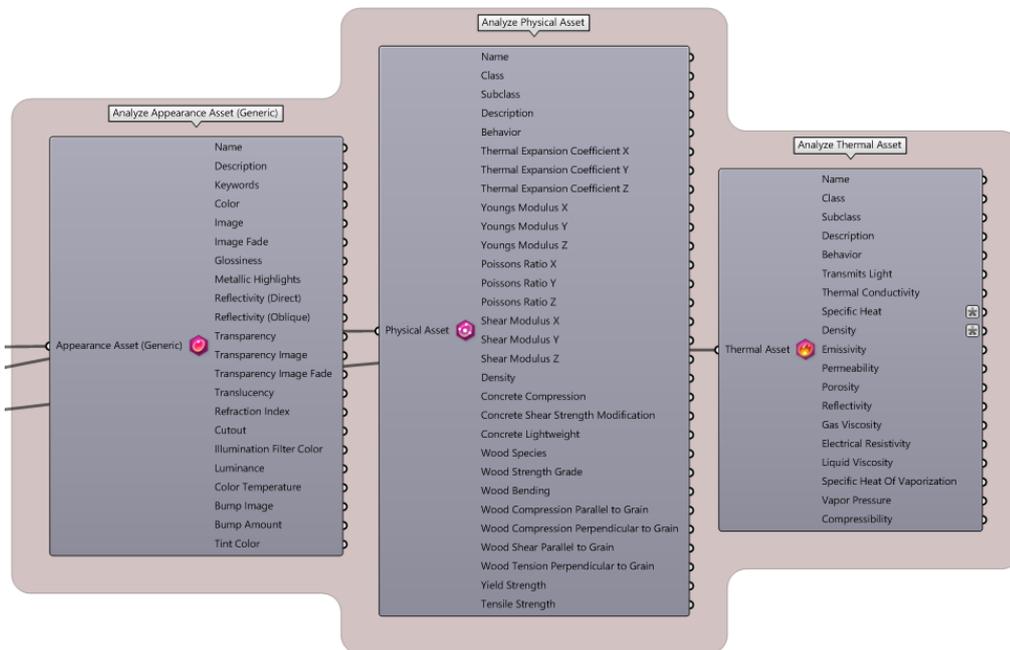


Figure 3.26 Screenshot from Grasshopper UI, material assets

Analyze Physical Asset and Analyze Thermal Asset. Multiple parameters are included in this component and it contains information about the material that can be useful if doing different types of analysis. For the purpose of the research only the Thermal Assets were used as inputs when assigning materials on Honeybee Faces.

### 3.5.6. CREATING THE HONEYBEE MODEL

This part of the methodology chapter focuses on creating the Honeybee Model and preparing it for the evaluation. As previously mentioned Honeybee has multiple groups of components that do specific analysis. For these analysis the initial rule is to have a closed surface boundary, which means to not having any surface intersect or having a gap inside the model. It is really important to check all the surfaces and see if there is an error. Usually if there is a gap the component meant for creating the Honeybee Model won't accept the constructed geometry. All geometries that exist in the building, such as walls, roof, floor, windows, doors, shading devices, etc, can be assigned with a direct Honeybee component from the menu called "Create". After every created component there is an ability to visualize it with the components under the menu "Visualize". Many others are provided within Honeybee library, also there are two separate ones that were used for the purpose of creating the Honeybee Model and simulating analysis on it, which are HB - Energy and HB - Radiance. They give an opportunity to choose between specific schedules, basic properties, modifiers, light sources, components for performing variety of analysis and then using the components from results menu, previewing the results.

#### 3.5.6.1. HONEYBEE FACES, APERTURES AND ASSIGNING MODEL PROPERTIES

- *Opaque components*

When making the Honeybee Model there are few prior steps to be made. As the algorithm flows the first action to do is to take the component "HB Face" and plug the node from a surface into the "geo" input (*Figure 3.27*).

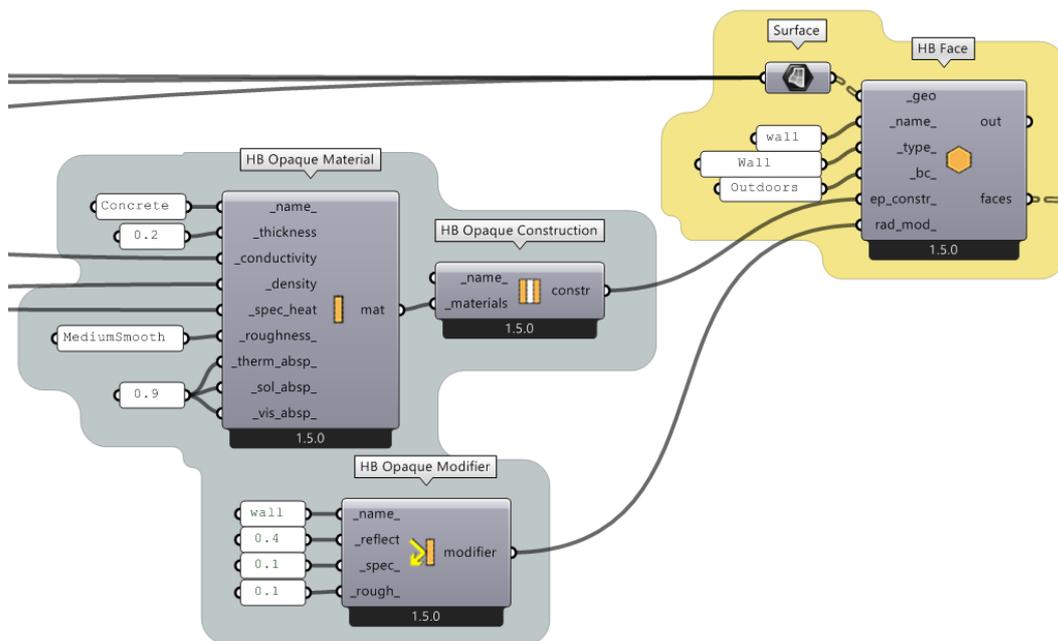


Figure 3.27 Screenshot from Grasshopper UI, HB Face and surface properties

The other inputs are for the name of the surface (ex: roof). If they are not specified using the inputs below, the face type will be used to set the surface's material and construction. On the basis of the Face's typical direction, the default is automatically selected (up being RoofCeiling, down being Floor and vertically-oriented being Wall). Additionally, default building materials and constructions are assigned using the boundary condition. It offers the options of Ground, Outdoors, and Adiabatic. Next is the input "ep\_construction" which can also be a custom opaque construction object, modified through the component "HB Opaque Construction". Here in this part an opaque material was constructed with the help of "HB Opaque Material" which had the inputs plugged in from the previously mentioned Thermal Assets extracted from Revit. As *Figure 3.28* shows, the Thermal Assets are taken from the extracted element material properties, but there is a slight change in the units of the parameters.

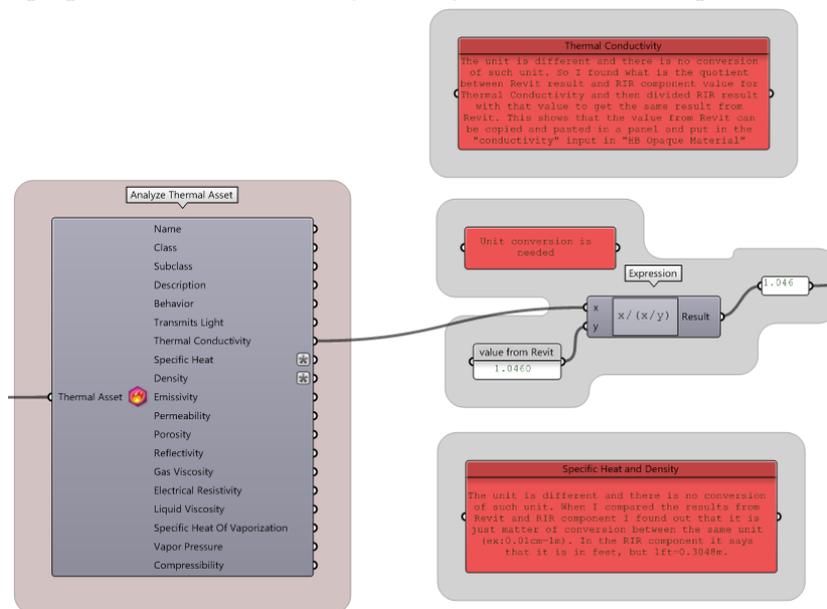


Figure 3.28 Screenshot from Grasshopper UI, material properties

In Revit the project units can be changed, so for this project they were assigned in metric. But the Honeybee component for material assets has a different type of units that need to be converted. This is a step that can take a lot of time, due to unit conversion of every parameter. Also another thing is that it was needed to be found which is the correct conversion equation for the corresponding unit and that took me a lot of time. The solution for this is either to convert the units or write the values of the needed parameters, from Revit material editor, as a text in a panel and input them in the desired components. For all the opaque surfaces, in this case walls, roof and floor, the same algorithm is used to construct a HB Face. Also, in this part of the research the indoor stiles and sills needed to be considered as HB Faces, due to the same material and their role as surfaces who make the closed surface boundary.

- *Transparent components*

Next was creating a HB Aperture, which presents all window openings. In this case study there is only one window and the glass surface was assigned as a face to the "HB Aperture". The component asks for the name of the aperture, if it is operable or not ("Boolean Toggle: True" was inputted) and the material properties as well as the radiance modifier. Here the radiance modifier was connected to "HB Glass Modifier 3" which allows to create it according to the red, green and blue transmittance. The

parameters that were entered was a 0.7 for all of them. Lastly the index of refraction was set to 1.52 for float glass. Overall the glass performances resulted with Solar Facto  $g=86\%$  and Thermal Transmittance of  $U_g=5.8$ . For the ep\_construction again the material properties were used from the Thermal Assets of the glass element from the window component from Revit. The used component was "HB Window Construction" where as an input is "HB Glass Material" and all properties of the glass were plugged in. One problem that occurred was the same problem as with the other material properties for the opaque components. The values got different units or some of the inputs that were asked from "HB Glass Material", weren't provided within the Revit component. The solution to this was using the default parameters or using another software to calculate them. All of this is shown as an algorithm of Honeybee components on *Figure 3.29*.

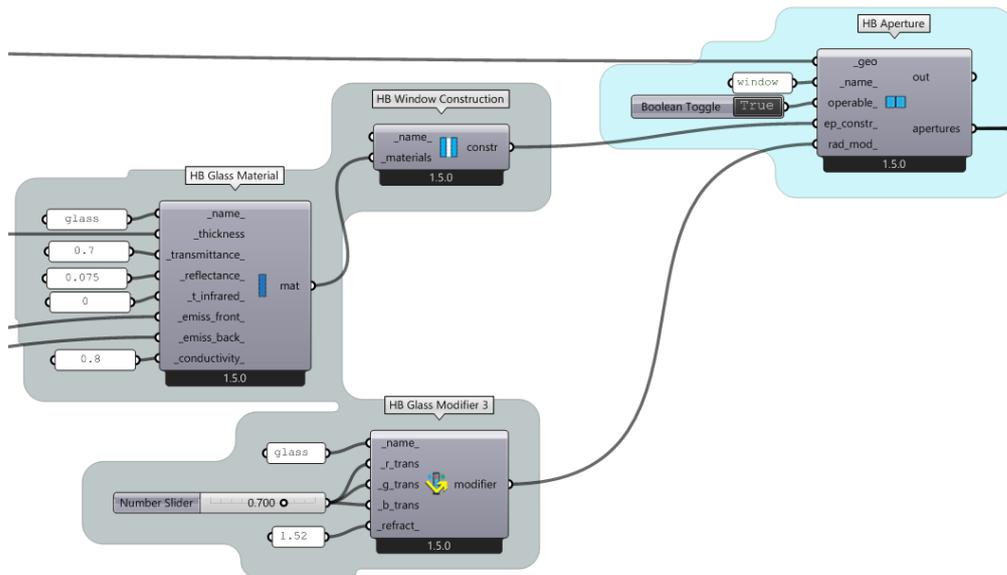


Figure 3.29 Screenshot from Grasshopper UI, HB Aperture and surface properties

### 3.5.6.2. HONEYBEE ROOM

Creating the HB Room is the second to last step of constructing the HB Model. In this step all of the HB Face components needed to be inputted in "faces". The other inputs that were plugged in are the name, program and whether the room is conditioned or not, in this case "Boolean Toggle: False" was used. A program for occupancy, activity and lighting was constructed with the components "HB ProgramType", where with "HB People" and "HB Lighting" the properties of it were adjusted. The schedules for occupancy and activity were chosen for a small office from the component "HB Building Programs". Also parameters for the lighting were put and the schedule was controlled with "HB Daylight Control Schedule". The component "HB Room" will be active and working if the analytical model doesn't have any gaps and all the data is correctly inputted. This follows the rule mentioned above, which is about the closed surface boundary. With all elements such as walls, roof, floor and indoor stiles and sills, a HB Room was successfully created (*Figure 3.30*), which will later be evaluated and used into the

HB Model. In this case study there was only one room, which was provided for the daylight and glare analysis. All the inputs until now were adjusted and were specifically chosen according to the requirements for calculating visual comfort KPIs. After this step the room can be visualized with the component "HB Visualize All" in order to see the prepared model.

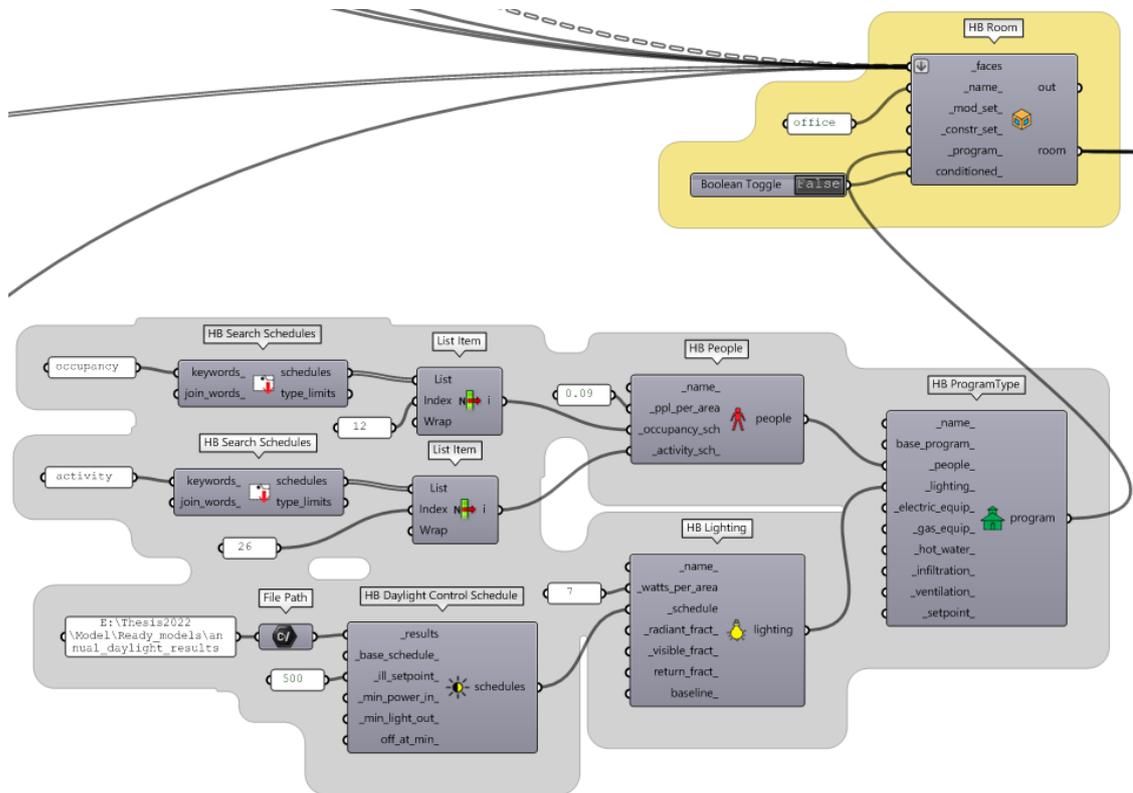


Figure 3.30 Screenshot from Grasshopper UI, HB Room and Program Type

### 3.5.6.3. ASSIGNING OUTDOOR STILES AND SILLS AS STATIC SHADING ELEMENT

There are three ways of applying shades. It can be done with modeled shades in Revit and then exporting them in Grasshopper, assigned a surface from Grasshopper or creating them with Honeybee components for shading elements. Both needed to be correctly modeled in order to fit the rules for Honeybee analysis. The first example is if there are shading device elements that were previously modeled in Revit. They can be selected through the same algorithm used for the importation of the other components (ex: walls, roof, floor, etc).

However, in this case study there was a use of different types of outdoor shading device which were constructed in Grasshopper. The first type were the outdoor stiles and sills, which were assigned as a "HB Shade" (Figure 3.31). With this type of shades Honeybee provides components in order to adjust their parameters. As an input for the "HB Shade" the used plugged components were "HB Shade Construction" and "HB Opaque Modifier". This was due to the fact that within these components the parameters of reflectance, specularity and roughness can be adapted for the specific shading element.

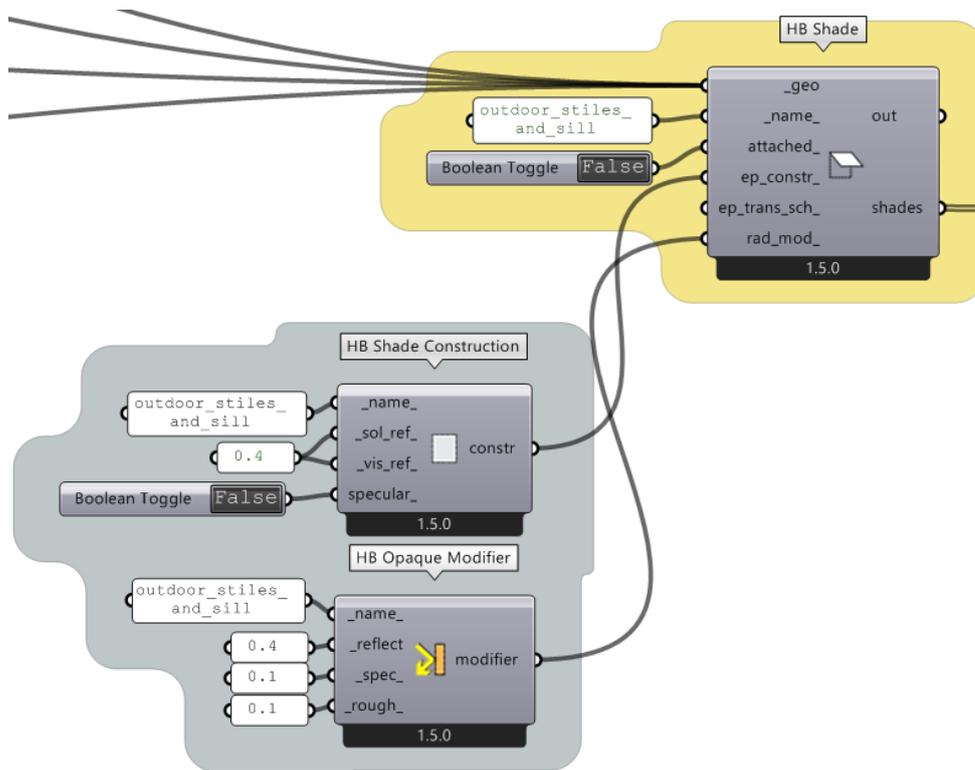


Figure 3.31 Screenshot from Grasshopper UI, outdoor stiles and sills assigned as static shades in Grasshopper

With this type of shades Honeybee provides components in order to adjust their parameters. As an input for the "HB Shade" the used plugged components were "HB Shade Construction" and "HB Opaque Modifier". This was due to the fact that within these components the parameters of reflectance, specularity and roughness can be adapted for the specific shading element.

After this the last step before plugging all surfaces in HB Model is the step where a component called "HB Add Subface", where as subfaces the apertures are added and as hb\_obj is the HB Room. Finally I used the component "HB Add Shade", added the node from already constructed static shades as outdoor shades and plugged this component into "HB Model" (Figure 3.32).

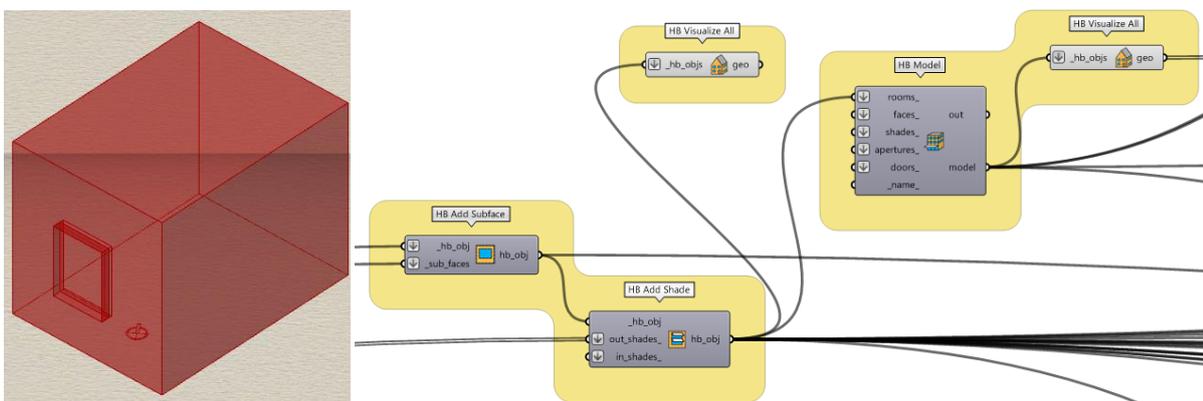


Figure 3.32 Screenshot from Rhino and Grasshopper UI, shades application

It is important to mention that this is the first type of prepared model for simulations, which means that later when the daylight and glare analysis were performed, the comparison between results will be done on the model with no blinds (the one explained above) and with dynamic shading (later explained).

### 3.5.6.4. INDOOR DYNAMIC SHADING DEVICE

Exploring in Revit about dynamic shading, it was found that it's not possible to give a schedule for the shading device, if it is constructed there. There is only ability of modeling the geometry in Revit and then import it in Grasshopper. Although in both case studies few types of workflows were tested, on how to apply dynamic shading devices, not all of them succeeded with the desired results. The reason will be better explained later. For the single-zone model 2 types of curtain were used, with different transparency, which have a role of indoor blinds. The steps to create them was basically choosing a surface in Grasshopper and then assigning it as a "HB Shade" with modifiers for the construction and radiance parameters (*Figure 3.33*).

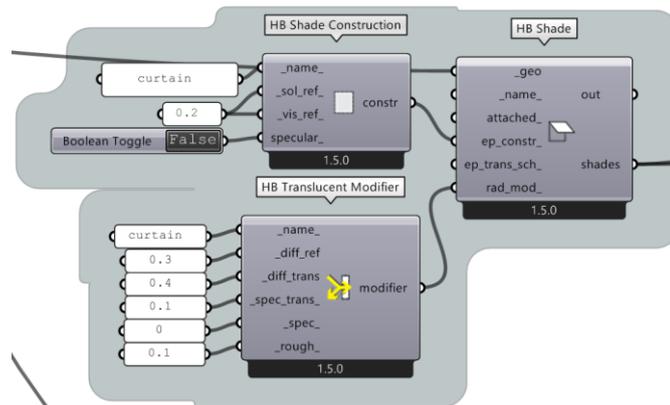


Figure 3.33 Screenshot from Grasshopper UI, shades

In order to automatically move the curtains, test simulations for Daylight Glare Probability (DGP) were performed. The idea was to see what the Glare Index is in a given point in the space while using one of the two curtains and without a curtain. Three simulations were ran and the first step of the workflow to construct the algorithm is shown in *Figure 3.34*, where the HB Model was created. This means that as outdoor shades the outdoor sills and stiles were plugged in and for the indoor shades the simulation was ran once with the first type of curtain and then with the second type.

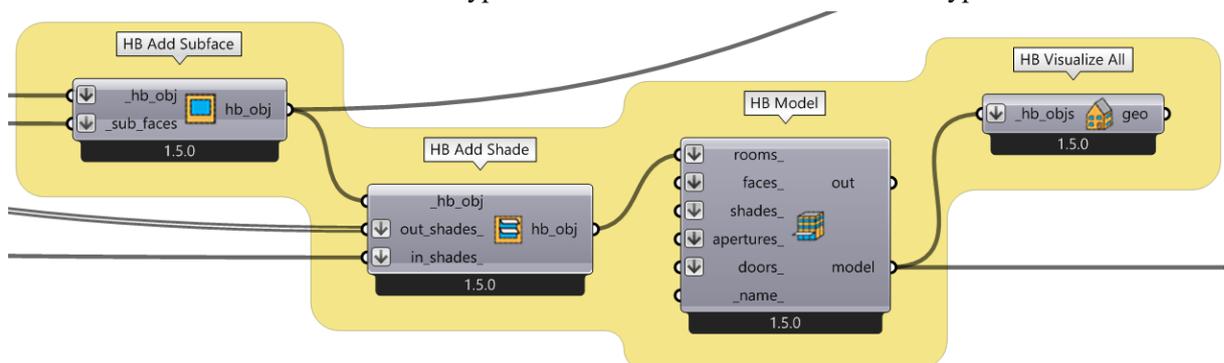


Figure 3.34 Screenshot from Grasshopper UI, test simulations for dynamic shading

For all three tests different modifiers were used. *Figure 3.35* shows the 2 types of modifiers (since the first test was done where the window has no blind). Here for first curtain there is a higher transmission of sun rays due to its material which has a higher light transmittance, higher diffuse and specular transmittance. The second type of curtain is less transparent and has different values which are lower than the first type of curtain.

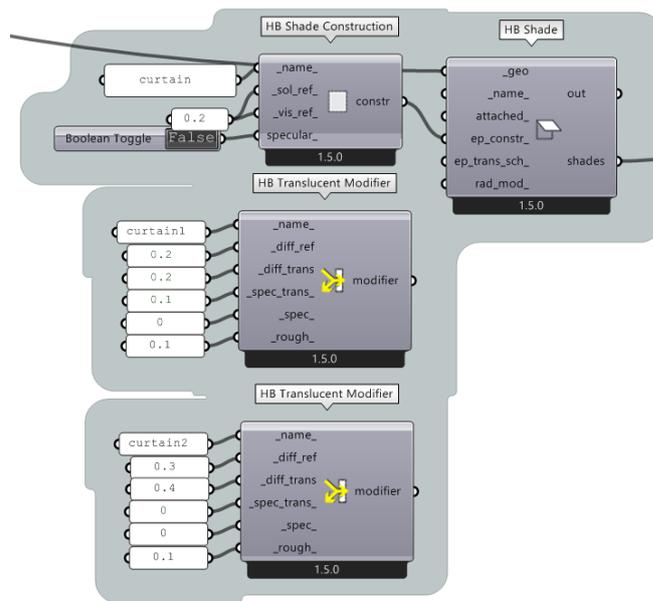


Figure 3.36 Screenshot from Grasshopper UI, test simulations translucent modifier

DGP simulations are made based on a grid which has to be assigned from the floors in HB Room with selected components. The first component to use is "HB Assign Grid and Views" which has an input of grids and views. In this case the "HB SensorGrid from Rooms" was inputted into grids. The grid size was set to 10, the distance from the floor is 90cm and wall offset of 10cm. The point which was used was positioned somewhere near the window (Figure 3.36).

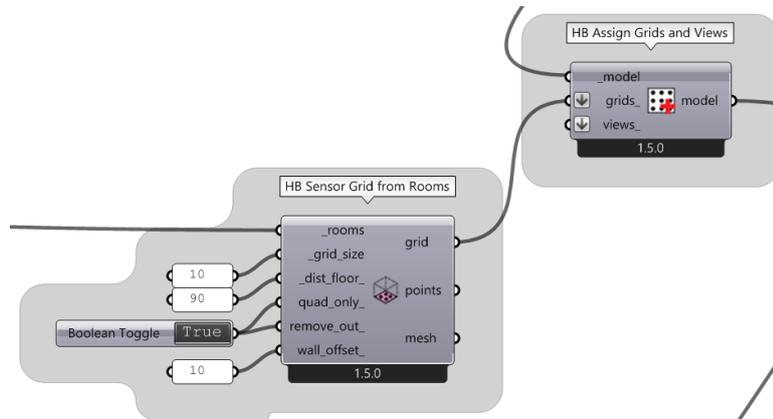


Figure 3.35 Screenshot from Grasshopper UI, test simulations grid

The next step was connecting the last component with the "HB Imageless Annual Glare" component. This component is used to get the hourly Daylight Glare Probability (DGP) for each sensor in a model's sensor grids, an annual glare study for a Honeybee model. Calculating Glare Autonomy is done using the generated DGP (GA). This represents the proportion of occupied time when a view is glare-free. Also an .epw file from Turin was used as a weather file and the north rotation of 35 degrees (Figure 3.37).

When the simulations were ran the results were connected to a new component called "HB Annual Results to Data" which is used to gather all the results into a data file. To extract the values from this data "LB Deconstruct Data" was used and then with "Split Tree" the results from just one point were chosen (Figure 3.38).

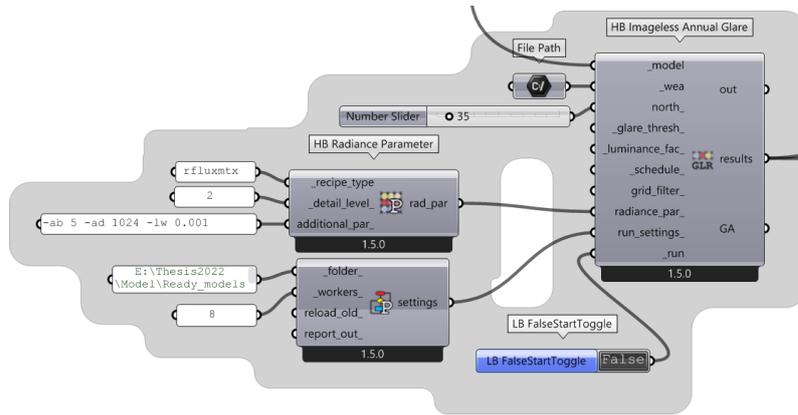


Figure 3.38 Screenshot from Grasshopper UI, test simulations DGP

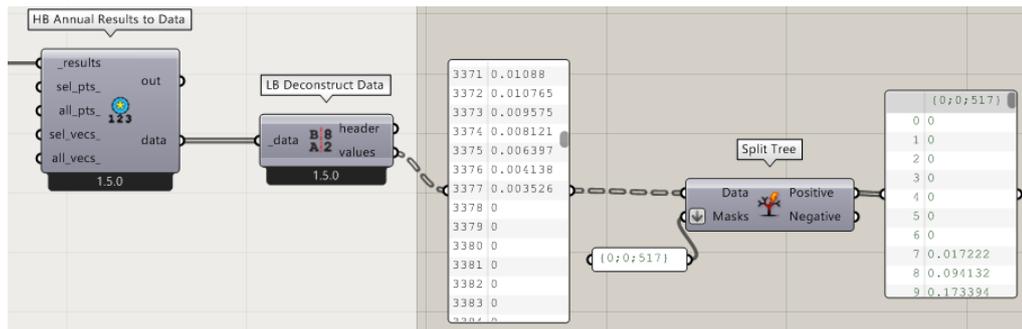


Figure 3.37 Screenshot from Grasshopper UI, test results DGP

The values were previewed in a panel and they were 8760 values in total. This means that the DGP was calculated for each hour in one year for one point. These values needed to be extracted in a file and later used to create the schedule. In order to save a file with all values there is an option to transfer the data with the "Stream Content" button, which is in the falling menu when clicking right click on the panel with values. Then the content was streamed into a folder and the file saved was a .txt file. Later this file was opened in Excel and the same algorithm was performed for the other two test simulations. When all the information was gathered I used few formulas on Excel to transform the values into 3 specified values that were from the type of transparency of the curtains. For example the first value was 1, which means the transmittance is with the highest value. For the first more transparent curtain the transmittance was 0.8 and with the lowest transmittance of 0.5 was the second curtain. The retrieved results are shown in *Table 3.1*.

	1	0.8	0.5
1	1		
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0.123665	0.019958	0.008254
10	0.207823	0.138345	0.032295
11	0.239989	0.25409	0.124583
12	0.255612	0.268315	0.157284
13	0.235653	0.166789	0.092546
14	0.224429	0.129435	0.048046
15	0.215743	0.065614	0.022381
16	0.009368	0.004231	0.003667
17	0	0	0

Table 3.1 Excel results from test simulations of DGP

Next was the process of selecting the values that are the highest lower values from 0.4, because a value of  $DGP > 0.4$  is considered as disturbing and intolerable. After this all values were supposed to be found in which column they belong. The columns were 1, 2 and 3. Then all of the values were changed with 1, 2 and 3, which means that  $1=1$ ,  $2=0.8$  and  $3=0.5$  (Table 3.2).

Finally the last column has values that represent the ultimate list, which represents the schedule for the transmittance of the indoor blinds. This schedule consists data for each hour, which curtain should be used based on the transmittance.

DGP HIGHEST VALUE LOWER THAN 0,40	VALUE LOCATION IN COLUMN	SCHEDULE WITH TRANSMITTANCE VALUES
0	1	1
0	1	1
0	1	1
0	1	1
0	1	1
0	1	1
0	1	1
0	1	1
0.123665	1	1
0.207823	1	1
0.25409	2	0.8
0.268315	2	0.8
0.235653	1	1
0.224429	1	1
0.215743	1	1
0.009368	1	1
0	1	1

Table 3.2 Excel results from test simulations of DGP (prepared schedule)

Ultimate step before finalizing the HB Model was done considering the shades. The shades schedule was applied to a component which is provided from the Honeybee Schedules menu. This component is called "HB Fixed Interval Schedule" and it was used to create a schedule from a list of values at a fixed interval or timestep, according to the given simulation period of whole year. This was plugged in the input of "HB Apply Shade Schedule", which got the input for "hb\_obj" from the last modified analytical model with the outdoor stiles and sills (Figure 3.39). After this the HB Model is ready to be used for daylight and glare simulations (Figure 3.40).

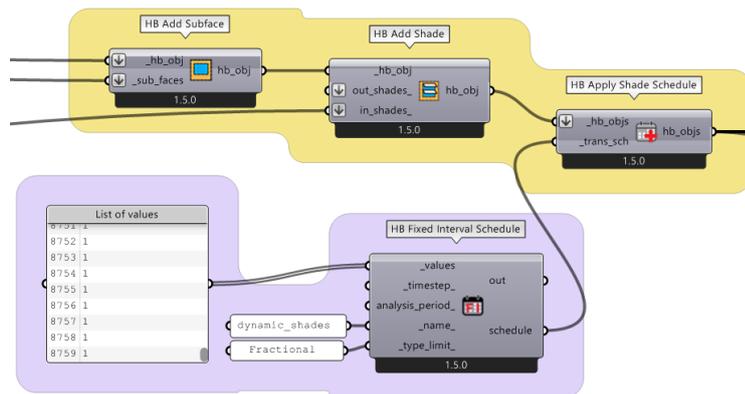


Figure 3.39 Screenshot from Grasshopper UI, final input for HB Model, dynamic shading schedule

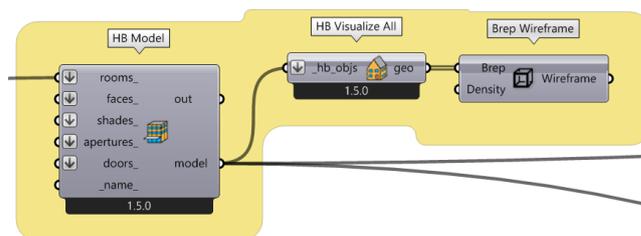


Figure 3.40 Screenshot from Grasshopper UI, final HB Model with dynamic shading

### **3.5.7. DAYLIGHT SIMULATIONS**

With the aim of performing daylight analysis, using Honeybee and Ladybug components, simulations were carried out for the building envelope of the first case study. Because of the simplicity of the model it was possible to simulate the analysis with not a lot of modifications and simple parameters. Two different approaches were considered while running the simulations. In order to see how the dynamic shades will work, the first test simulation was done without any daylight obstacle, rather than the geometry of outdoor and indoor stiles and sills and the glass itself. And the second one using the curtain blinds with a dynamic schedule applied to the transmittance of light. For this evaluation of building envelope performance, few KPIs were selected. They were chosen due to the fact that in Honeybee and Ladybug here are components that directly perform the simulations for multiple daylight analysis. Daylight Factor (DF), Daylight Autonomy (DA), Spatial Daylight Autonomy (sDA), Continuous Daylight Autonomy (cDA), Useful Daylight Illuminance (UDI) and Annual Sunlight Exposure (ASE) were the performance indicators that were tested for the single-zone model. In both cases, no shading and dynamic shading, the workflow after completing the HB Model was almost the same. The only thing that was different was the time that the analysis took to be performed by the Radiance engine, but that will be explained later in the discussion chapter. In the next few pages a detailed algorithm will be explained on which components are used for the analysis, what inputs they need and what outputs they produce. Also the process of visualizing the results is shown, with the use of Ladybug components. The preview of the legends produced from these components will be shown in the results chapter.

#### **3.5.7.1. SIMULATION ALGORITHM OF COMPONENTS ACCORDING TO DAYLIGHT**

To start with (*Figure 3.41*), a component "HB Daylight Factor" which directly simulates an analysis of the daylight factor of a HB Model. It basically evaluates the ratio of the indoor illuminance to outdoor illuminance with an unobstructed overcast sky. Default radiance parameters were used and for the specific settings on how the recipe should be run I used "HB Recipe Settings" which saves the results in a selected folder and gives an opportunity to select the number of workers (CPUs) in the execution of the recipe. Next to run the simulation a "Boolean Toggle: True" was inputted in "run".

The results output was connected to a "LB Spatial Heatmap" in order to visualize the results with provided colors and parameters, in this case from 0-10 for the Daylight Factor expressed in percentage. "LB Legend Parameters" component was used to input the above mentioned modifications for the outcome of the legend. For the mesh "HB Sensor Grid from Rooms" component needed to be assigned in order to input the HB Room, grid size, distance from floor and wall offset.

Lastly an "Average" component was used to calculate the average Daylight Factor from the previously given results.

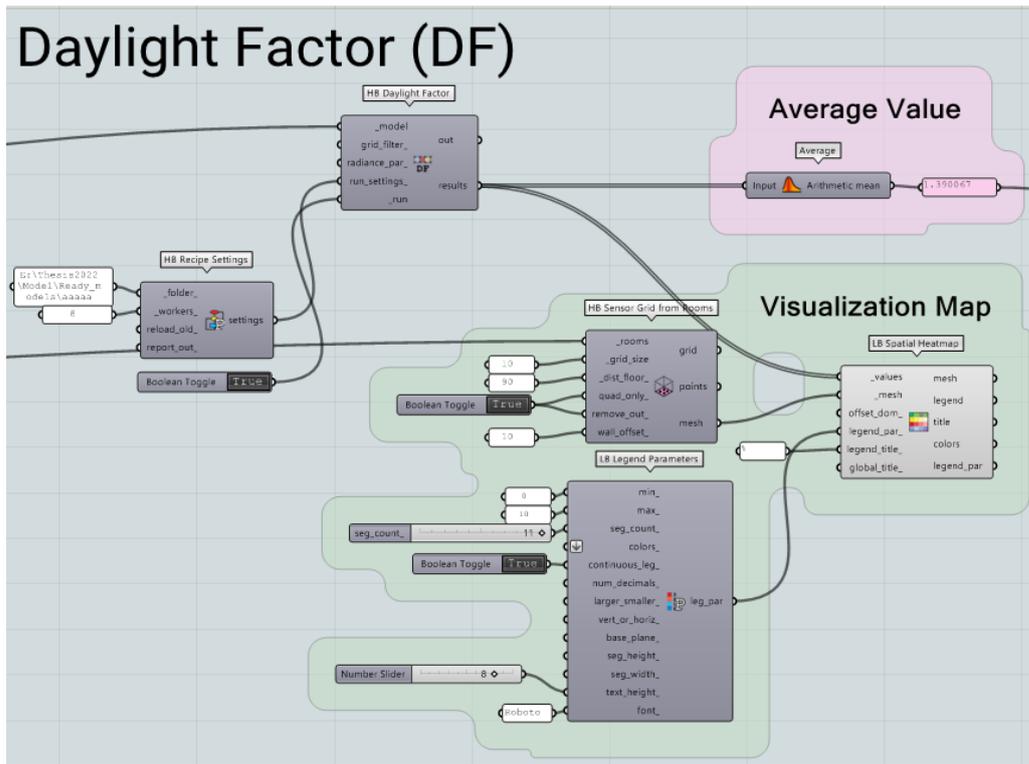


Figure 3.41 Screenshot from Grasshopper UI, Daylight Factor simulation algorithm

Figure 3.42 shows the next KPIs which were evaluated with only one component from Honeybee. The component used for this was "HB Annual Daylight" which allows to run simulations for evaluating few daylight performance indicators. It needed the following inputs: model (prepared HB model), .wea or .epw file (used: .epw), north (rotated: 35°), inputs which were set to default (threshold, schedule, grid filter) radiance parameters, run settings (CPU: 8) and run (Boolean Toggle: True).

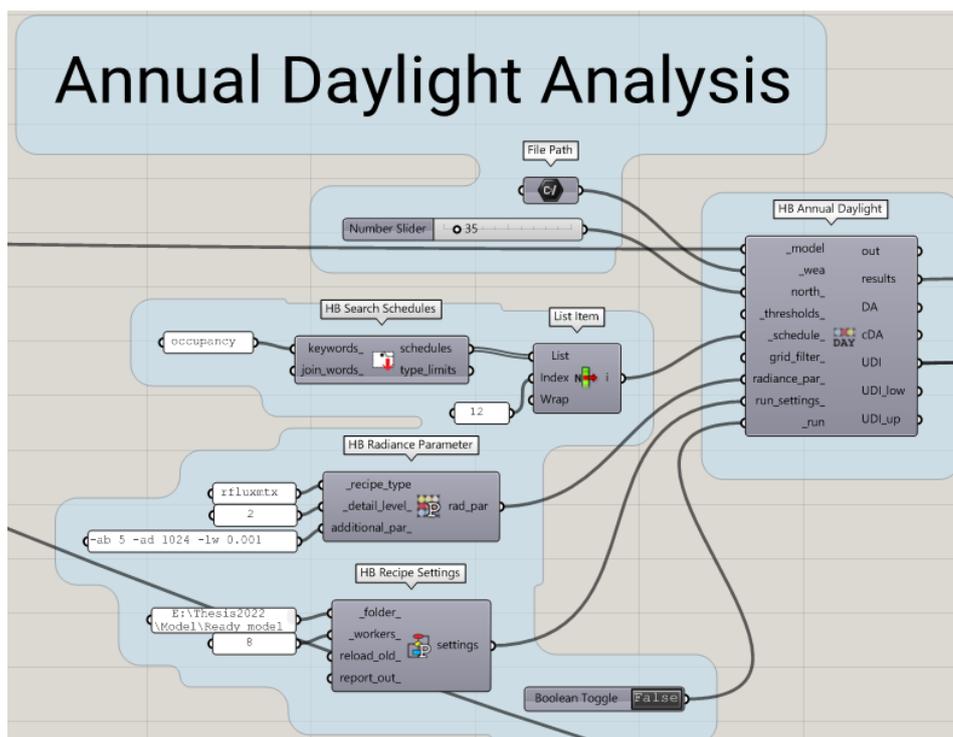


Figure 3.42 Screenshot from Grasshopper UI, Annual Daylight simulation algorithm

Next step (Figure 3.43) was connecting the results output to "HB Annual Daylight Metrics" which allowed me to choose an occupancy schedule with the "HB Search Schedules", where the schedule is a normal schedule from 8am to 6pm. This was selected with using a "List Item" for choosing the desired schedule from the list. Other inputs that can be provided to "HB Annual Daylight Metrics" are min and max illuminance thresholds for useful daylight illuminance in lux (Default: (100,3000)). But in this specific case a threshold for daylight autonomy was needed and provided as 500 lux.

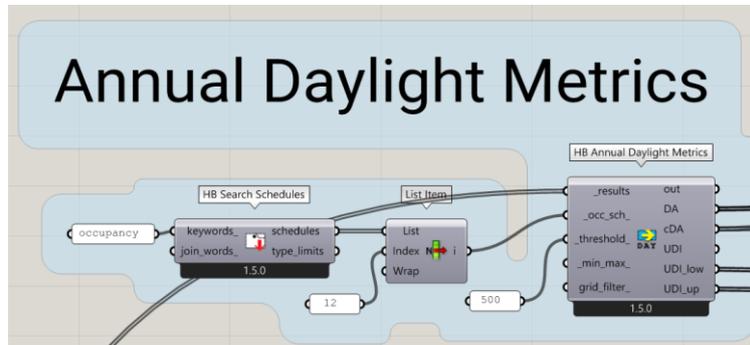


Figure 3.43 Screenshot from Grasshopper UI, Annual Daylight Metrics occupancy schedule

The outputs that this component produces are results from the Daylight Autonomy (DA), Continuous Daylight Autonomy (cDA) and Useful Daylight Illuminance (UDI), which were mentioned above, as key performance indicators for daylight analysis. The additional indicators are UDI fell short [0–100 lux] insufficient illumination, UDI supplementary [100–500 lux] integrate with electric light, UDI autonomous [500–3000 lux] no electric light required and UDI exceeded [>3000 lux] bad situation: glare, overheating, etc., which had results for the percentage of time that illuminance is between the minimum and maximum threshold for useful daylight illuminance, assigned in the “\_min\_max\_” input of the component “HB Annual Daylight Metrics”. Later these outputs consisting the results of the performed simulations, one by one were inputted in other components for visualization of the results (Figure 3.44).

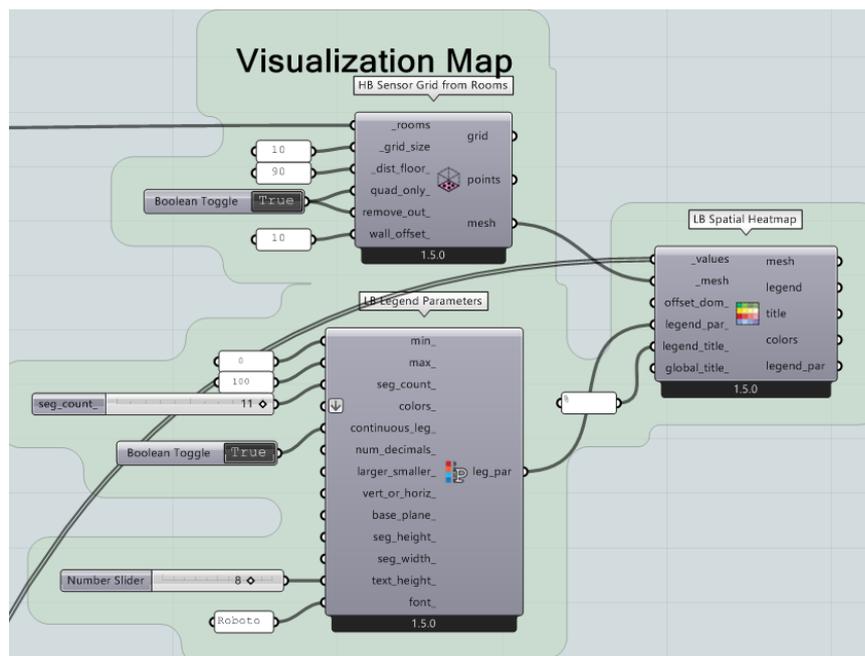


Figure 3.44 Screenshot from Grasshopper UI, Visualization of simulation results for daylight

Again for the visualization Ladybug tool's components were used. Same as in Daylight Factor, the "LB Spatial Heatmap" had its values input connected with a node to the variety of results from the previously performed simulations. All of them were visualized with the modification of the legend parameters using "LB Legend Parameters" and for the min and max value, the range was from 0 to 100, expressed in percentage. This was because all of the indicators are calculated to evaluate an area where the percentage represents time.

Only for the Spatial Daylight Autonomy (sDA) an additional step was taken. The component used for calculating this indicator was provided by Honeybee library too. This is an added process to the previous simulation for the other daylight metrics. In order to have this component perform the desired simulations, an input for Daylight Autonomy and mesh was needed. The Daylight Autonomy results were taken from the previously used component "HB Annual Daylight Metrics", and the mesh was constructed using "HB Sensor Grid from Rooms" with the same adjustments as for the others (Figure 3.45).

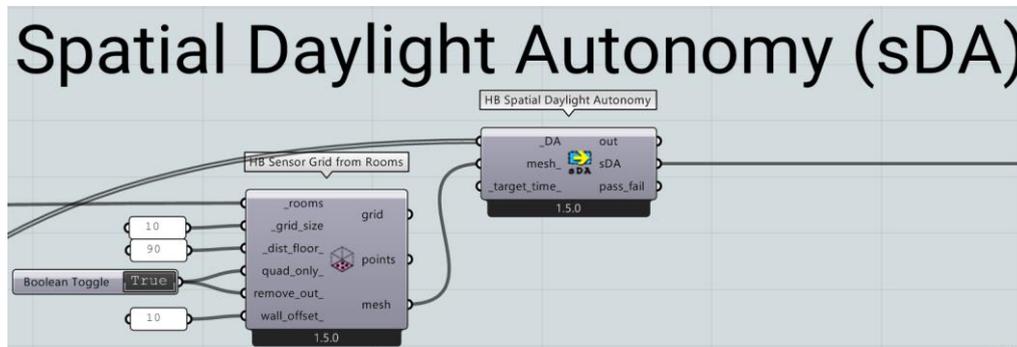


Figure 3.45 Screenshot from Grasshopper UI, Spatial Daylight Autonomy (sDA) simulation

The last simulation which was done for only no blinds situation of single-zone model, was the Annual Sunlight Exposure (ASE). This simulation was performed with a help of newly constructed algorithm using some of the same components from previous examples. This simulation uses "HB Direct Sun Hours", component which needs input of the model (prepared HB model), .wea or .epw file (used: .epw), north 35°, grid filter, run settings (CPU: 8) and run (Boolean Toggle: True). Then the results that I got from this component, were connected to the "HB Annual Daylight Metrics", where the data from the occupancy schedule was collected with the component "HB Schedule to Data". In order to perform the this type of analysis for the ASE, this component needed to have the input for threshold as 1000, which means that it will evaluate the area which has 1000lux. Here the analysis period was modified with the "LB Analysis Period" and it was set for a whole year (Figure 3.46).

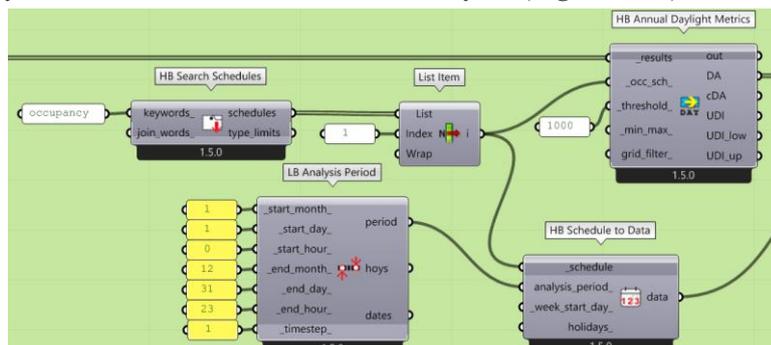


Figure 3.46 Screenshot from Grasshopper UI, Annual Sunlight Exposure simulation, schedule and analysis period

The values for Daylight Autonomy that were an output from "HB Annual Daylight Metrics" and the data from the occupancy schedule were then used with an expression (Figure 3.47).

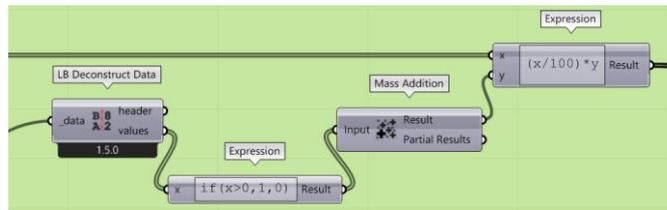


Figure 3.47 Screenshot from Grasshopper UI, Annual Sunlight Exposure calculation from results

The specified expression used two inputs: x (results from Daylight Autonomy) and y (deconstructed data, in which each value higher than 0 points out that there is an occupant in the room and this was transformed with an IF function to 1 and 0, for true and false, and the total number of existent 1s was then used as multiplication in the expression). This will give us the result of occupancy hours in which the DA is reaching the threshold. This result can be used with the "LB Spatial Heatmap" to visualize the results, but the most important part is that when adjusting the parameters of the legend in "LB Legend Parameters" the min value for the hours needs to be 250. Also the mesh needs to be inputted as this will provide the number of faces or vertices which will match the number of values in order to later color them for the visualized preview of the results (Figure 3.48).

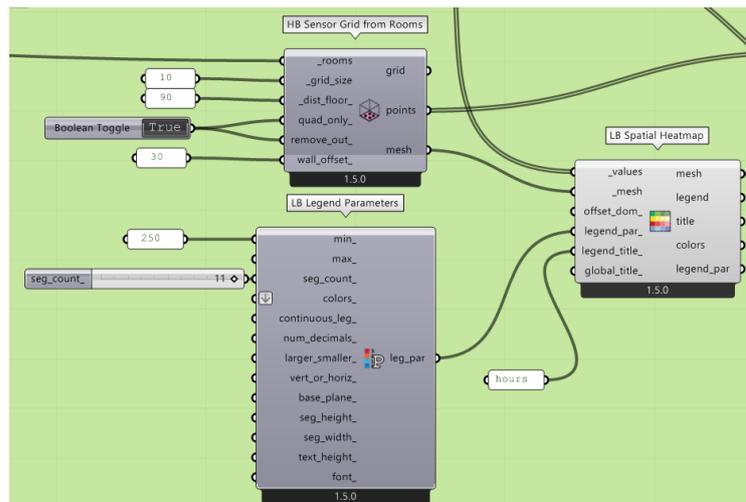


Figure 3.48 Screenshot from Grasshopper UI, Annual Sunlight Exposure visualization

Additional step was made in order to have the result as a text. After getting the results from the expression mentioned previously, where the total number of occupancy hours that DA is 1000lux was gotten, new expression was made in order to select the points where DA is 1000lux more than 250 occupancy hours.

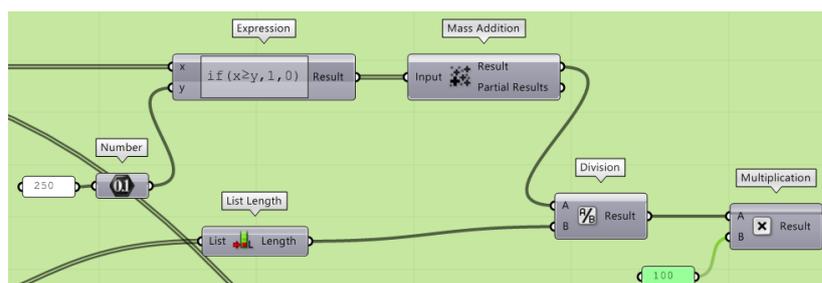


Figure 3.49 Screenshot from Grasshopper UI, Annual Sunlight Exposure visualization

This was then divided by the number of points which was derived from the component "HB Sensor Grid of Rooms" with a component called "List Length" which measures the number of values in a list. Then I multiplied the result of the division by 100, so I can get the percentage of an area (*Figure 3.49*).

This part rounds all the steps taken in order to simulate the daylight analysis in both scenarios, no blinds and with dynamic shading device. Some of the simulations were ran multiple times in order to simulate several algorithms.

### 3.5.8. GLARE SIMULATIONS

Evaluating glare within the first case study was the initial understanding of how these algorithms can perform simulations, with given results, that later can be used back into the model properties, for example for creating a schedule for the dynamic shading device. This particular part of the research was explained in the segment with the explanation of creating the shading elements. Those glare simulations that were done for the need of a schedule construction, were later used for the final HB Model evaluation. This means that some of the components of calculating glare parameters were initially introduced with the construction of curtain blinds which are dynamic. However for the single-zone model there were three types of performed simulations regarding glare analysis. The first one being Daylight Glare Probability (DGP), Glare Autonomy (GA) and Spatial Glare Autonomy (sGA). Honeybee provides multiple components which can be connected between themselves in order to run simulations for glare analysis. Therefore in the following few pages the used algorithm for performing such analysis will be explained in detail.

#### 3.5.8.1. SIMULATION ALGORITHM OF COMPONENTS ACCORDING TO GLARE ANALYSIS

Firstly it was started with assigning the view because the Daylight Glare Probability (DGP) will be calculated from a specific point of view inside the small office. *Figure 3.50* shows the workflow of choosing a point from already created surface, which is against the wall with the window on it.

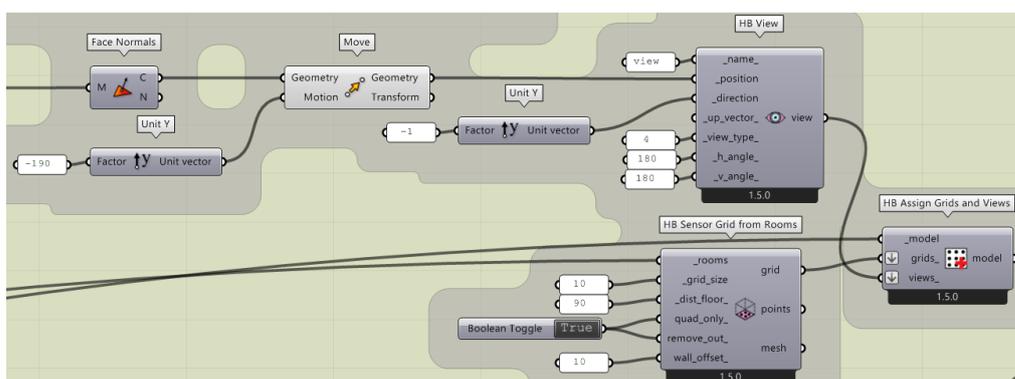


Figure 3.50 Screenshot from Grasshopper UI, Daylight Glare Probability, assigning grids and views

Then the point was moved 180cm along the y axis towards the window direction, to be situated in the middle of the room. This was inputted in "HB View" which allows to create a view for an image based analysis. It means that the DGP will be calculated and then also shown on an image. It is important to mention that "HB View" asks for the direction of the view, which was also set along the y axis in negative direction. The view type of the image was selected to be Hemispherical Fish-eye and the number of view horizontal and vertical size was set to 180. After this, the component "HB Assign Grids and Views" was again used in order to put the HB Model and the grids.

The next step is to use "HB CIE Standard Sky" which has few inputs: north 35°, location (.epw file of Turin), and for the date of the sky generator, different dates were chosen for multiple analysis. The idea was to perform DGP analysis for equinox, which occurs at the start of the spring and fall and solstice, which occurs during the summer and the winter. This component creates a standard radiance sky. *Figure 3.51* shows how this was connected to the "HB Point-In-Time View-Based" component which allows a point-in-time view-based analysis on the HB Model. The required inputs are the HB Model, already generated sky, as a metric luminance was chosen, radiance and settings parameters were assigned with the specific components for them and then the simulation was ran with "Boolean Toggle: True".

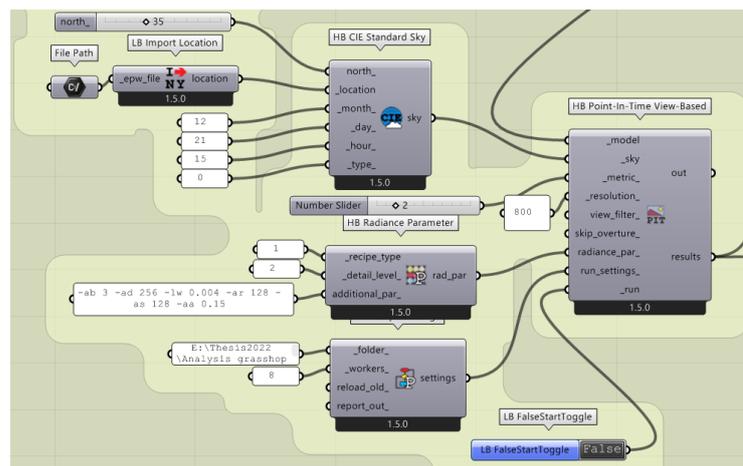


Figure 3.51 Screenshot from Grasshopper UI, Daylight Glare Probability, assigning sky and parameters

Lastly, in *Figure 3.52* the component "HB Glare Postprocess" was used in order to calculate the DGP, few more glare indices and see in which category by discomfort it belongs.

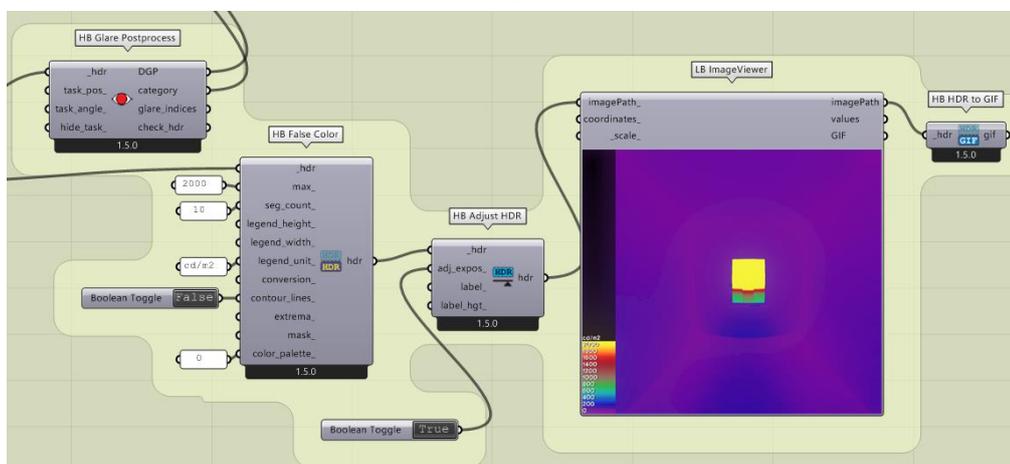


Figure 3.52 Screenshot from Grasshopper UI, Daylight Glare Probability, producing results

By this it means that this component divides the categories into the 4 types of glare, imperceptible ( $0.35 > DGP$ ), perceptible ( $0.4 > DGP \geq 0.35$ ), disturbing ( $0.45 > DGP \geq 0.4$ ) and intolerable glare ( $DGP \geq 0.45$ ). The supplied HDR image should be less than 1500 by 1500 pixels due to runtime requirements. The last output was check\_hdr, which is used as a path to High Dynamic Range (HDR) image from the glare study. It is connected to "HB False Color", used to convert HDR image into a false color version.

The inputted values were 2000 max cd/m<sup>2</sup> for luminance and 10 segment counts for the legend. Next step was to plug in the output of this component into "HB Adjust HDR" that allows to adjust the exposure of the image and also label the produced image. Into the "adj\_expos\_" there is a "Boolean Toggle: True" which brightens the image, in order to mimic what would be seen with an eye of a human. The resulted image with the legend scale can be seen with the component "LB Image Viewer" and later retrieved in order to be used for further evaluation. Additionally, the component "HB HDR to GIF" could be used for producing Graphics Interchange Format (GIF) which is much smaller than a HDR image.

The second simulation that was done was the Glare Autonomy. All of the steps used for this simulation were similar to all of the above algorithms used for the daylight and glare analysis. Components that were used along this algorithm are shown in *Figure 3.53*.

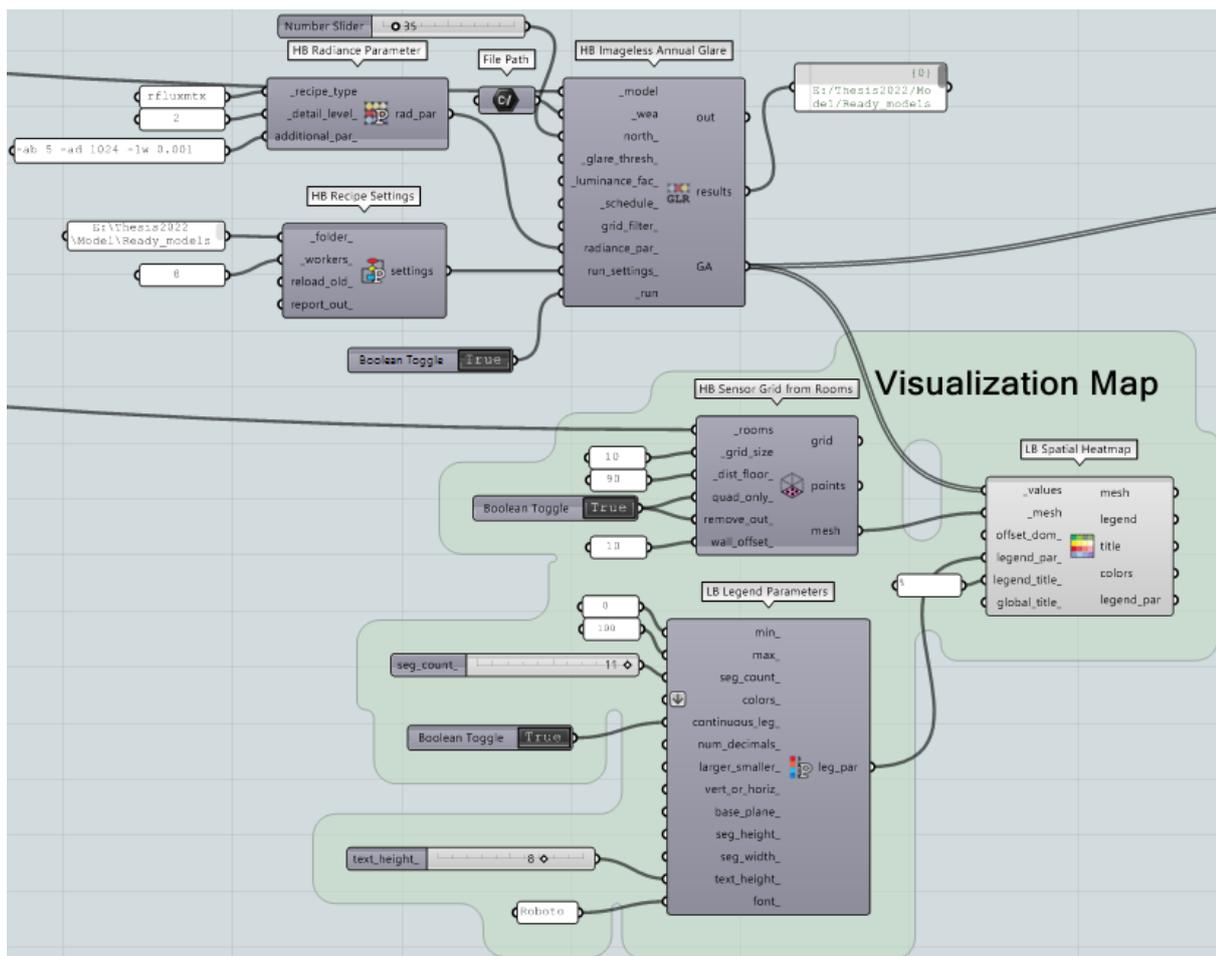


Figure 3.53 Screenshot from Grasshopper UI, Glare Autonomy, simulation adjustments and visualizing the results

"HB Imageless Annual Glare" is a Honeybee component that is used to compute DGP from which the GA is simulated. Few inputs are required in order to run the simulation. One is a fractional value representing the DGP threshold at which glare is thought to be caused. This value is used to calculate the percentage of hours in which the view is glare-free when calculating glare autonomy. (Default: 0.4). The once that were manually adjusted were the radiance parameters and run settings.

Same as all other performed simulations, the results in percent for Glare Autonomy (GA) can be plugged into the "LB Spatial Heatmap" component together with meshes of the sensor grids (*Figure 3.51*), which are adjusted with "HB Sensor Grid from Rooms" and the "LB Legend Parameters".

Lastly the Spatial Glare Autonomy (sGA) was calculated, using the results from the previously mentioned simulation for Glare Autonomy. *Figure 3.54* shows the components that were used in order to produce the needed results.

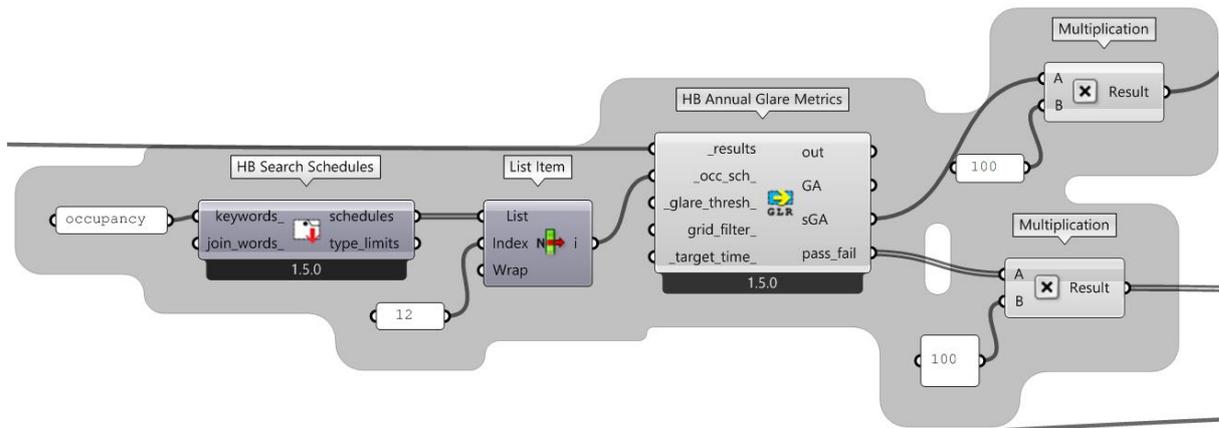


Figure 3.54 Screenshot from Grasshopper UI, Spatial Glare Autonomy, simulation adjustments

The first used component was "HB Annual Glare Metrics" which has multiple inputs that can be plugged in with text panels or resulted values from previous simulation. This simulates the percentage of grid that is free of glare according to the given threshold for given target time in percentage. An occupancy schedule was adjusted as in the past simulation algorithms. Still the glare threshold was set to 0.4 DGP. The outputs gave results for sGA, which needs to be multiplied by 100 in order to retrieve a percentage of area that is below the threshold.

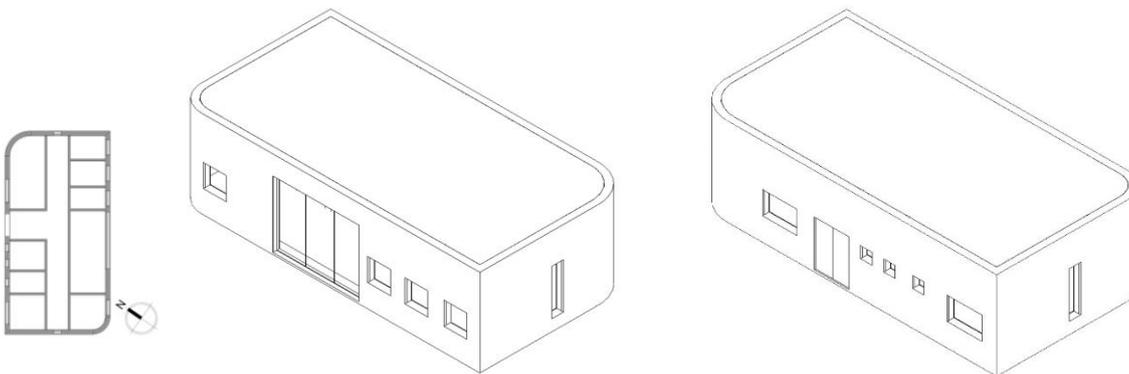
At the end the output "pass\_fail" is a data tree of zeros and ones indicating whether a particular sensor meets the standard for being glare-free (1) or does not (0). Glare-free does not necessarily imply that the sensor is glare-free 24/7, but rather that it is glare-free for a minimal portion of occupied hours as determined by the target time. Every value corresponds to a various grid sensor. These, along with meshes of the sensor grids, were inserted into the "LB Spatial Heatmap" component to see the results.

### 3.6. CASE STUDY II – MULTI-ZONE MODEL

Continuing from the previous steps that were taken in order to test a simple “shoe-box” model, with a greater knowledge about the interoperability between BIM and BPSTs (Revit and Grasshopper), a new more complex model was designed. The initial idea was to complete tasks that were faster and more accurate, because this model consists multiple zones and the previous workflow that was applied on the single-zone model, would be more time consuming if used for a building with larger complexity. Of course taking into account all the research papers and articles, mentioned in the second chapter, the study which was performed on the multi-zone model, pursued in a different method of data exportation. From constructing the 3D model in Revit, then transferring data to Grasshopper, using Rhino Inside Revit, to performing daylight and glare analysis using Honeybee and Ladybug. All these steps are explained in the following pages, elaborating each move with a purpose of using it as a user manual.

#### 3.6.1. DECRPTION OF MODEL

Initially a simple 3D model was constructed in Revit, consisting few rooms, with an office function. Compared to the single-zone model, this one was specifically used in order to level up the complexity of the testing model and also try and explore different shapes of model elements. Due to the found information about curved walls, an uncertainty about the amount of data transfer existed and of course this needed to be tested. Finally the new multi-zone model had two opposite corner rooms with curved exterior walls, including 10 rooms in total, length and width of 16x8m with an area of 128m<sup>2</sup>. The location is still considered in Turin, Italy and the north rotation being 35° (*Figure 3.55*).



*Figure 3.55 Multi-zone model*

Despite the fact that the single-zone model consisted elements with only one layer of material type, the multi-zone model elements were constructed with multiple material layers. Even though with the previous case study the materials properties were both selected and extracted from Revit to Grasshopper, in this case the materials’ properties were collected from a side excel file, consisting all the information about thermal and solar properties of different types of materials. This decision was made because with the first test there was an error with the material unit conversion within Grasshopper.

### 3.6.2. ASSIGNING ROOMS IN REVIT

Since Honeybee only accepts closed surface boundaries for comfort analysis in Grasshopper, it is crucial to thoroughly check the 3D model that was previously built. A closed surface boundary, for instance, is created by choosing the internal or external surfaces of a room's walls, roof, and floor. In this case a new method was used in order to choose a preferred type of surface. This workflow is based on extracting element data using the room assignment within Revit. This means that once rooms are assigned in Revit, the exportation of surfaces would be the faces that bounded the room area. The first step is shown on *Figure 3.56*, where the command “Room” under the menu “Architecture” is chosen from the Revit’s main toolbar. It basically creates a room which is bounded by the model elements and separation lines.

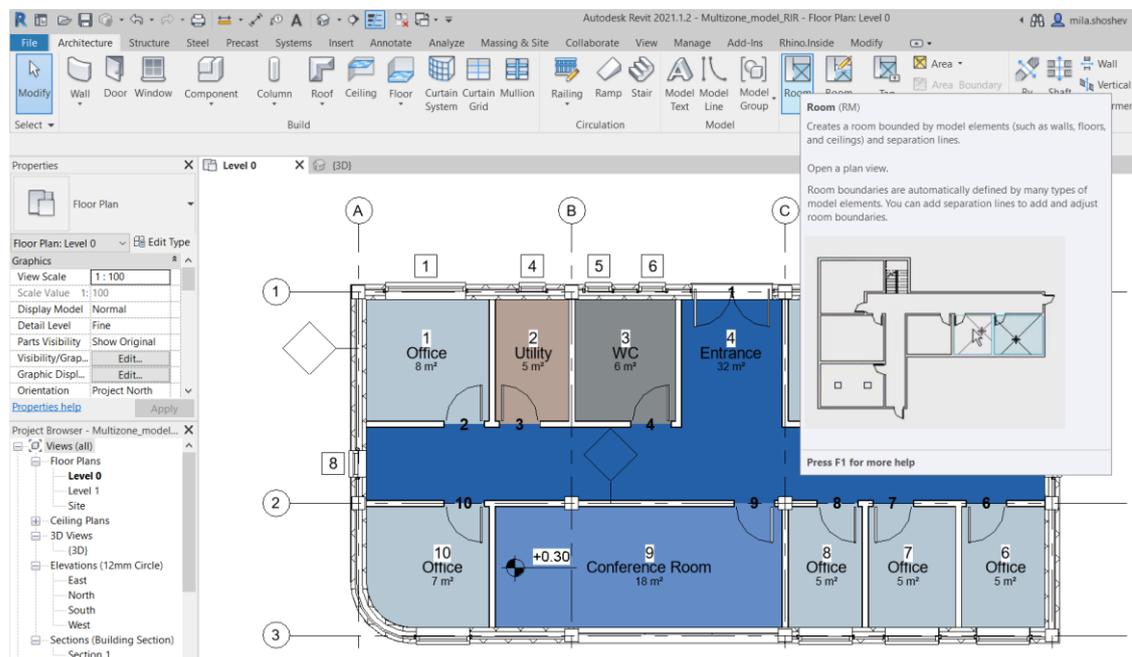


Figure 3.56 Screenshot from Revit UI, assigning rooms

Simply by clicking on the command “Room”, an area can be easily selected. Starting from left to right all the rooms within the multi-zone model were assigned. The step by step process is shown on *Figure 3.57*, where just by clicking on a specified area the program selects the bounded area, highlights and gives a number to the room by default.

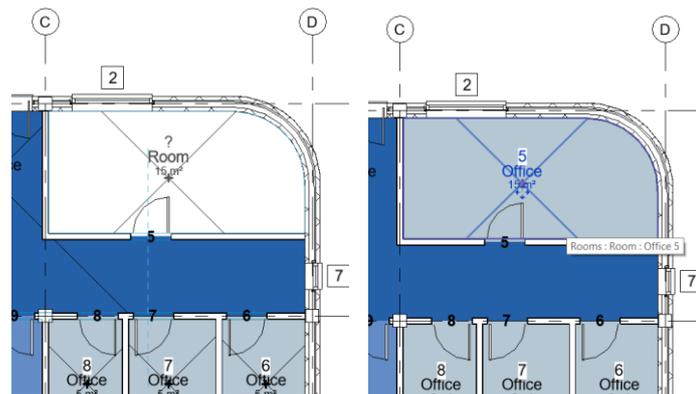


Figure 3.57 Screenshot from Revit UI, selecting the rooms

Moreover it shows also the room area and a label can be added too. After applying the rooms on each selected area, they would be assigned by a chronological number and the table can be retrieved as a legend, where each room is with a different color that signifies the function that was given by writing the label of the room.

Room type	Floor Area [m <sup>2</sup> ]	Glazing Area [m <sup>2</sup> ]	WFR [%]
Office 1	8	2.2	27.5
Utility 2	5	0.4	8
WC 3	6	0.8	13.3
Entrance 4	32	1.4	4.5
Office 5	15	2.2	14.7
Office 6	5	2.4	48
Office 7	5	2.4	48
Office 8	5	2.4	48
Conference Room 9	18	16.2	90
Office 10	7	2.4	34.3
All Rooms	106	32.8	31.0

*Table 3.3 Multi-zone model WFR calculation*

Table 3.3 shows all rooms with their floor and glazing area size. Here the calculation of Window to Floor Ratio (WFR) is calculated in order to see the difference between various sizes of floor and glazing areas, which could help in the analysis and simulations.

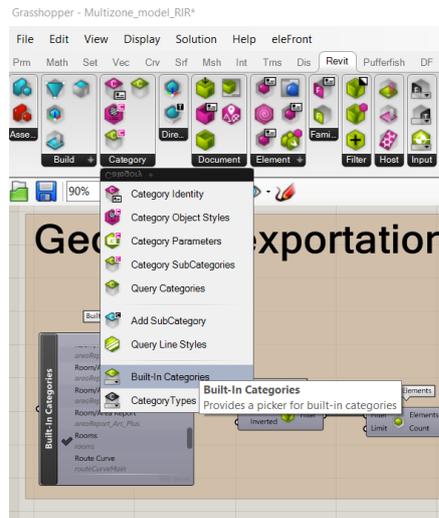
### **3.6.3. IMPORTING ROOMS WITH RIR IN GRASSHOPPER**

Rhino Inside Revit enables the opportunity of having a multiple ways of data exportation processes from Revit to Grasshopper. As mentioned previously the model elements can be exported separately according to their element category or their family. In this specific case the newly used method was room exportation. This allows an exportation which extracts the surface boundaries of a room, created in Revit. The general idea behind the use of this exportation process was to test if all the surfaces would import correctly, including the curved walls. Also another thing worth to mention is that by performing daylight and glare analysis all internal surfaces are needed, so in this case study when choosing the room importation method, it was easy to extract a closed surface boundary from the beginning of the exportation process. On the other hand this could possibly limit the idea of separately extracting model elements, but a much easier and cleaner Grasshopper script where all other properties, such as, material, geometry, etc, could be modified within Grasshopper UI.

#### **3.6.3.1. EXTRACTING EACH ROOM ACCORDING TO ROOM TYPE**

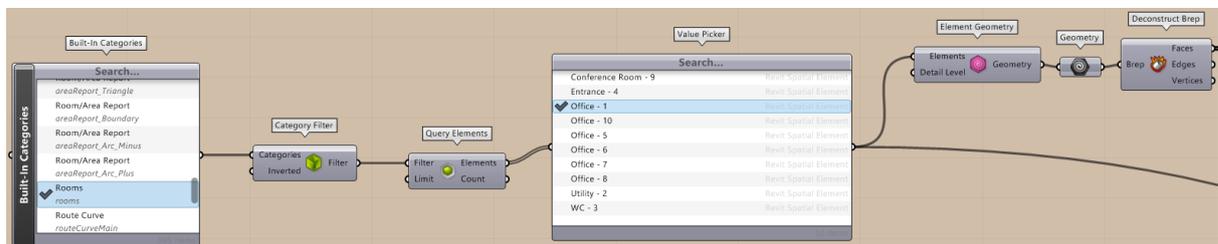
In the following pages an elaborated explanation will be presented on behalf of room extraction from Revit to Grasshopper, using Rhino Inside Revit commands. Since all the rooms were labeled in Revit it was much easier to select them according to their number and name. Within Grasshopper main

toolbar there is an availability to choose within built in categories, provided from Revit. As *Figure 3.58* shows, same as in the single-zone model, the component “Built-In-Categories” was used in order to extract the Rooms.



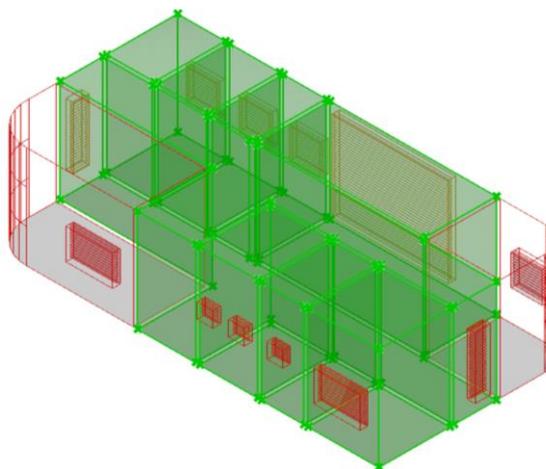
*Figure 3.58 Screenshot from Revit UI, category extraction*

Next step is using other components which will help taking out the elements of the room (*Figure 3.59*). The algorithm provided helps defining all the rooms that were already assigned in Revit. With “Value Picker” a specific room can be chosen from the dropping menu. This allows to extract the surfaces, edges and points from the selected room. A geometry will be then deconstructed with the use of “Deconstruct Brep” and the faces could be further divided into sections, according to the model element type (ex: interior/exterior wall, roof, floor, etc.).



*Figure 3.59 Screenshot from Revit UI, specifying room selection*

When completing this part of the Grasshopper script, most of the rooms were extracted correctly (*Figure 3.60*).



*Figure 3.60 Screenshot from Rhino UI, correctly extracted rooms*

### 3.6.3.2. EXTRACTING APERTURES

The apertures in the multi-zone model were extracted similarly to the single-zone model. The only difference is that for each room there needed to be a separate extraction for the number of windows it consists. An algorithm which is the same as in single-zone model was used (Figure 3.17).

The only difference is the extraction of the Curtain Wall. The script for this is pretty simple, but different from the other ones, “Wall System Family” component being the first one where a “Curtain” is selected. Using the “Query Walls” component it is transformed into a wall, where with the “Value Picker” can be selected the glazing and its geometry is easily taken out (Figure 3.61).

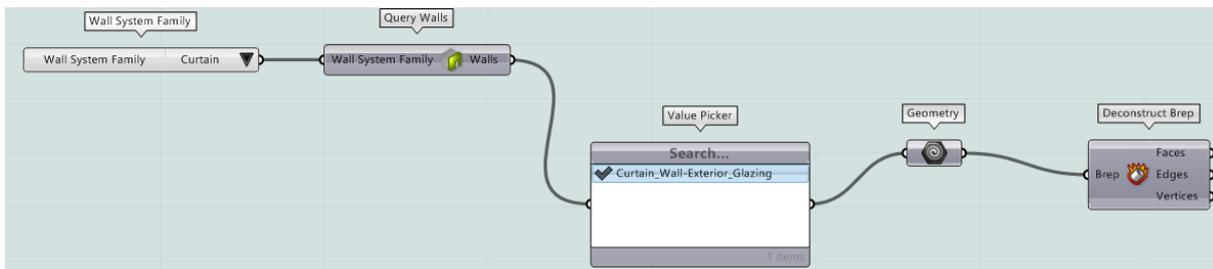


Figure 3.61 Screenshot from Revit UI, Wall System Family exportation type

### 3.6.4. GEOMETRY MODIFICATIONS ON GRASSHOPPER

As mentioned previously, the curved walls could become a problem, when exporting their surfaces. This results due to the multiple faces which construct the curved surface. These faces could result with not closing the boundaries, which would then not be able to make a HB Face, Room or Model (Figure 3.62).

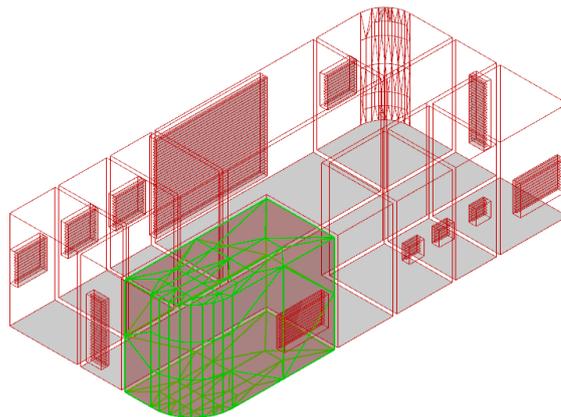


Figure 3.62 Screenshot from Rhino UI, incorrectly extracted room

Even with using this type of faces, which could be put into the HB Room together as one surface, the results that would be retrieved at the end will show wrong analysis on scattered surfaces. A way of solving this issue is constructing the surface from faces, edges or vertices. The algorithm of doing such thing could be performed in three ways. Firstly, a surface was constructed from points. Figure 2.63

consists an algorithm of multiple connected components, starting with “Face Boundaries” which allows to convert all the meshes into faces with edges and vertices.

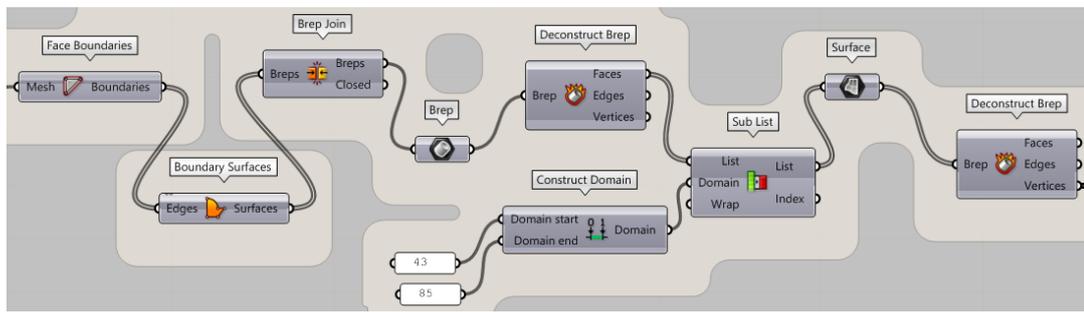


Figure 3.63 Screenshot from Grasshopper UI, selecting surfaces’s faces

Later they are deconstructed with “Deconstruct Brep” and in order to define a specific model element, for example floor component, the correct faces are selected using “Construct Domain”. This component helped with creating a domain with two numeric values, which was later plugged in as a fully constructed surface, created from the list of faces. Lastly, all the points are selected and plugged in in “Polyline” in order to create a boundary for the new surface, after the previous surface was deconstructed (Figure 3.64).

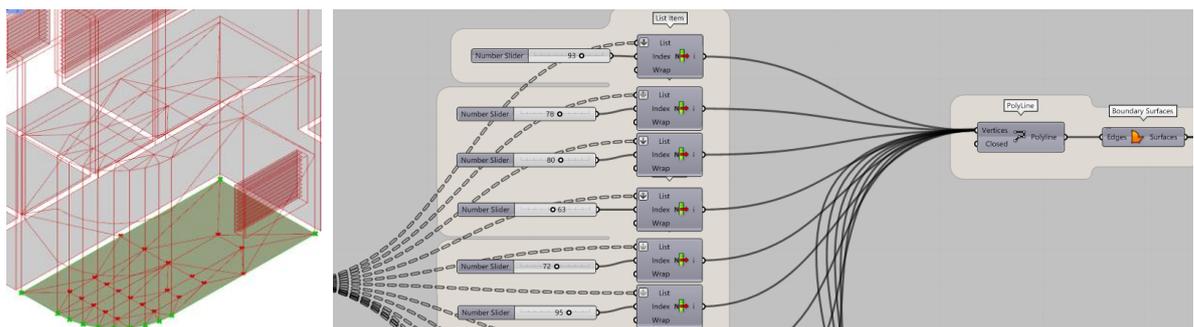


Figure 3.64 Screenshot from Grasshopper UI, surface from points

Next, the second type of surface construction was done by extracting multiple faces from the component “Deconstruct Brep” and selecting the needed ones with “Construct Domain”. Using the component “Sub List”, the list of faces was plugged into the input of “Surface” component. This resulted with seamless surface (Figure 3.65).

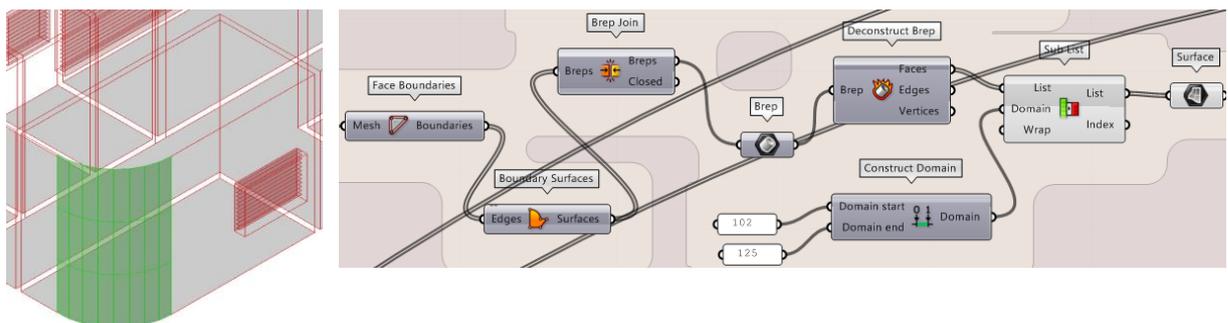


Figure 3.65 Screenshot from Grasshopper UI, surface from faces

Finally, the last surface was constructed from edges, which means that they were selected again from the component “Deconstruct Brep” (Figure 3.66). This time the new edges had to be merged so they can form a correct surface.

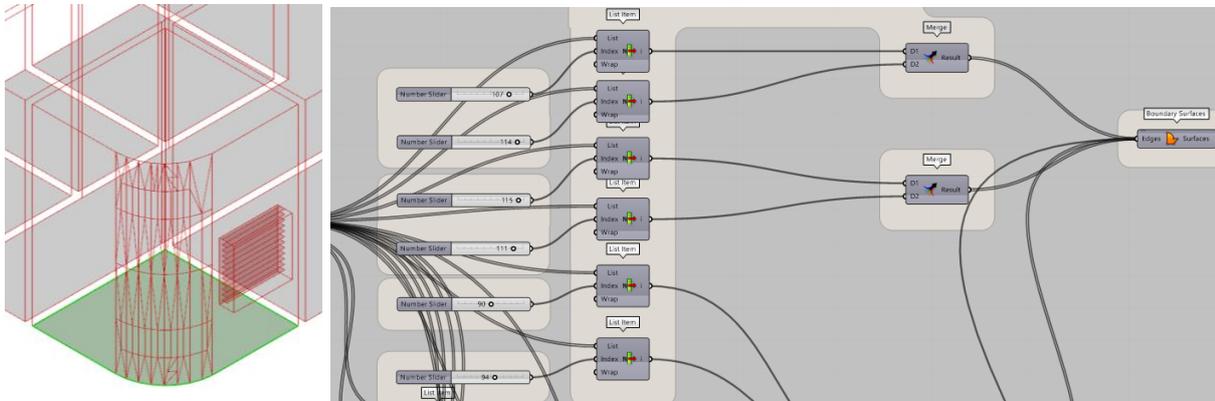


Figure 3.66 Screenshot from Grasshopper UI, surface from edges

After all these modifications the geometry is almost ready for the next part of the algorithm. All the created or extracted surfaces are taking part of groups which then form a whole room. This means that each room has assigned surfaces for interior and exterior walls, roof and floor.

Before continuing with the last part prior to simulations, it is worth to mention that the apertures didn't have any geometry modifications and they were extracted the same as in the single-zone model.

### 3.6.5. CREATING HONEYBEE MODEL

The development of the Honeybee Model for the multi-zone building and its readiness for the evaluation are the main topics of this section of the methodology chapter. As was already indicated, Honeybee has several groups of components that do different types of analysis. The first criteria for these analyses is to have a closed surface boundary, which indicates that there must be no surface intersections or gaps within the model.

The component intended for building the Honeybee Model will typically reject the created geometry if there is a gap. Many commands are available in the Honeybee library, but I specifically selected HB - Energy and HB - Radiance for the creation of the Honeybee Model and simulation of analysis on it, almost similarly to the single-zone model. I was given the option to select from a variety of schedules, fundamental properties, modifiers, light sources, and components to run a variety of analyses before using the components from the results menu and viewing the results in advance.

What was different from the first case study is that in this one, there are multiple zones that needed to be inputted and then perform the analysis at once, having the results for all rooms, but resulting with separately grouping the results for each zone.

### 3.6.5.1. HONEYBEE FACES, APERTURES AND ASSIGNING MODEL PROPERTIES

- *Opaque components*

There are few preparations that must be taken before building the HB Model for the multi-zone. Most importantly to address is that each zone has its faces assigned separately, before creating a HB Room for each room. The first thing to perform is to take the component "HB Face" and insert the node from the previously retrieved surfaces into the "geo" input (*Figure 3.67*).

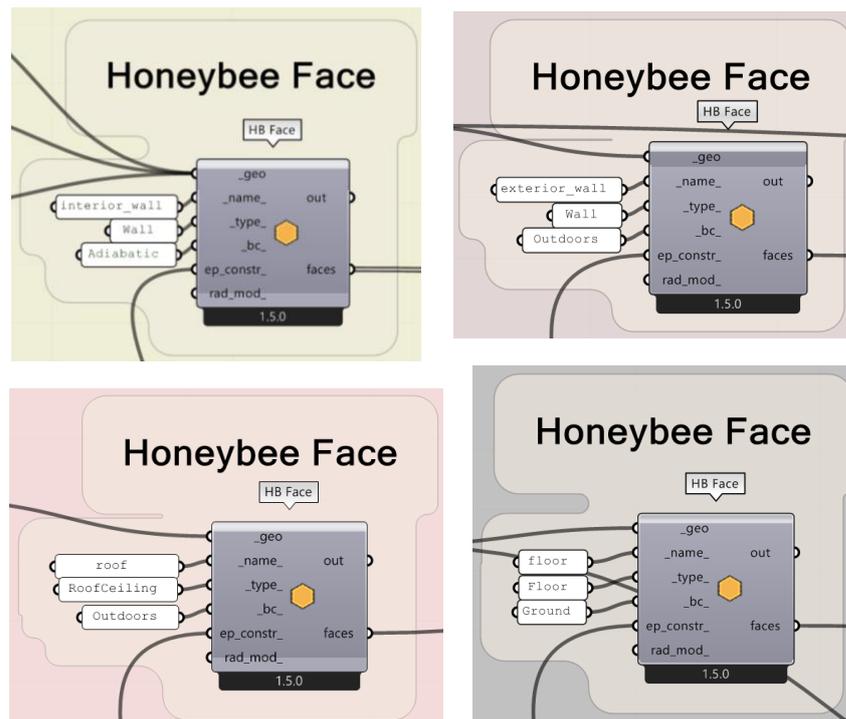


Figure 3.67 Screenshot from Grasshopper UI, HB Face for opaque surfaces

These actions are made while inputting the opaque surfaces, exterior and interior walls, floors, roofs, for each zone. The component “HB Face” has few inputs, such as “\_type\_” for the face type, “\_bc\_” for the boundary condition where a selection between Ground (Floor), Outdoors (Roof and Exterior Walls) and Adiabatic (Interior Wall) is given. Lastly, the “\_ep\_construction\_” input is the most important one, because it gives the opaque material a construction of materials.

- *Transparent components*

Also all the transparent components were assigned including all windows and the curtain wall in one of the rooms. Firstly, a “HB Aperture” that displays the window openings was created. The glass surface was allocated as the “HB Aperture's” face in this case study with various windows in each room. Aperture name, operability status (“Boolean Toggle: True”), material characteristics, and radiance modifier are all things that the component requests. The "HB Glass Modifier 3" was used in this instance to connect the radiance modifier and allow for the creation of the radiance according to the transmittance

of red, green, and blue light. A 0.75 was entered for each of the parameters. Finally, this modification can be used to modify the index of refraction; in this instance, the value was set to 1.52 for float glass (Figure 3.68).

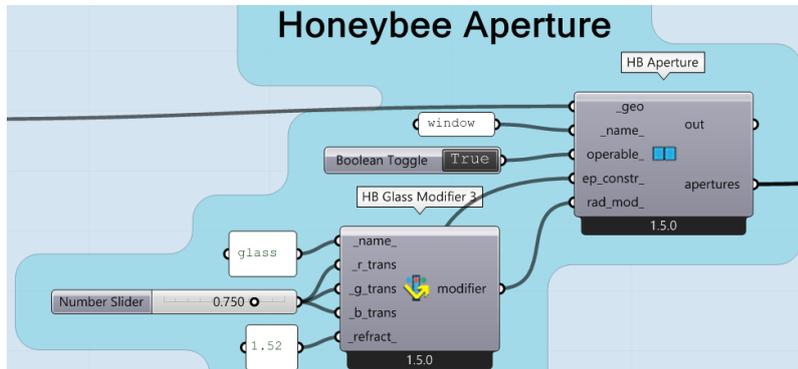


Figure 3.68 Screenshot from Grasshopper UI, HB Aperture for transparent surfaces

### 3.6.5.2. CONSTRUCTING MATERIALS AND ASSIGNING THEIR PROPERTIES TO THE COMPONENTS

Due to the lack of information from the extracted material assets from Revit, there was a need of writing the properties for all the material layers manually. For this case study the building elements consist multiple layers in their material type, so they had to be constructed and given all values of performance, with a specific components from the Honeybee library. The most important part was to include as much as information there was for the selected materials. There was not a preferable choice for the material type, but the materials that were used were chosen in order to see how the building will perform with a general selection of material type.

- *Opaque material*

Multiple layers are made with the “HB Opaque Material”, consisting their properties and then merged into the component “HB Opaque Construction”, which at the end is plugged into the input “ep\_constr\_” from “HB Face” component. This type of material construction is the same for all surfaces which represent different model element. For example the roof has the same properties in each room. The building elements are with defined layer properties which as mentioned previously were taken from a file consisting information about types of material layers.

In Figure 3.69 the construction of the roof material is shown. “HB Opaque Material” has several inputs, such as name, thickness, conductivity, density, specific heat, roughness, etc. Few of them were just left as default from Grasshopper. According to the roof element, the layers that were used are listed as following, starting from the outside to the inside of the room: gravel, bitumen, rigid insulation, vapor retarder, cement screed, concrete and finish plaster.

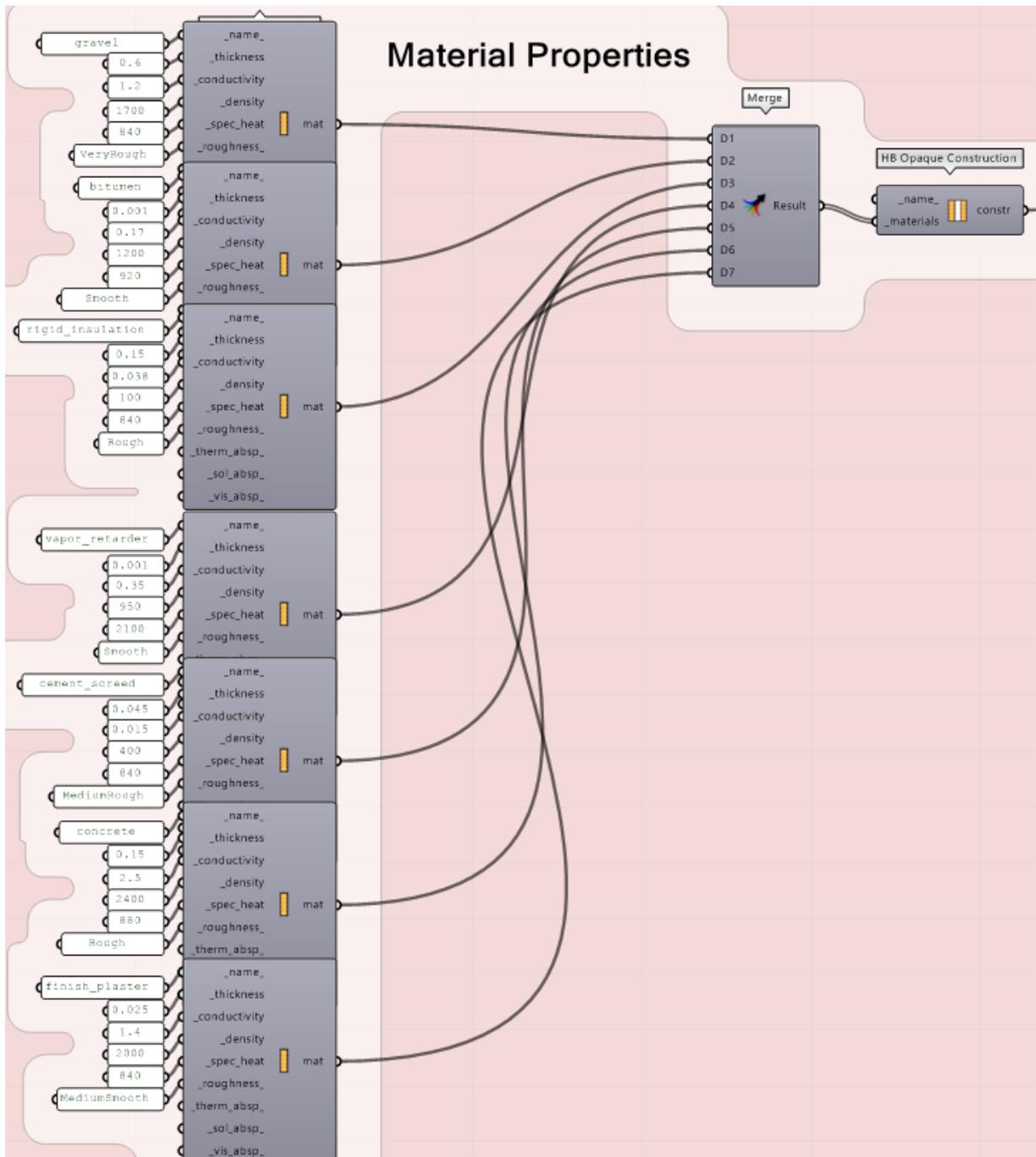


Figure 3.69 Screenshot from Grasshopper UI, HB Opaque Material construction, Roof

In the following *Figure 3.70*, *Figure 3.71* and *Figure 3.72*, the elements, such as, exterior and interior walls and floor are shown with their material construction. The material layers of the exterior wall are: finish render, concrete masonry units, air infiltration barrier, fiberglass batt, concrete masonry units and finish plaster. Then for the interior wall, two finish gypsum boards with fiberglass batt in the middle.

Lastly for the floor there are: concrete precast, concrete masonry units, dump proofing, air infiltration barrier, rigid insulation, vapor retarder, concrete screed and a finish of oak flooring. All of these constructed materials are merged and plugged in “HB Opaque Construction”.

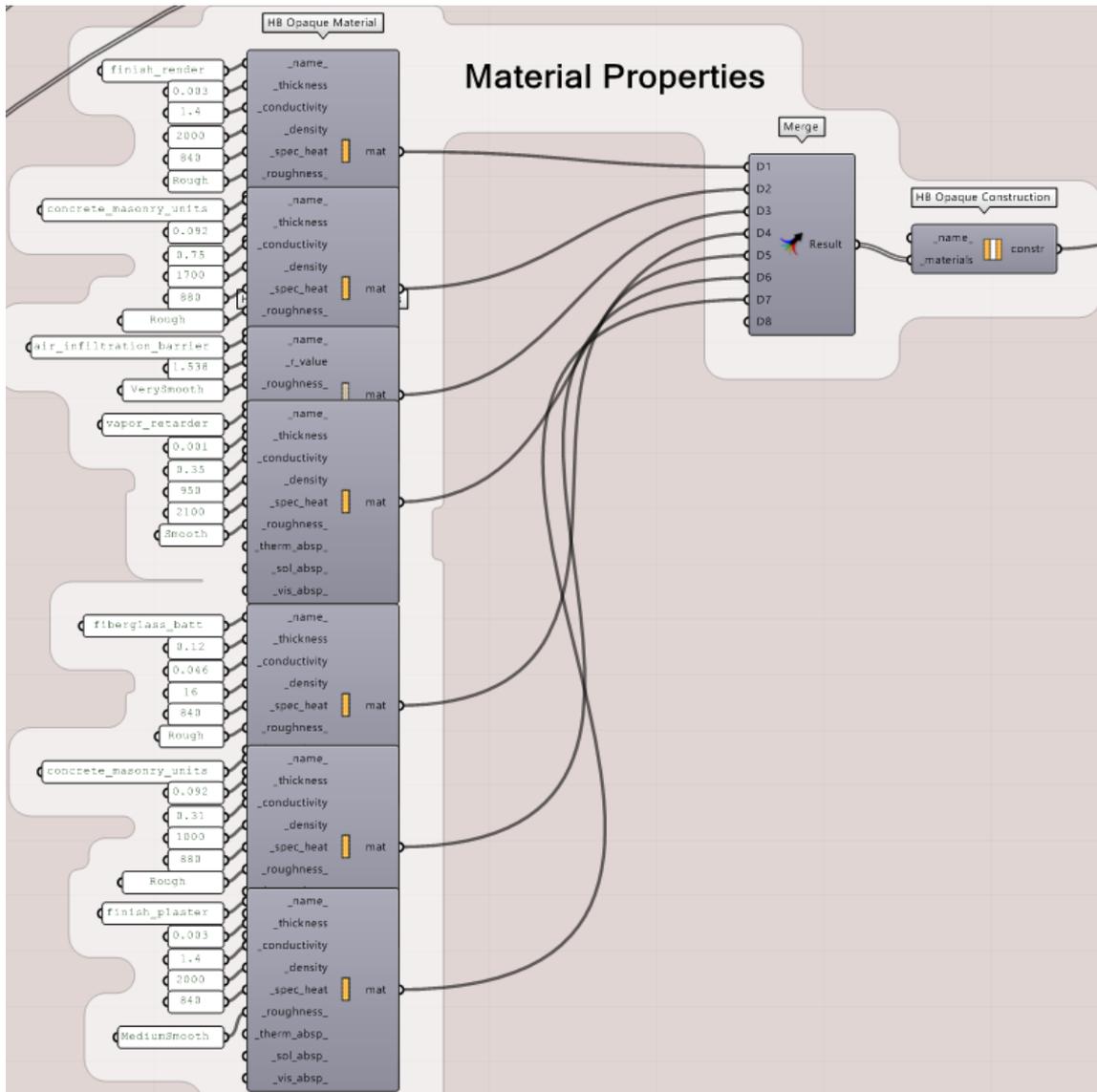


Figure 3.70 Screenshot from Grasshopper UI, HB Opaque Material construction, Exterior Wall

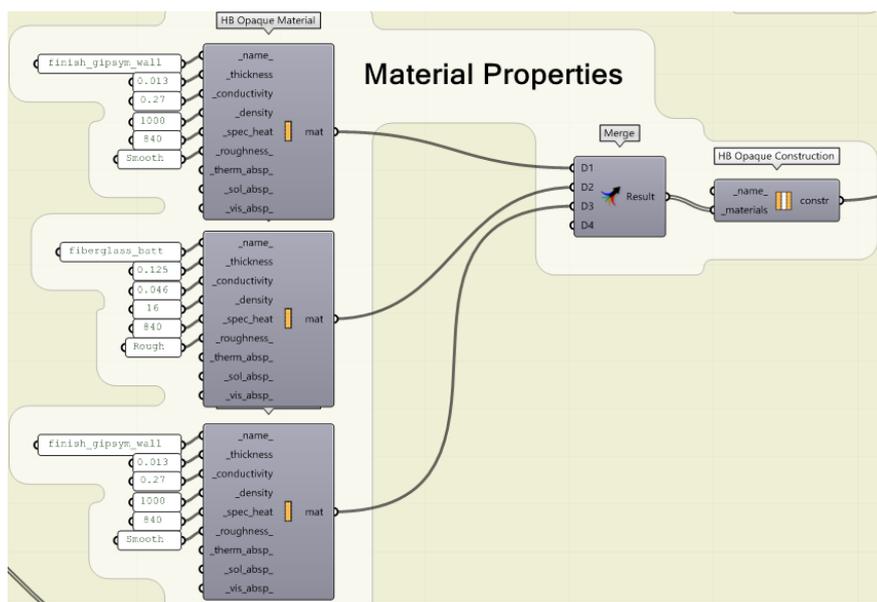


Figure 3.71 Screenshot from Grasshopper UI, HB Opaque Material construction, Interior Wall

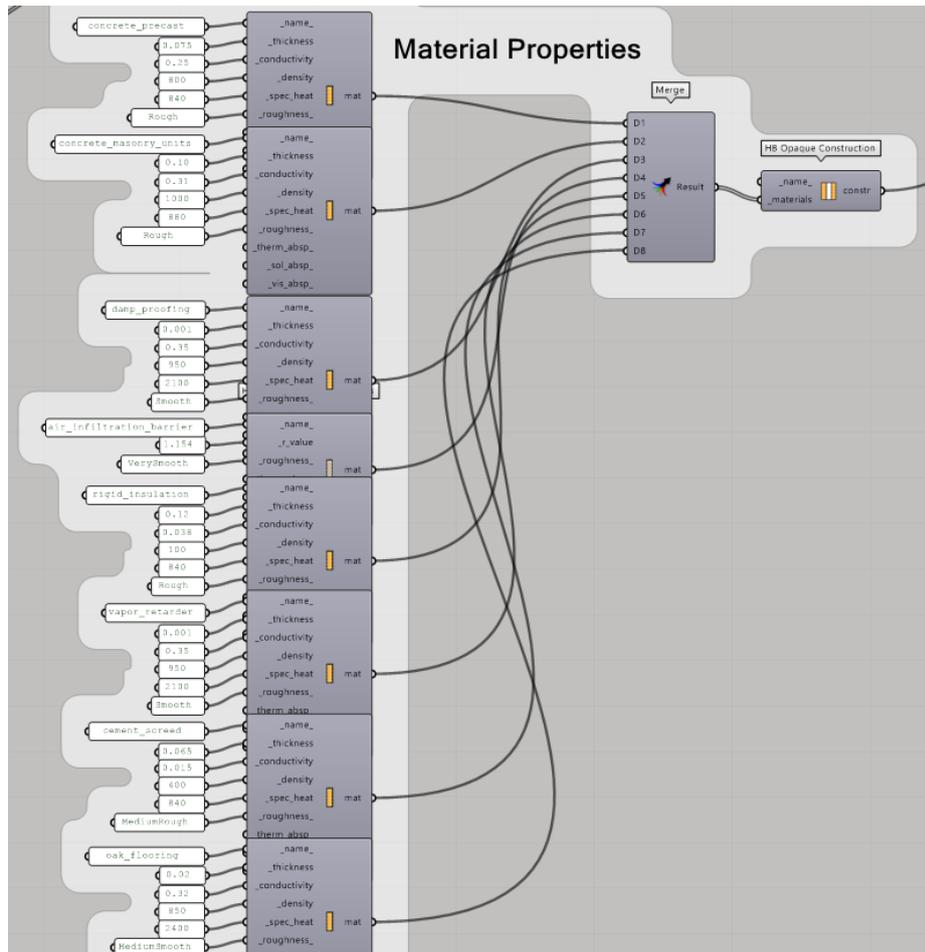


Figure 3.72 Screenshot from Grasshopper UI, HB Opaque Material construction, Floor

- Transparent materials

Furthermore, since the interior and exterior walls, roofs and floors were opaque material, a different approach was used for the transparent surfaces, windows and curtain wall. Each room had the same material properties, Solar Factor  $g=35\%$  and Thermal Transmittance of  $U_g=0.4$  and for their construction a specific algorithm was used. This algorithm is better explained in the following Figure 3.73.

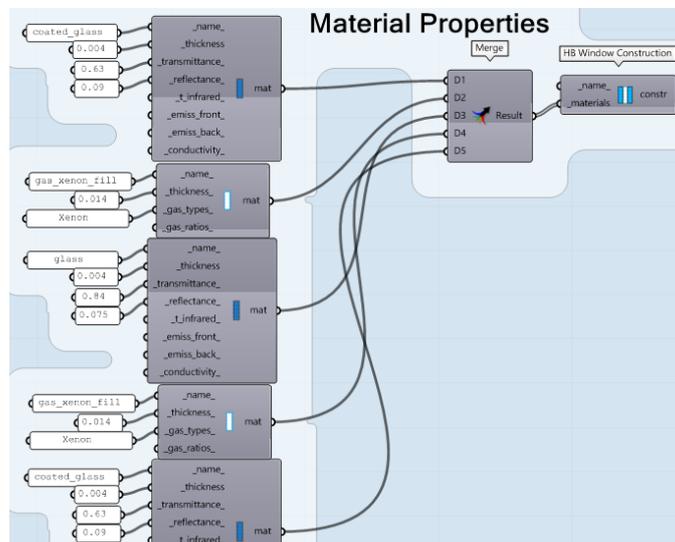


Figure 3.73 Screenshot from Grasshopper UI, HB Window Construction

### 3.6.5.3. HONEYBEE ROOM

The HB Model construction process's next to final phase is to create the HB Room. It was necessary to input all of the HB Face components in "faces" in this phase. Other plugged-in inputs include name, program, and whether or not the room is air-conditioned; in this case, "Boolean Toggle: False" was utilized. A program for a Small Office was selected from the component "HB Built-in Programs". This step was done for each zone existing in the multi-zone model. If all the data is accurately input and there are no gaps in the analytical model, the component "HB Room" will be active and functioning. This adheres to the closed surface boundary rule that was previously mentioned. Ten HB Rooms were successfully built using all the necessary components, including walls, roofs, floors. These rooms will be assessed and incorporated into the "HB Model" later on. For the investigation of the daylight and glare in this case study, all of the rooms were tested together. All of the inputs used up to this point have been modified and carefully selected in accordance with the standards for calculating visual comfort KPIs. Following this phase, the component "HB Visualize All" can be used to visualize each room in order to see the finished model (*Figure 3.74*).

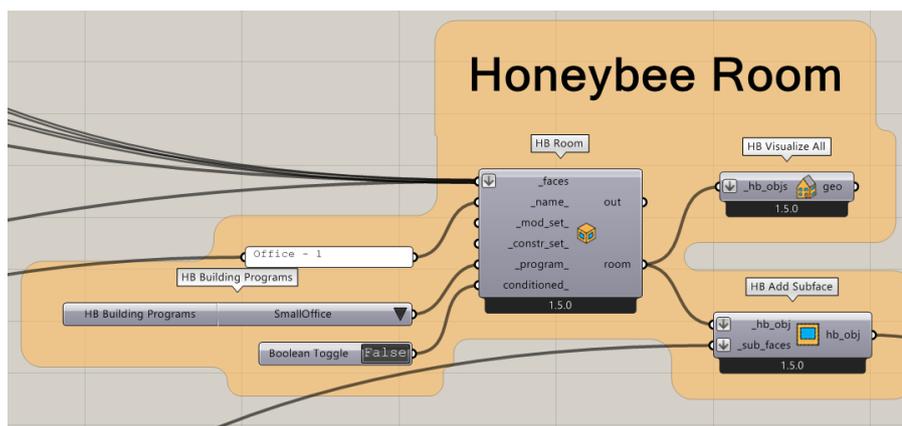


Figure 3.74 Screenshot from Grasshopper UI, HB Room (Office - 1)

Lastly if there is no shading devices each aperture has to be connected in the input “\_sub\_faces” of “HB Add Subface” component that accepts the HB Room in the input “\_hb\_obj”.

### 3.6.5.4. OUTDOOR LOUVER SHADES

Since the dynamic shades were used in the first case study, for the single-zone model, in this case study there was an application of louver shades, which were provided with a component that exists in the Honeybee component library. In *Figure 3.75* an algorithm is constructed in order to make the wanted louver shades. The component “HB Louver Shades” allows to produce the louver shades on a selected aperture. This means that the input “\_hb\_objs” is connected with the “HB Aperture” and then for the properties of the louver elements other inputs can be plugged in as values. The louver shades are horizontal outdoor louvers with a depth of 10cm, façade offset of 5 cm and two tested ways of inclination

angle 0° and 30°. As construction input there was a default opaque material from the “HB Shade Construction” component.

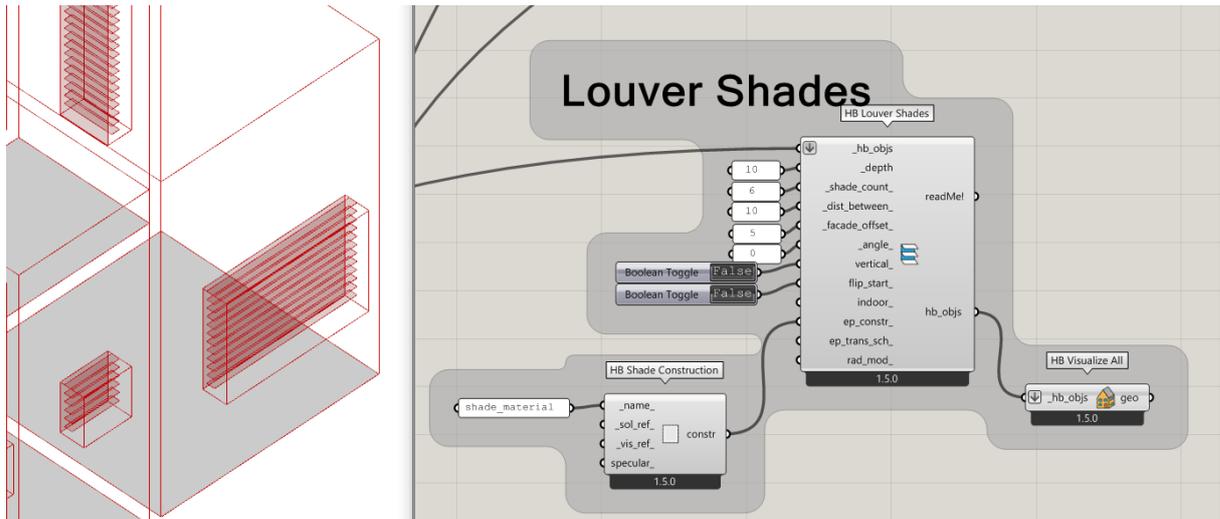


Figure 3.75 Screenshot from Grasshopper and Rhino UI, HB Louver Shades

Lastly, the output “hb\_objs” from the component “HB Louver Shades” can be plugged into the component “HB Add Subface” as a “\_sub\_faces” input.

### 3.6.5.5. STILES AND SILLS CREATION

In the previous model, the stiles and sills were created with the help of the exported geometry from Revit, but in this case study, for the multi-zone model, the sills and stiles were created with a help of the component “HB Extruded Border Shades”. This component allows to provide the depth of the wall, closest accuracy to the real dimension (Figure 3.76).

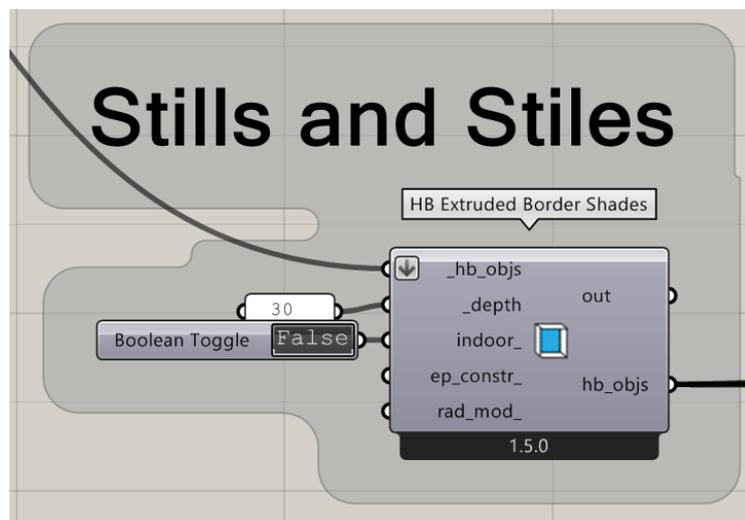


Figure 3.76 Screenshot from Grasshopper UI, HB Extruded Border Shades

This component is lastly plugged into the “HB Model” as a final version of the room that was previously created. Each room should then be inputted into the component “HB Model” and if there is no error it means that all the steps of the algorithm are correct. If there is a case of an error with the output “out” connected to a panel, an explanation will be carried out for where the error occurred (Figure 3.77).

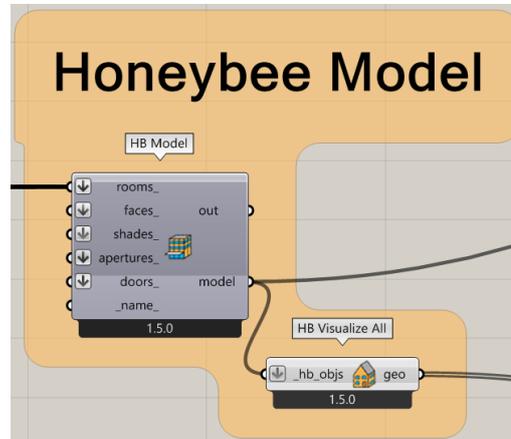


Figure 3.77 Screenshot from Grasshopper UI, final HB Model with louver shades

### 3.6.6. DAYLIGHT SIMULATIONS

Simulations for the building envelope of the second case study were run with the intention of performing daylight and glare analysis utilizing Honeybee and Ladybug components. It was able to run the analysis thanks to the algorithm simplicity, which required few adjustments and straightforward parameters. When doing the simulations, two different methodologies were taken into account. Instead of using the geometry of the outdoor and indoor stiles and sills and the glass itself, the initial test simulation was done without any daylight obstruction to see how the louver shades will function. An inclination was used to control the light transmittance in the second method, which used louver shades. Several KPIs were chosen to evaluate the performance of the building envelope.

The two differences between the first case study simulations and the second one was that in the second one the analysis were done simultaneously on all rooms, existing in one Honeybee Model and the different type of shading devices.

#### 3.6.6.1. SIMULATION ALGORITHM OF COMPONENTS ACCORDING TO DAYLIGHT ANALYSIS

Due to the same approach which was taken in the test simulations in the first case study, the single-zone model, in these following paragraph there will be only a brief explanation of the algorithm that was used for the second time in the multi-zone model. The multi-zone model's performance metrics

included Daylight Factor (DF), Daylight Autonomy (DA), Spatial Daylight Autonomy (sDA), Continuous Daylight Autonomy (cDA), Useful Daylight Illuminance (UDI), and Annual Sunlight Exposure (ASE). Following completion of the HB Model, the approach was nearly same in both scenarios—no shading and louver shading. The Radiance engine's processing time for the analysis was the only thing that was different, but that will be detailed later in the discussion chapter. The components used for the analysis, the inputs they require, and the outputs they generate were all described in detail in the pages below, which are almost the same as in the single-zone model. An important thing to mention is that for the grid assignment, all the rooms must be inputted in the input “\_rooms” of the “HB Sensor Grid from Rooms” component (Figure 3.78).

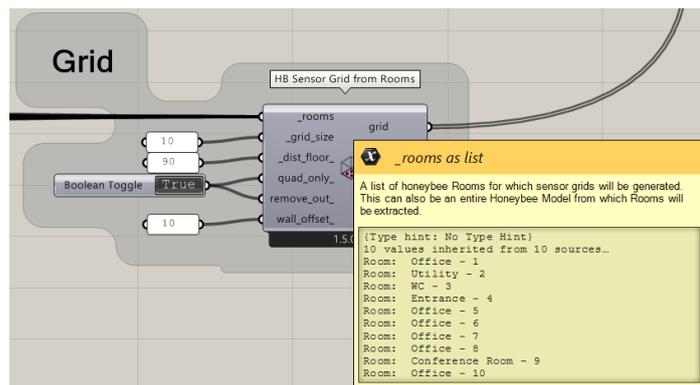


Figure 3.78 Screenshot from Grasshopper UI, Grid with list of all inputted rooms

The "HB Daylight Factor" component, which directly mimics a study of the daylight factor of an HB Model, is where the daylight simulation analysis starts. Default radiance parameters were utilized, and "HB Recipe Settings," which saves the results in a chosen folder and allows the option to choose the number of workers (CPUs) involved in the recipe execution, was used for the specific settings on how the recipe should be run. The simulation was then ran by adding a "Boolean Toggle: True". In order to visualize the results with the supplied colors and parameters—in this case, same as in the single-zone model, a range from 0 to 10 for the Daylight Factor stated in percentage—the results output was attached to a "LB Spatial Heatmap". The mesh for the visualization needed to be join. To input the aforementioned alterations for the legend's outcome, the "LB Legend Parameters" component was used (Figure 3.79).

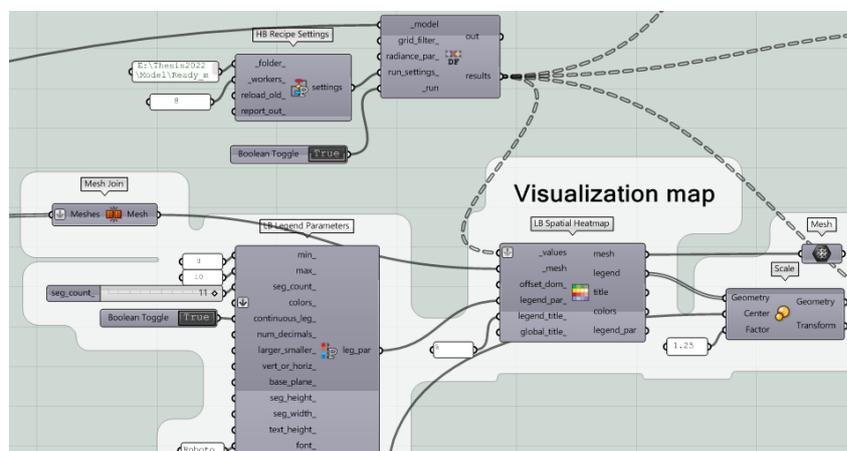


Figure 3.79 Screenshot from Grasshopper UI, Daylight Factor simulation algorithm II

The next KPIs were assessed using just one Honeybee component, the same one used in the single-zone case study. The "HB Annual Daylight" component, which enables simulations for analyzing a few daylight performance metrics, was employed for this. Model (prepared HB model), .wea or .epw file (used: .epw), north (rotated: 35°), default inputs (threshold, schedule, grid filter), radiance parameters, run settings (CPU: 8), and run (Boolean Toggle: True) were required inputs (*Figure 3.80*).

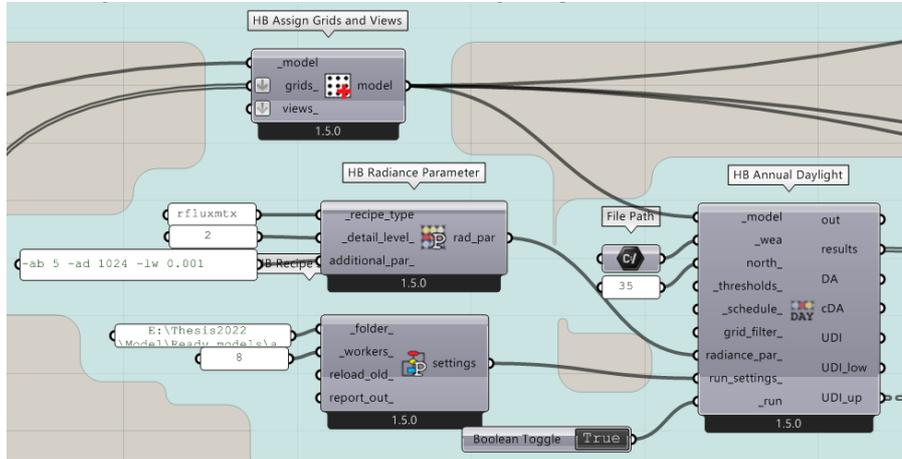


Figure 3.80 Screenshot from Grasshopper UI, Annual Daylight simulation algorithm II

The next phase, which allowed for the selection of an occupancy schedule with the "HB Search Schedules" (where the schedule is a regular schedule from 8am to 6pm), involved connecting the findings produced to the "HB Annual Daylight Metrics." This was chosen by selecting the required timetable from a list using a "List Item." The minimum and maximum illuminance criteria for useful daylight illuminance in lux (Default: (100,3000)) can be added as inputs to "HB Annual Daylight Metrics" (*Figure 3.81*).

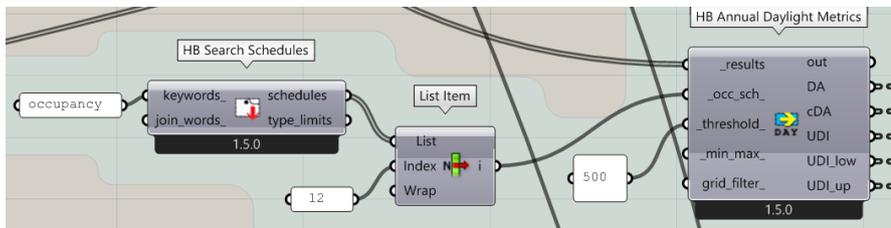
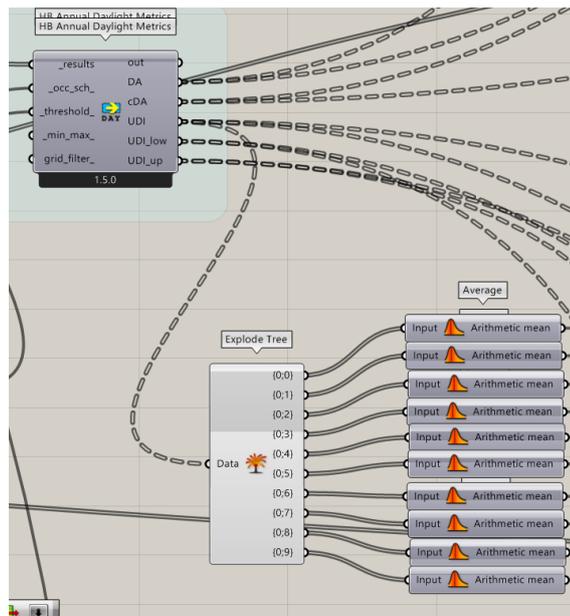


Figure 3.81 Screenshot from Grasshopper UI, Annual Daylight Metrics algorithm II

However, in this particular situation, a daylight autonomy threshold of 500 lux was supplied for all zones in the multi-zone model. The outputs of this component are the outcomes of the key performance metrics for daylight analysis, Daylight Autonomy (DA), Continuous Daylight Autonomy (cDA), and Useful Daylight Illuminance (UDI). The additional indicators include UDI's insufficient (0-100 lux) insufficient illumination, UDI's supplemental (100-500 lux) integration of electric light, and UDI's autonomous (500-3000 lux) illumination, no need for electric lighting and UDI exceeded (>3000) undesirable conditions such as glare, overheating, etc., which had results for the percentage of time that illumination is between the minimum and maximum threshold for useful daylight illuminance, assigned in the "\_min max\_" input of the component "HB Annual Daylight Metrics." Later, these outputs, which comprised the simulation results, were entered one by one into other components to enable the depiction of the outcomes of all rooms.

The components of the Ladybug tool were once more used for visualization. The "LB Spatial Heatmap" has its input values connected via a node to the various outcomes of the prior simulations, just like in Daylight Factor. All of them were displayed using "LB Legend Parameters" to modify the legend parameters, and the range for the minimum and maximum value was 0 to 100, stated in percentage. Differently from the first case study, the multi-zone model consists multiple rooms and because of that an average value for the whole building can't be used. Therefore as shown on *Figure 3.82* the results were split with the component "Explode Tree" in order to retrieve each value for each room in a separate branch.



*Figure 3.82 Screenshot from Grasshopper UI, Retrieving each room's values for calculating an average value*

One more step was taken only for the Spatial Daylight Autonomy (sDA). The simulation for the other daylight measures has this technique as an addition. This component required a mesh and Daylight Autonomy input in order to execute the desired simulations. Results for Daylight Autonomy were collected from the component "HB Annual Daylight Metrics" that was previously used, and the mesh was created using "HB Sensor Grid from Rooms" with the same modifications as for the other components, consisting all the rooms from the multi-zone model.

The Annual Sunlight Exposure (ASE) simulation was the final one run for the no blinds multi-zone model. With the use of a newly developed method and some of the same elements from earlier examples, this simulation was carried out. "HB Direct Sun Hours," a component of the simulation that requires model input (prepared HB model), a .wea or .epw file (used: .epw), north 35°, grid filter, run parameters (CPU: 8), and run (Boolean Toggle: True). The findings I obtained from this component were then linked to the "HB Annual Daylight Metrics," which used the component "HB Schedule to Data" to get data from the occupancy schedule. This component's threshold input value of 1000 is required for this type of analysis to be performed for the ASE, which indicates that it will analyze an area with 1000 Lux (*Figure 3.83*).

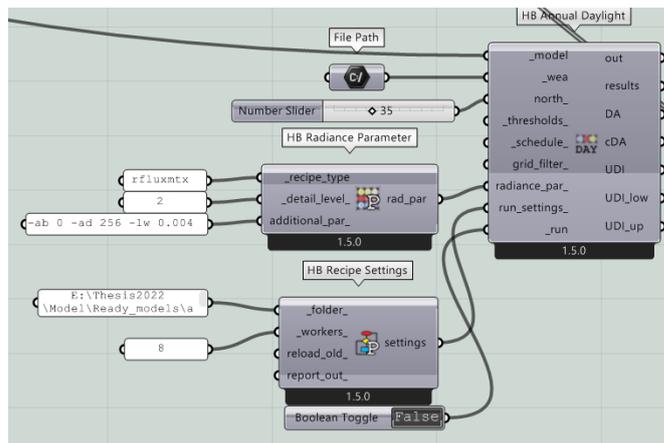


Figure 3.83 Screenshot from Grasshopper UI, ASE analysis II

Here, the "LB Analysis Period" was used to modify the analysis period and set it to last for an entire year. The data from the occupancy schedule and the Daylight Autonomy values that were a result of "HB Annual Daylight Metrics" were then combined using an expression. Two inputs were used in the defined expression: x (Daylight Autonomy findings) and y (deconstructed data, in which each value higher than 0 points out that there is an occupant in the room and this was transformed with an IF function to 1 and 0, for true and false, and the total number of existent 1s was then used as multiplication in the expression). As a result, the occupancy hours during which the DA reaches the threshold was produced for each zone in the multi-zone model. The "LB Spatial Heatmap" was once again used to illustrate the findings, but the most crucial aspect is that the minimum value for the hours needs to be 250 when setting the legend's specifications in "LB Legend Parameters." Additionally, the mesh must be entered because it will supply the number of faces or vertices that correspond to the number of values and allow them to be colored for the graphical preview of the results. Next the same expression was created to select the locations where DA is 1000lux more than 250 occupancy hours after the results from the previously mentioned expression, which yielded the total number of occupancy hours where DA is 1000lux, were obtained. This was then divided by the total number of points that were determined by combining the "List Length" component, which counts the number of values in a list, with the "HB Sensor Grid of Rooms" component. Then the division result was multiplied by 100 to obtain the percentage of an area.

This section summarizes every step that was performed to mimic the daylight analysis in both the no-blinds and louver blinds scenarios. In order to simulate multiple algorithms, some of the simulations were repeated. Lastly, the outputted results will be shown in the results chapter, but it is worth to mention that all of them should be separated in order to produce average values for each room.

### 3.6.7. GLARE SIMULATIONS

There were three different types of simulations run for the multi-zone model for glare analysis, just like there were for the single-zone model. Daylight Glare Probability (DGP) is the first, and Glare

Autonomy (GA) as the second one and Spatial Glare Autonomy (sGA) as third. To run simulations for glare analysis, Honeybee offers a variety of components that can be coupled to one another (*see 3.5.8 Glare Simulations*). The opportunity to test if the louver shades make any difference within the analysis, was the main reason of simulating glare analysis for each room.

### 3.6.7.1. SIMULATION ALGORITHM OF COMPONENTS ACCORDING TO GLARE ANALYSIS

The procedures followed for the simulation of Glare Autonomy (GA) were identical to those followed for the analysis of daylight and glare using the aforementioned algorithms, also used for the single-zone model. A Honeybee component called "HB Imageless Annual Glare" is utilized to calculate DGP, which is then used to simulate the GA. The simulation only requires a small number of inputs to execute. The DGP threshold at which glare is believed to be caused is represented by the fractional value one. When measuring glare autonomy, this value is used to determine the proportion of hours during which the view is glare-free. The run settings and radiance parameters were the only ones that required manual adjustment. *Figure 3.84* displays the elements that were used along this algorithm.

Similar to all other simulations run, the Glare Autonomy (GA) results in percent can be plugged into the "LB Spatial Heatmap" component together with sensor grid meshes that have been modified using the "HB Sensor Grid from Rooms" and the "LB Legend Parameters".

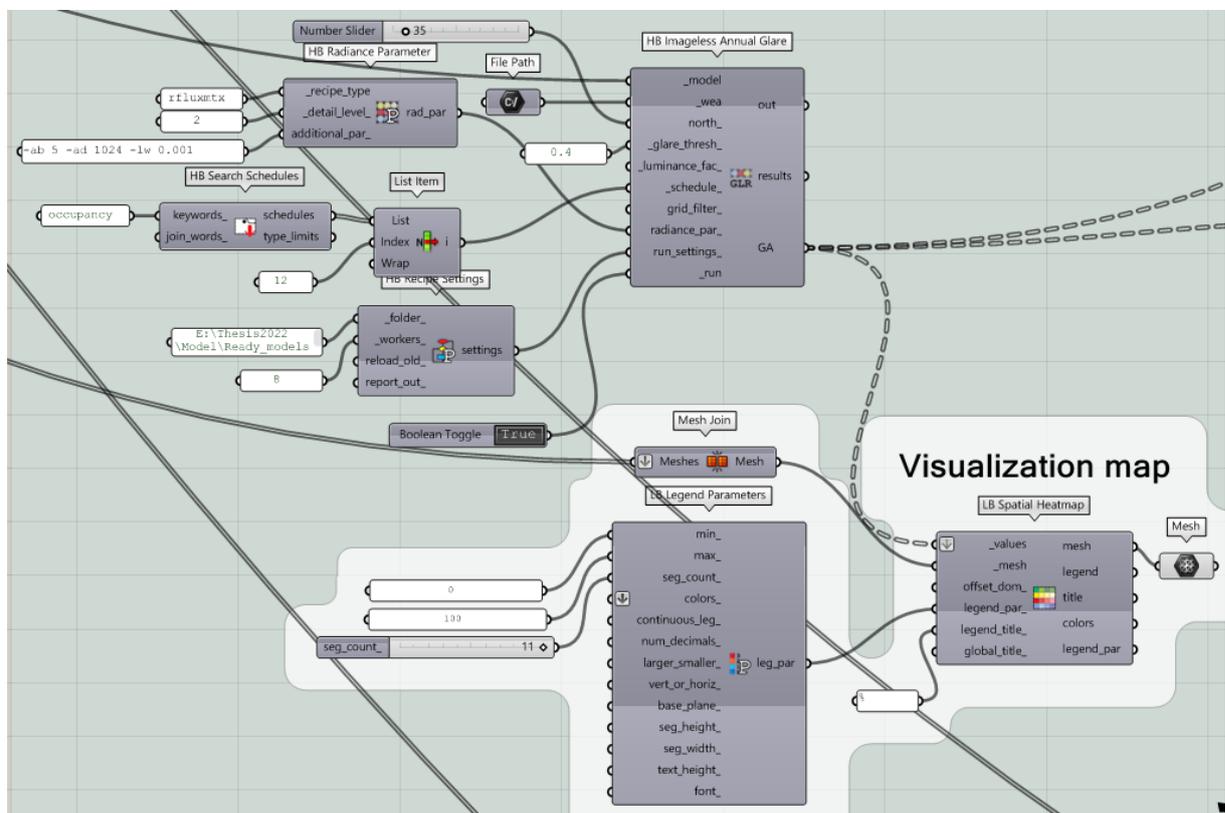


Figure 3.84 Screenshot from Grasshopper UI, Daylight Autonomy, simulation adjustments and visualizing the results II

Since the Daylight Glare Probability (DGP) will be determined from a certain point of view inside the each room, assigning the perspective came first. Figure 3.85 depicts the process of selecting a point from a surface that has already been produced and is up against a wall with a window on it.

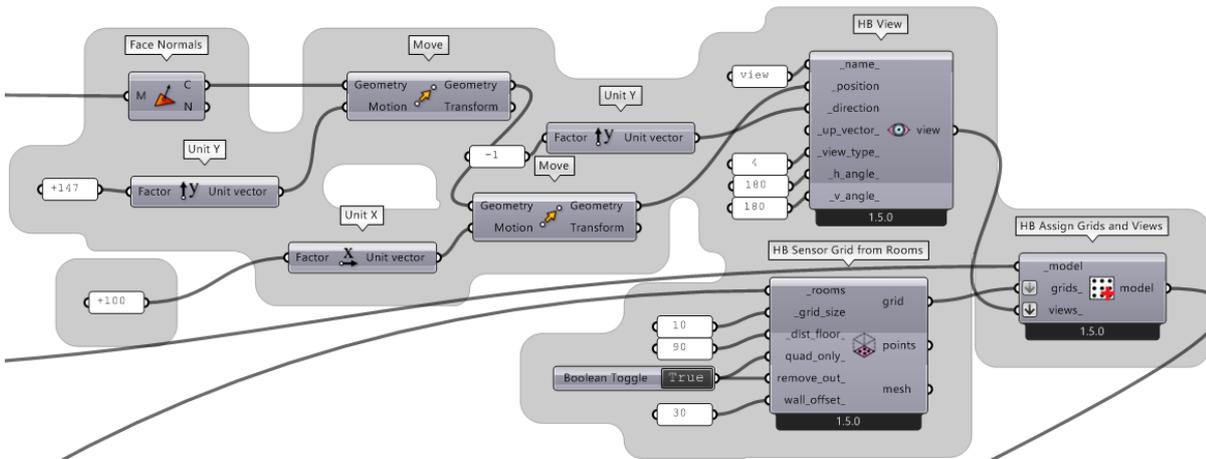


Figure 3.85 Screenshot from Grasshopper UI, Daylight Glare Probability II – Grid

The point was then shifted 147 cm along the y axis in the direction of the window, placing it in the center of the space. Additionally the point was moved on the x axis in order to be positioned in front of the window, but just for the Conference Room. This information was entered into "HB View". In other words, the DGP will be determined and then displayed on a picture. The Angular Fish-eye view type was chosen for the image, and the number of views and their size were both set to 180. The HB Model and the grids were then inserted using the "HB Assign Grids and Views" component once more.

The next step involves using "HB CIE Standard Sky" (Figure 3.86). The plan was to conduct DGP analyses for the equinox, which marks the beginning of spring and fall, and the solstice, which marks the start of summer and winter. A typical radiance sky is produced by this component. This is linked to the "HB Point-In-Time View-Based" component.

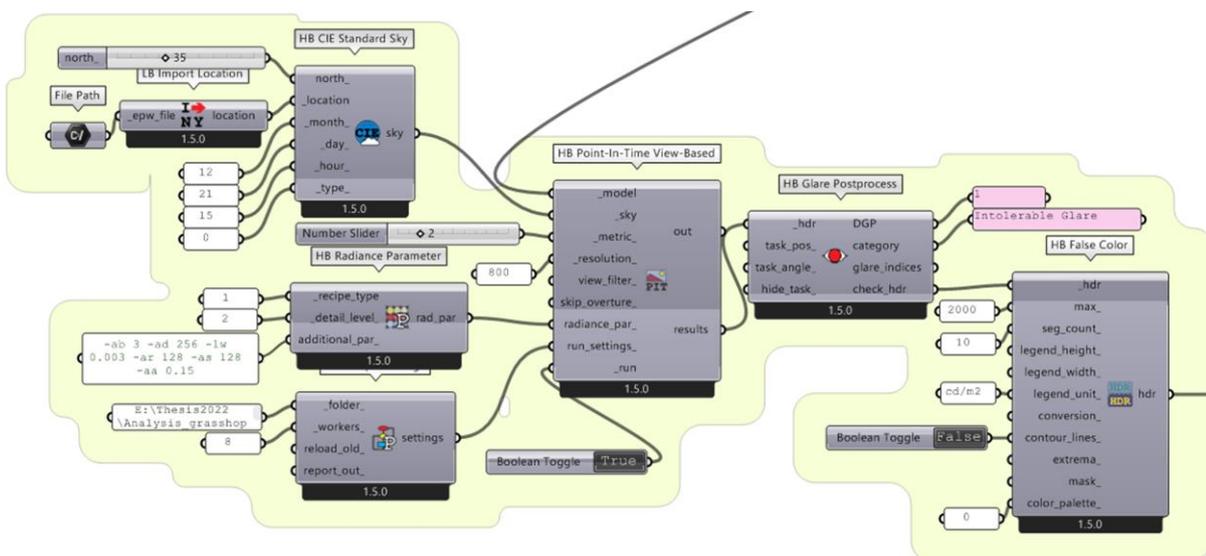


Figure 3.86 Screenshot from Grasshopper UI, Daylight Glare Probability II - Simulation

The HB Model's already-generated sky serves as the necessary input. Radiance and setting parameters were assigned with the appropriate components, and the simulation was then run using "Boolean Toggle: True". In order to determine which category of discomfort it falls under and to calculate the DGP and a few more glare indices, the component "HB Glare Postprocess" was used. Accordingly, this component separates the categories into the four forms of glare: undetectable ( $0.35 > \text{DGP}$ ), perceptible ( $0.4 > \text{DGP} 0.35$ ), disturbing ( $0.45 > \text{DGP} 0.4$ ), and intolerable ( $0.45 \text{ DGP}$ ). Due to runtime restrictions, the given HDR image should be less than 1500 by 1500 pixels. A path to the High Dynamic Range (HDR) image from the glare analysis are presented by the final output, check\_hdr. It is associated with "HB False Color," a tool for transforming HDR images into false color versions. The luminance inputted values were 2000 max cd/m<sup>2</sup> and the legend inputted values were 10 segment counts.

The next step was to input the output of this component into "HB Adjust HDR," which enables exposure adjustments and labeling of the resultant image. In order to simulate what would be perceived with a human eye, a "Boolean Toggle: True" has been added to the "adj\_expos\_" and brightens the image (Figure 3.87).

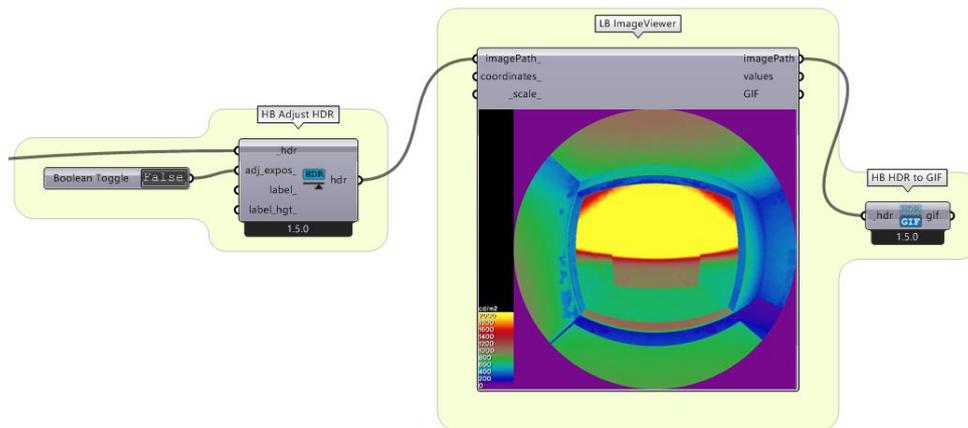


Figure 3.87 Screenshot from Grasshopper UI, Daylight Glare Probability II, producing results

The results from the aforementioned simulation for Glare Autonomy were then used to derive the Spatial Glare Autonomy (sGA). The components that were used to achieve the desired outcomes are depicted in Figure 3.88 and are the same as in the single-zone model, just the results are performed for all rooms together and separately.

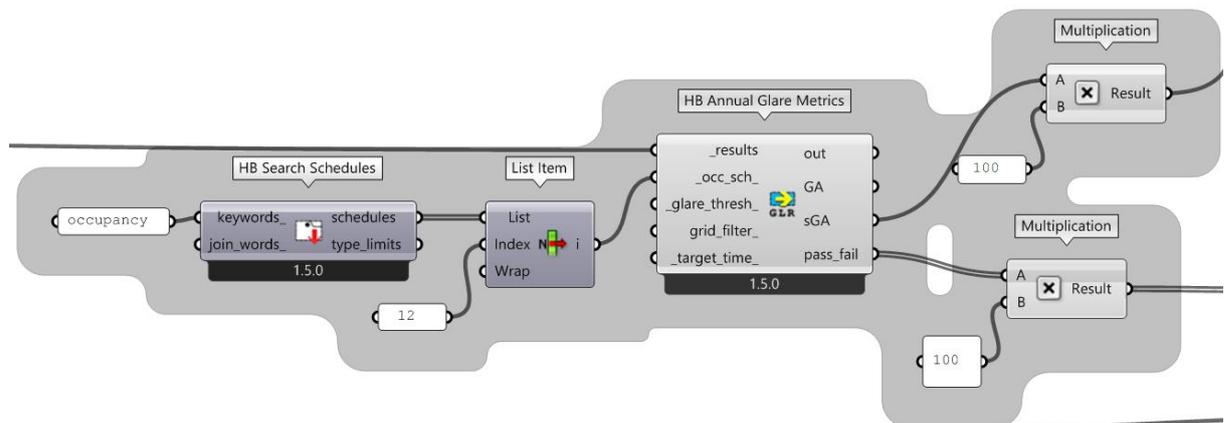


Figure 3.88 Screenshot from Grasshopper UI, Spatial Glare Autonomy II, simulation adjustments

## **4. RESULTS**

This chapter demonstrates the outcomes of using the methods described in the preceding chapter. The process involved simulating various daylight and glare indicators. To produce the results multiple simulations were performed using the Honeybee components from the Grasshopper toolbar. Moreover they were visualized with the Ladybug components. Both of the case studies were tested on two occasions: without any shading device and with applied shading elements. Previously mentioned KPIs were evaluated and later discussed, reviewing the results that were gathered from running the simulations for the single-zone and the multi-zone model. The produced results are visualized in charts, maps, tables, etc. in order to better understand their outcome.

### **4.1. CASE STUDY I – SINGLE ZONE MODEL**

Multiple configurations were inputted in order to perform the simulations for daylight and glare analysis. The components used for the simulations enabled getting the desired results, even though the accuracy of the final outputs could be discussed. Most importantly the workflow with the first type of extraction of the Revit model elements and the algorithm for visual comfort analysis, succeeded in smoothly performing the tests and obtaining values for the above mentioned KPIs, which will be later acknowledged in the following pages.

The results gathered from the simulations performance are various, due to the fact that both scenarios were tested, without blinds and with the dynamic curtain. The different types of representation gives the opportunity to better understand the comparison between having an obstruction situated in front of the window opening and having no shading element.

#### **4.1.1. DAYLIGHT ANALYSIS RESULTS**

With the help of the building performance simulation tools provided within Grasshopper, supplied by Honeybee and Ladybug plugins, multiple key performance indicators for daylight, were evaluated. In the following part of this chapter the results are shown starting with the daylight analysis, simulated on the internal surfaces of the model with specified properties in order to obtain the most accurate results. It is important to mention that in both of the scenarios the only difference was the use and non-use of the shading devices.

The idea behind testing a dynamic curtain was to try and simulate a dynamic operation of an element within the Honeybee Model and later provide the information to the simulation components, which resulted in dynamic motion recognition and different outputs for the various daylight analysis indicators. All results are produced on the sensor grid from the room containing 910 sensor points with a wall offset of 10cm. The height of the grid was situated on a distance of 90cm from floor.

- *Daylight Factor (DF)*

According to the parameters inputted for glazing properties and size of the window opening, the results obtained from Daylight Factor (DF) simulation, were quite low for the requirements for an office room. The ratio between indoor to outdoor daylight illuminance didn't exceed the 3%, on the contrary it resulted as 1.4%, which doesn't provide the desired outcome (*Figure 4.1*).

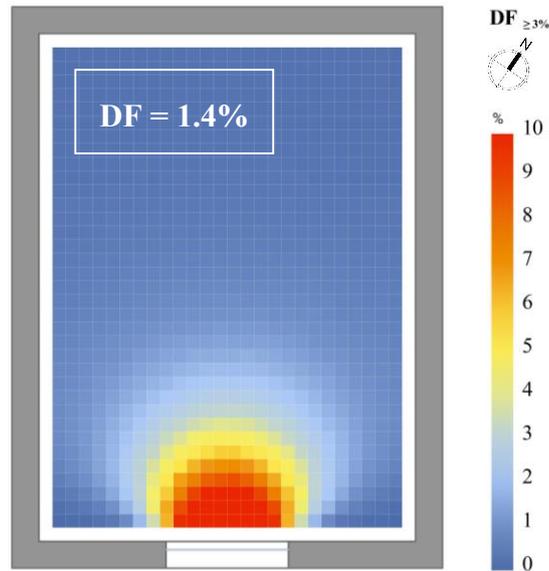


Figure 4.1 Daylight Factor (DF) heatmap for the single-zone model

- *Daylight Autonomy (DA)*
- The results of Daylight Autonomy (DA) within the office room, were outputted as a percentage value and a heat map, for both scenarios.

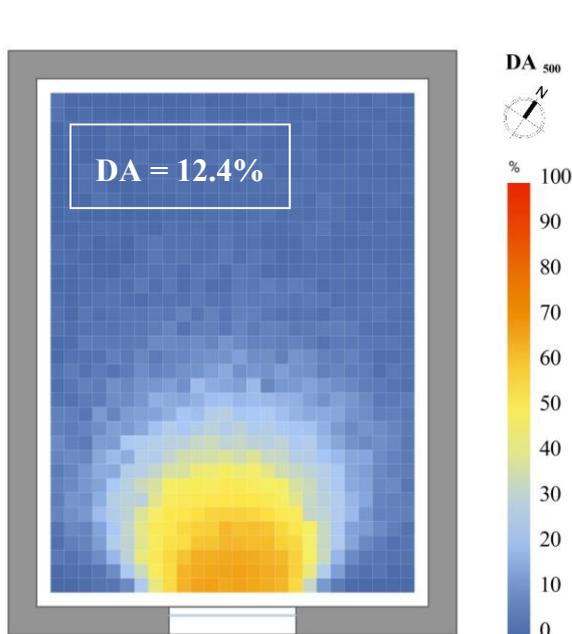


Figure 4.3 DA for Scenario I – no blinds

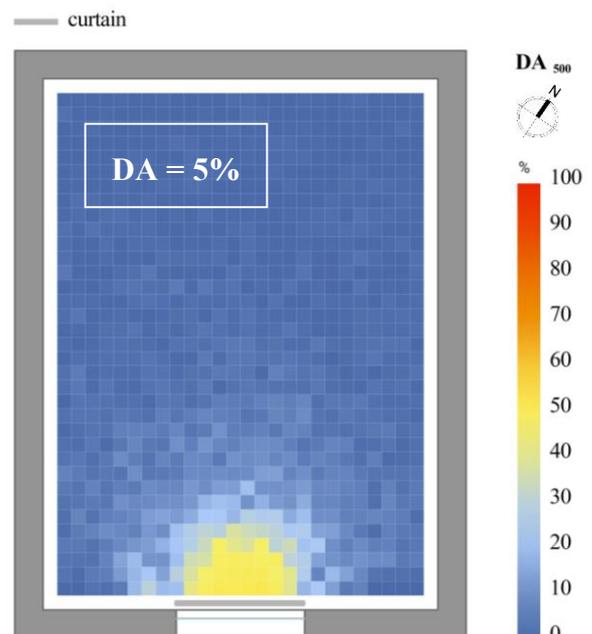


Figure 4.2 DA for Scenario II – dynamic curtain

The results were produced where each point of the sensor grid of the room was examined for an annual period. The average values were obtained from all data, for both first and second case scenario. Observing *Figure 4.2* it can be seen that 12.44% of the occupied hours in the small office room got an illuminance equal or more than the inputted threshold of 500lux. The second case scenario with the curtain blinds (*Figure 4.3*) produced an average value of 5% daylight autonomy. Applying the dynamic curtain is minimizing the daylight autonomy in the small office by more than 50% from the no blinds scenario.

- *Continuous Daylight Autonomy (cDA)*

Continuing with the next indicator, Continuous Daylight Autonomy (cDA), the data outputted from the daylight metrics component was represented in a heat map. The results were retrieved again from all sensor grid points. In this case *Figure 4.5* shows the results for the single zone model without blinds, resulting with 26.1% of cDA and *Figure 4.4* showing a map with again colored grid mesh, where the cDA equals 13.5%. Both obtained values are an average value for all sensor grid points.

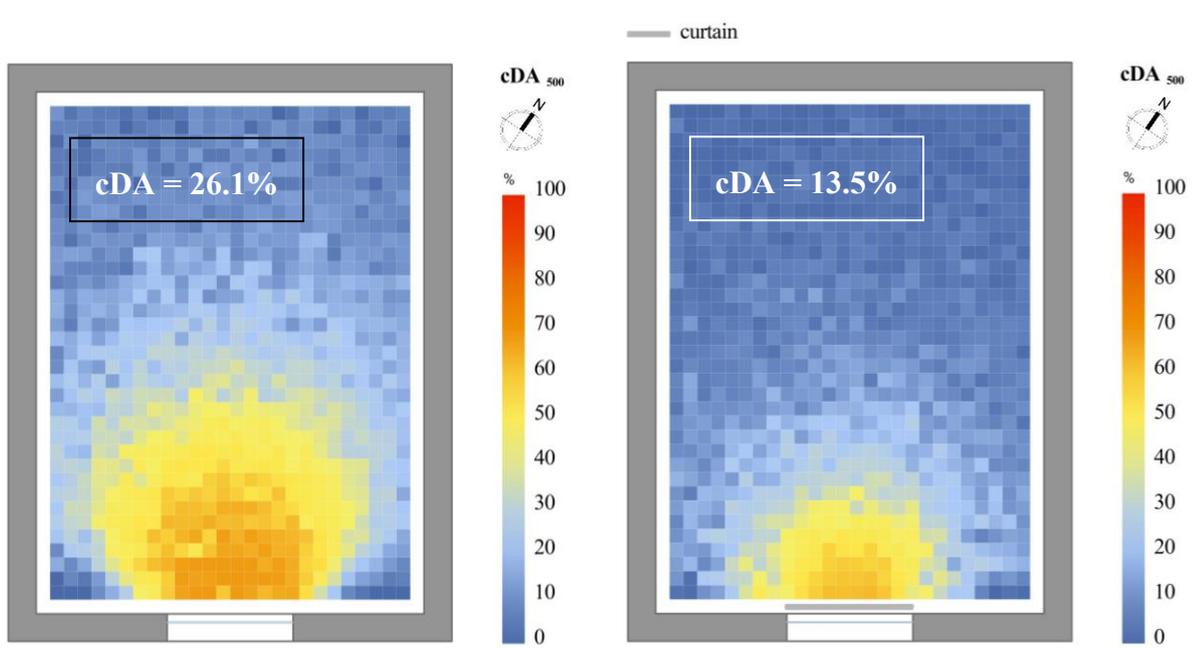


Figure 4.5 cDA for Scenario I – no blinds

Figure 4.4 cDA for Scenario II – dynamic curtain

- *Spatial Daylight Autonomy (sDA)*

With Spatial Daylight Autonomy (sDA) the annual sufficiency of ambient daylight levels in the interior environment of the single zone model was calculated. It outputted a defined percentage of an analysis area (sensor grid from room). For both heat maps representation, the colored map shows two colors, yellow visualizing the grid points that passed the criteria and grey colored mesh, representing the area that didn't meet the minimum daylight illuminance. The result from *Figure 4.7* shows a yellow colored area of 7.8% that meets a minimum daylight illuminance of 500lux for 50 of the operating hours per year. Same for the *Figure 4.6*, the dynamic curtain scenario, the threshold of 500lux was met only for 0.6% of area, which is situated right below the window opening.

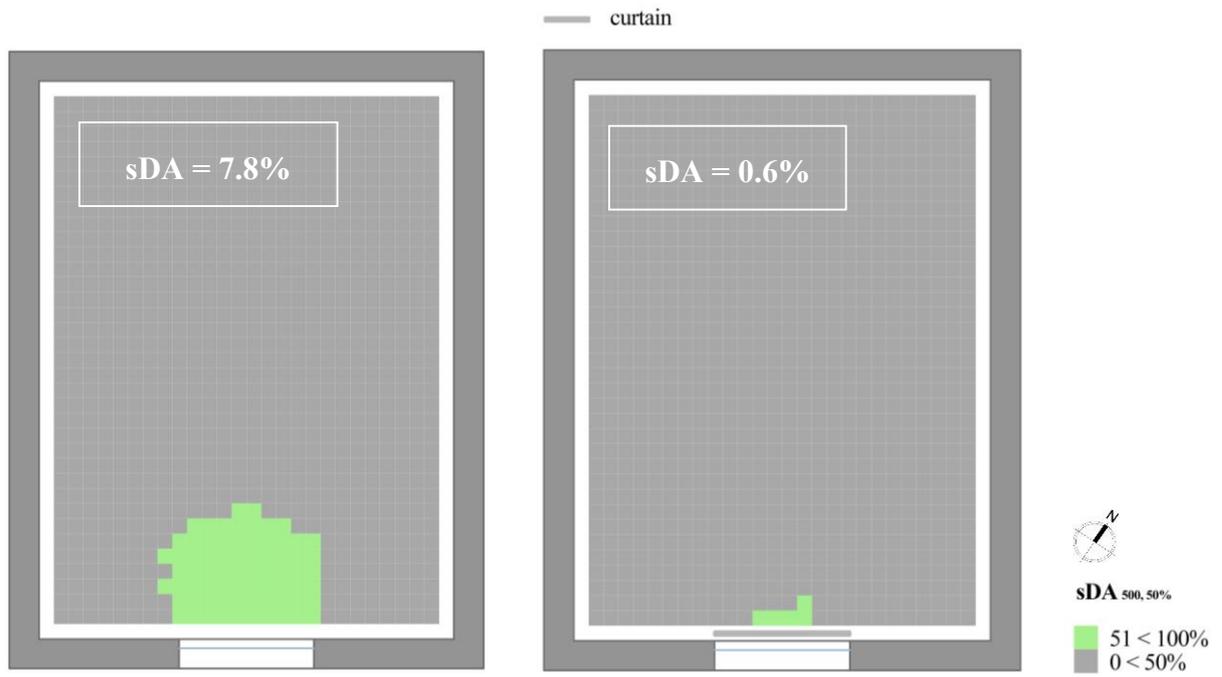


Figure 4.7 sDA for Scenario I – no blinds

Figure 4.6 sDA for Scenario II – dynamic curtain

- Annual Sunlight Exposure (ASE)

Finally, following the sDA, the percentage of space that has 1000 lux for over 250 hours a year was calculated. With the provided algorithm the possibility to give a threshold for the direct sunlight, gave a correct output at the end. The values of Daylight Autonomy were then divided by 100 and then multiplied by the number of occupied hours according to the given schedule for small office and analysis period for a whole year. Then all the values which met the threshold for more than 250 hours were divided by the points of the evaluating surface. The presented heat map (*Figure 4.8*) shows the red colored area of 9%, where the grid points received 1000 lux for over 250 hours annually.

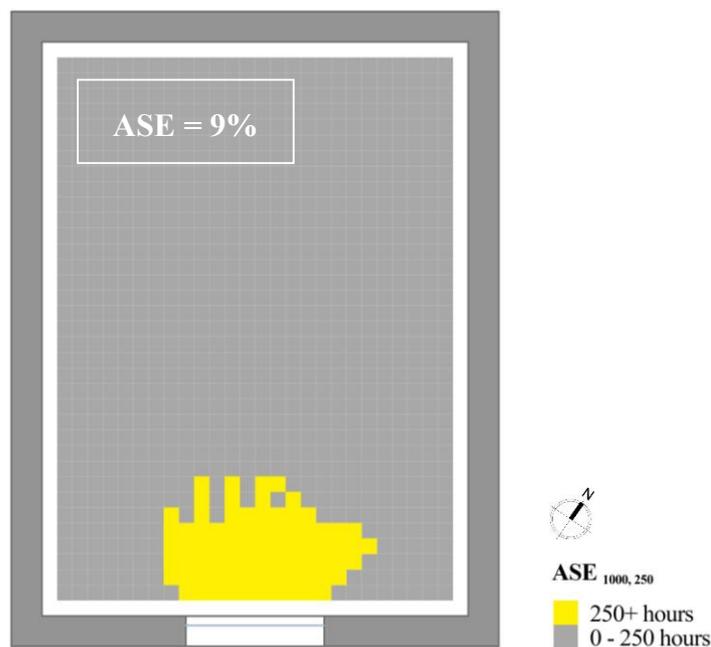


Figure 4.8 ASE for Scenario I – no blinds

- *Useful Daylight Illuminance (UDI) fell short (0-100 lux)*

The results of Useful Daylight Illuminance (UDI) fell short, represent the insufficient illumination, which means that the average value is the percentage of time that the illuminance fell between 0 and 100 lux. As shown on both *Figure 4.10* and *Figure 4.9*, the resulted colored heat maps represent the highlighted area with range of colors pointing out the different percentage of time. In the small office room with no blinds UDI (0-100 lux) average value is 60.1% and with dynamic curtain is 81.6%. These results can be explained as following, the second scenario (dynamic curtain) is obviously containing a larger area with illuminance of 0-100 lux, due to the curtain obstruction.

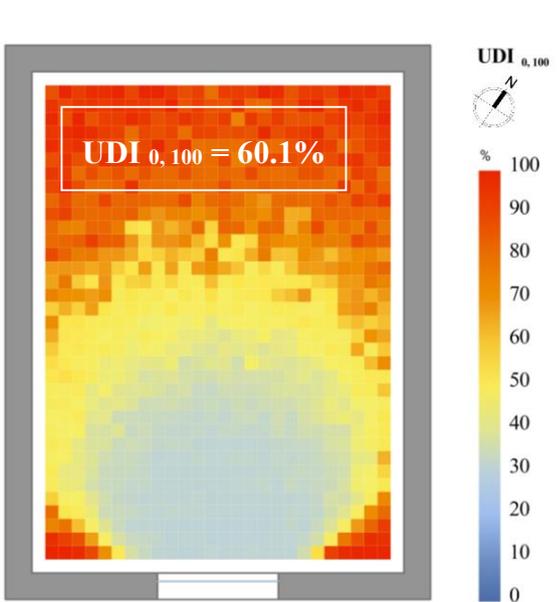


Figure 4.10 UDI (0-100 lux) for Scenario I – no blinds

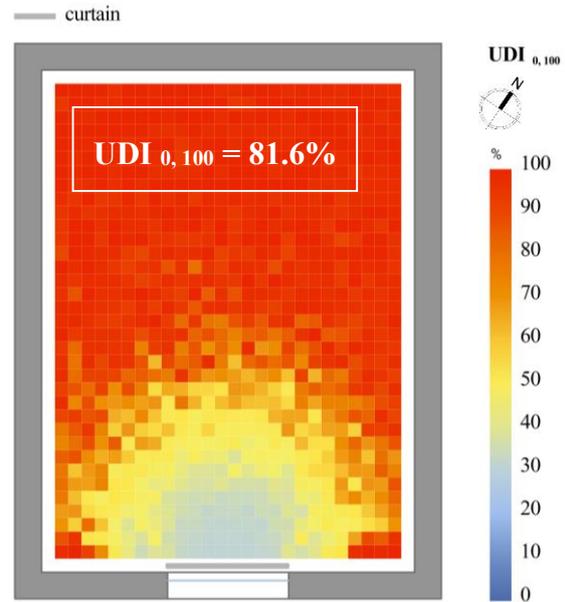


Figure 4.9 UDI (0-100 lux) for Scenario II – dynamic curtain

- *Useful Daylight Illuminance (UDI) supplementary (100-500 lux)*

Second type of Useful Daylight Illuminance (UDI) is the supplementary one, which has the lower and upper thresholds as 100-500 lux of illuminance, shown on *Figure 4.11* and *Figure 4.12*.

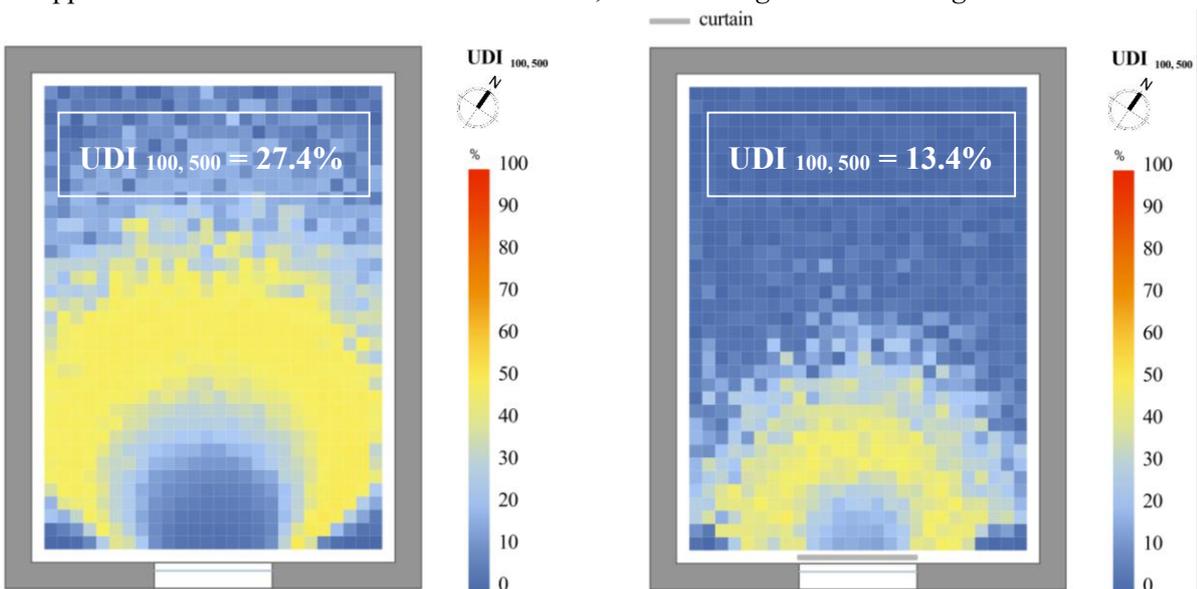


Figure 4.12 UDI (100-500 lux) for Scenario I – no blinds

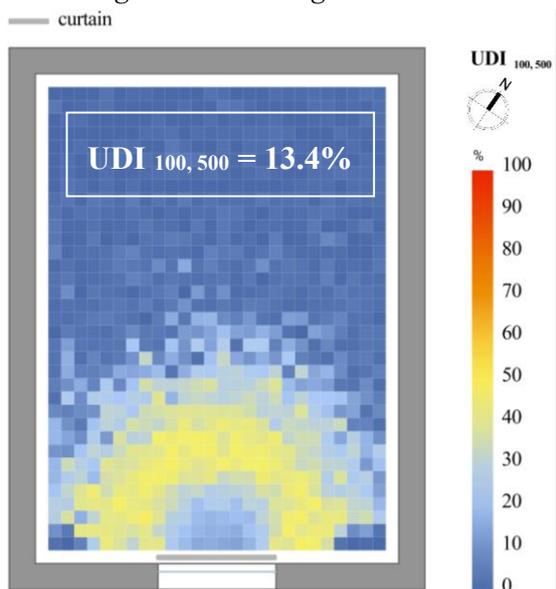


Figure 4.11 UDI (100-500 lux) for Scenario II – dynamic curtain

The supplementary UDI in the first case without blinds shows an average result of 27.4% of the analyzed area and the second one with 13.4%.

- *Useful Daylight Illuminance (UDI) autonomus (500-3000 lux)*

In the third category of Useful Daylight Illuminance (UDI) autonomus, the upper and lower thresholds were set between 500-3000 lux of illuminance. The average value of 9.8% (Figure 4.13) and 4.5% (Figure 4.14) represents how big is the area where the illuminance is meeting the values from 500-3000 lux.

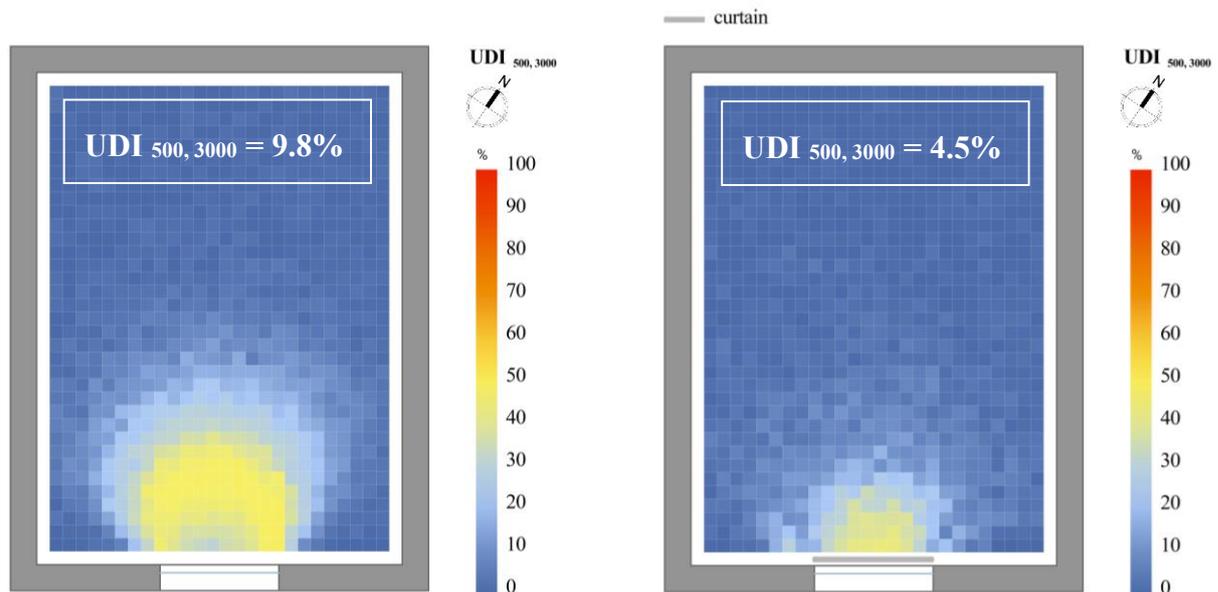


Figure 4.14 UDI (500-3000 lux) for Scenario I – no blinds Figure 4.13 UDI (500-3000 lux) for Scenario II – dynamic curtain

- *Useful Daylight Illuminance (UDI) exceeded (>3000 lux)*

Finalizing the Useful Daylight Illuminance (UDI) calculations, the last results were the ones where the illuminance exceeded 3000 lux. Figure 4.16 shows the average value of 2.7% and Figure 4.15 shows for the dynamic curtain scenario 0.5% of the area where the illuminance is over 3000 lux.

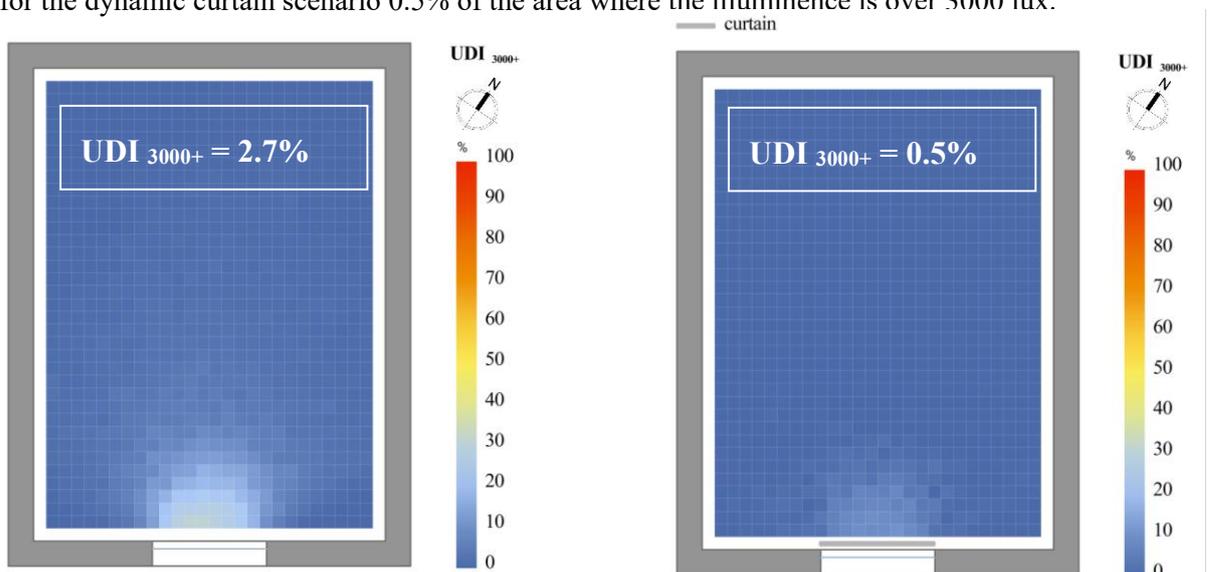


Figure 4.16 UDI (>3000 lux) for Scenario I – no blinds

Figure 4.15 UDI (>3000 lux) for Scenario II – dynamic curtain

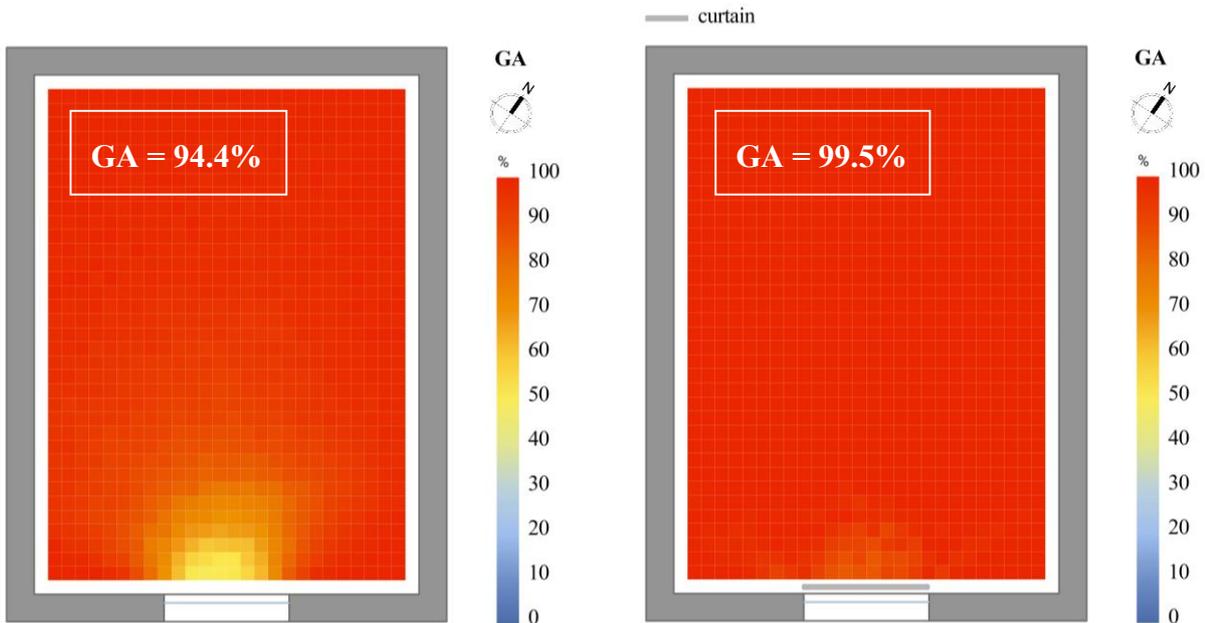
## 4.1.2. GLARE ANALYSIS RESULTS

The recipe for producing the glare results uses an image-less glare method. Running the simulations for glare analysis with the help of the Honeybee component “HB Imageless Annual Glare”, gave annual results from which the Glare Autonomy was evaluated. This was done for both case scenarios for the single-zone model.

Firstly simulating the single-zone model without blinds and then simulating the model with dynamic curtain shading element. The results are going to be shown in the following pages, represented with heat maps and their average values for all the sensor grid points from the room. Another glare analysis was done and the results for Daylight Glare Probability (DGP) for a single point situated in the middle of the room, with a direction towards the window opening, were obtained with values and images.

- *Glare Autonomy (GA)*

Presenting the percentage of occupied hours that are free of glare, Glare Autonomy (GA) simulations were performed with a threshold of 0.4 DGP above which conditions are considered to induce glare. The results bellow are shown on *Figure 4.17* for the single-zone model free of blinds and *Figure 4.18* with the application of dynamic curtain.



*Figure 4.17 GA for Scenario I – no blinds*

*Figure 4.18 GA for Scenario II – dynamic curtain*

The obtained results from the simulations made on both scenarios show that the single-zone model excluding any constructed shades is having an average value of 94.4% of occupied hours in the total grid area which are free of glare.

Secondly, the dynamic curtains obstructed the existence of glare inside the room, which resulted with 99.5% of total occupied hours to be free of glare.

Overall it is noticeable that the glare is almost totally stopped with the dynamic curtain and the comments on the obtained results are represented in the next few pages.

- *Daylight Glare Probability (DGP)*

This indicator was used to evaluate glare from daylight. Daylight Glare Probability (DGP) is the fraction of people disturbed by daylight glare source and within the results it is expressed from 0 to 1. The results were obtained for both scenarios, no blinds and dynamic curtain. They are represented for both solstice and equinox, meaning that the evaluated periods are June 20<sup>th</sup> and December 21<sup>st</sup>, three times a day during the working hours at 9am, 12pm and 3pm.

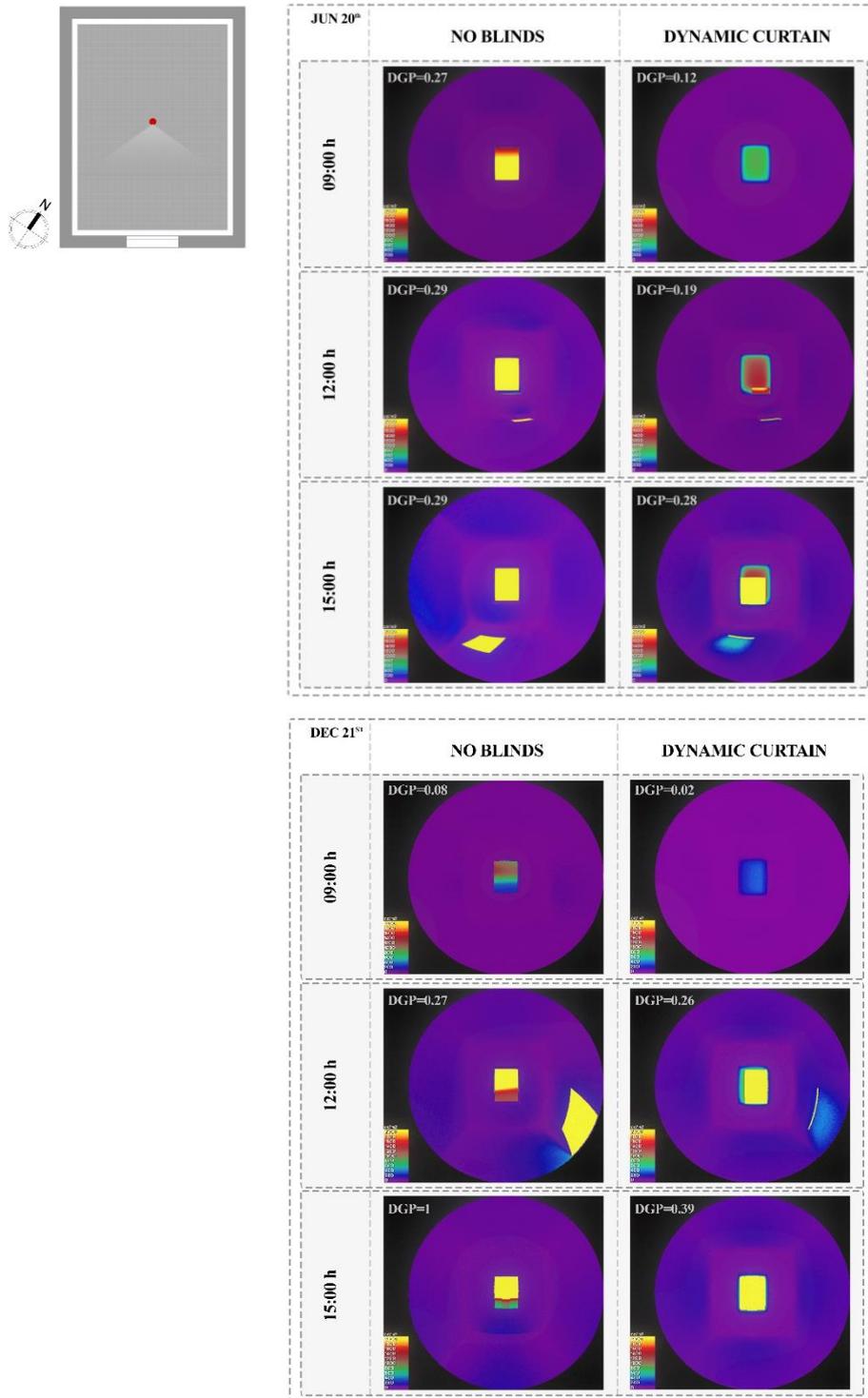


Figure 4.19 DGP results for single-zone model

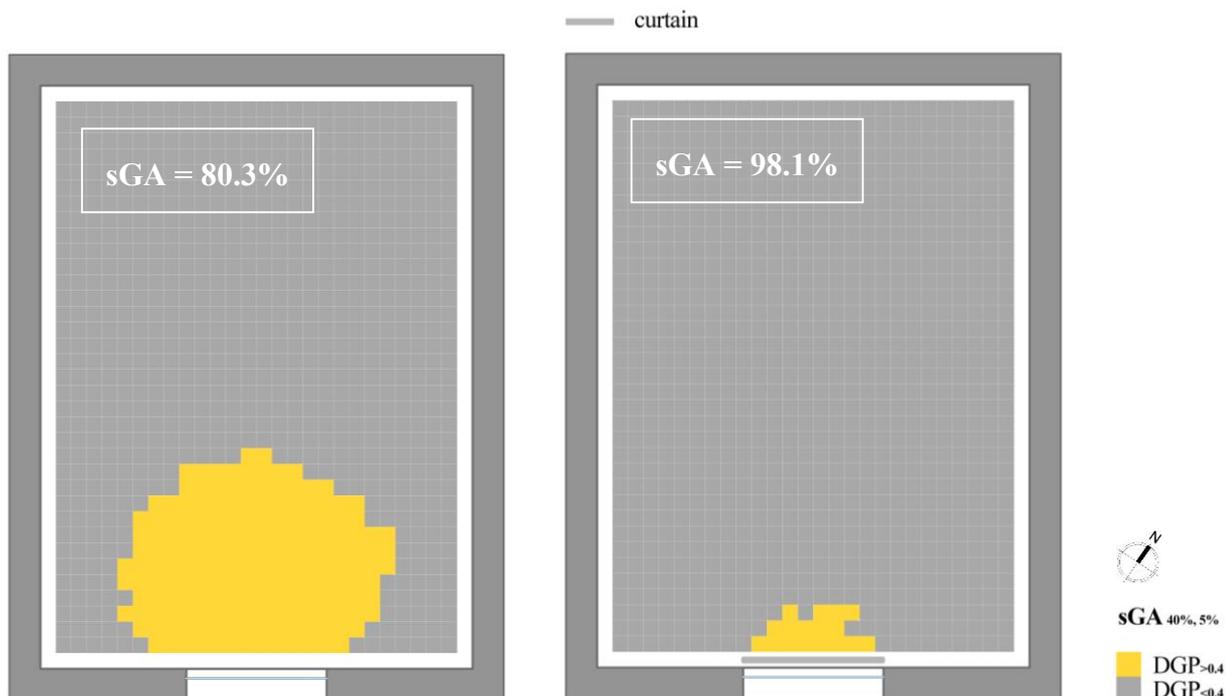
The DGP results shown on *Figure 4.19* show that the dynamic curtain is working and it's blocking the glare which could increase up to 0.4 disturbing the people in the office. Since the point for the view was taken in the middle of the room, the examined direction shows how the glare is influencing the situation in the room. It is important to mention that for these simulations the radiance parameters were lowered in order to produce results which could be simulated faster than the once that are with higher performances. Of course these results bring accurate values for DGP from the performed simulation, rendering an image with color's range for luminance from 0 to 2000 cd/m<sup>2</sup>, which presents more accurate image where the range of colors can be visibly noticed on the surfaces.

- *Spatial Glare Autonomy (sGA)*

The last simulation outputs were gathered and presented in a heatmap from the Spatial Glare Autonomy (sGA). *Figure 4.20* shows the single zone model without any shading elements and it results in 80.3%, which means that it shows the percentage the tested grid points that they are below the given threshold for DGP which is 0.4. It means that for the targeted time of 5% the area is free of glare, but for a minimum percentage of the targeted time.

Following the second heatmap shown on *Figure 4.21* represents the single zone model with the dynamic curtain. It can be seen that the percentage is higher and that the dynamic curtain is functioning in a way of stopping the direct sun rays which will produce glare over 0.4 to enter the room. The spatial glare autonomy results in 98.1%, which means that almost all area of the room is free of glare for a minimum of 5% of the occupied hours.

The yellow colored squares show when the target value of 0.4 is surpassed and the grey squares are presenting which point received glare lower than 0.4.



*Figure 4.21 sGA for Scenario II – no blinds*

*Figure 4.20 sGA for Scenario II – dynamic curtain*

### 4.1.3. COMMENTS ON THE RESULTS

The following pages represent tables and graphs which are showing the difference between two types of shading technology, no blinds and dynamic curtain. *Table 4.1* shows the mean value for all KPIs for the single-zone model.

Shading element type	Key Performance Indicators (KPIs) (Mean value) [%]										
	DF	DA	cDA	sDA	ASE	UDI <sub>full-short</sub> (0-100 lux)	UDI <sub>supplementary</sub> (100-500 lux)	UDI <sub>autonomous</sub> (500-3000 lux)	UDI <sub>exceeded</sub> (>3000 lux)	GA	sGA
No shading element	1.4	12.4	26.1	7.8	9	60.1	27.4	9.8	2.7	94.4	80.3
Dynamic shading element	/	5	13.5	0.6	/	81.6	13.4	4.5	0.5	99.5	98.1

*Table 4.1 KPIs Results for the Single-zone model, both without shading element and dynamic curtain*

To start with the daylight factor resulted from the simulation performed on the single zone model without blinds, it is noticeable that it is way below the requirement for an office room. A value over 3% daylight factor is not reached and when observing the whole geometry and the size of the window, it is evident that the window is 1/10 of the floor area, which is not acceptable.

The following results that were retrieved from the single-zone model in both case scenarios, had some differences. Firstly, it means that the dynamic curtain shades were constructed correctly and secondly they are working properly resulting with outputs that vary between the two states.

Multiple comments could be added on the heat maps that were produced from daylight autonomy simulations. For example the average daylight autonomy in the room is not even close to 50%, but the requirement stresses a target value which is equal or higher than 50%. The average value for the single-zone model with dynamic curtain is 5% and it is more than two times lower comparing to the single-zone model without any shading element obstructing the window, which resulted with 12.4%.

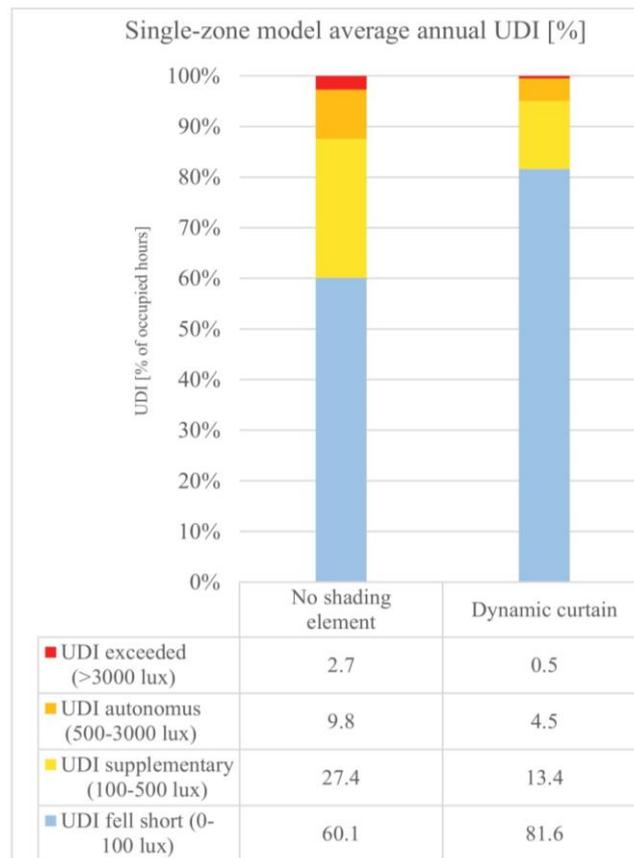
Moreover the results of 26.1% and 13.5%, for both case scenarios, from continuous daylight autonomy are also not reaching the targeted value of larger than 55%, given by IES LM-83-12 (reference for Lighting Measurement). It can be seen that all these results are showing that the window opening is small and all of the area that is far from the window is not lighted enough.

The spatial daylight autonomy as explained in the above mentioned KPIs table, has three divisions for target values. This means that the result of 7.8% and 0.6% is insufficient. But the annual sunlight exposure has a value of 9% which has passed the Leadership in Energy and Environmental Design (LEED) requirements for the percentage of area that has 1000 lux for over 250 hours on annual basis.

Finally the useful daylight illuminance results for all four categories, simulated on the single-zone model with dynamic curtain, were again not meeting the requirements. This means that if the useful daylight illuminance between 100-3000 lux is over 80% of the occupied time in a tested area, it reached the target value.

Comparing all UDI results, it can be seen that most of the area is getting 0-100 lux during the occupied hours. But overall all the divided categories had an expected values, but it is worth to mention that the dynamic curtain helped in creating a pleasant environment.

The following chart on *Figure 4.22* shows the stacked UDIs, resulting in 100%, which gives a better representation of the differences between the single-zone model without blinds and with the dynamic curtain.



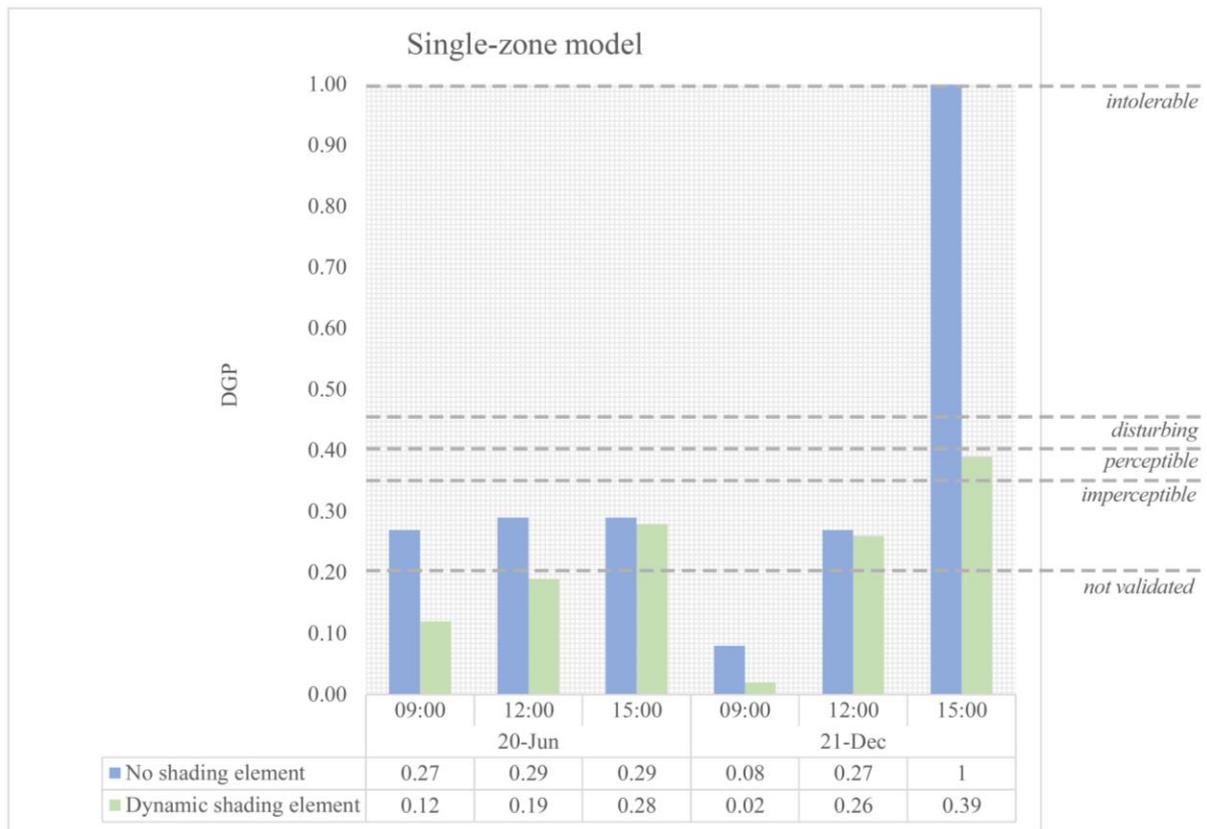
*Figure 4.22 Single-zone model average annual UDI*

From the chart it can be seen that the occupied hours where the illuminance is below 100 lux are with the most consistent. This means that this room would probably need an electric lightning during most of the occupied hours, which is not the best result that was desired, but it shows that two of the most important things when performing daylight analysis to pay attention to are the size of the glazing element and the performance of the glass.

Continuing with the glare analysis, again the dynamic curtain successfully stopped almost the total disturbing glare that could happen within the office room. It can be seen on Table 4.1 that the glare autonomy of 94.4% for the single-zone model was the outcome of not having blinds. But even though there isn't a large value where the glare is exceeding 0.4, the result of 94.4% is a really high value which maybe is not needed. The result with an average of 99.5% of occupied hours which are free of glare (as mentioned previously tested with a threshold of 0.4 - disturbing glare), is showing that most of the tested points are below 0.4 DGP and that the dynamic curtain stopped the possible intolerable glare.

Second to last is the spatial glare autonomy which results with again a high value for the percentage of area that is free of glare, not exceeding the threshold of disturbing glare of 0.4 for at least 5% of the occupied hours. The single-zone model without blinds results with sGA of 80.3% and with the dynamic curtain sGA of 98.1%.

At the end the daylight glare probability was a tricky indicator to calculate, even with lower radiance parameters, just to get a result for the specific point where it has been tested. The results are shown on the chart presented on *Figure 4.23*. Also the thresholds are set as a limitations, with whom the glare range is shown.



*Figure 4.23 Single-zone model DGP for June 20th and December 21st, at 9pm, 12pm and 3pm*

On the other hand the actual values that were processed within the room resulted with an imperceptible glare most of the time and few times it was perceptible or disturbing. As seeing the results it could be commented on the seasonal calculations, for example normally when the sun is lower on the horizon the glare in the room could be increased during the day. This happens mostly at the end of the day, especially at 3pm, because the window and the view are oriented on the South West.

Due to the north rotation for 35° for the whole room, the window without blinds, in summer doesn't get direct sunlight resulting with glare over 0.3. Instead the results during the winter, are having a higher value for the DGP, again especially at 3pm. It is interesting to point out that at 3pm on 21<sup>st</sup> of December a DGP of 1 happened. This is due to the direct sunlight towards the simulated point and its view. But again with the dynamic curtain, the value was dropped to 0.39, where glare is perceptible but it is not disturbing.

To conclude the overall comments on the results, the simulations performed on the single zone model in both case scenarios, resulted with average values that didn't reach all the target values, but they show that the implementation of dynamic curtain works.

## **4.2. CASE STUDY II – MULTI-ZONE MODEL**

The simulation's components made it possible to get the expected results, even though there was some debate over the final outputs' accuracy. The simulations for the examination of daylight and glare were run using a variety of parameters. Most importantly, the workflow using the second type of Revit model room extraction and the visual comfort analysis algorithm was successful in carrying out the tests and obtaining values for all zones included in the multi-zone model for the aforementioned KPIs, which will be acknowledged in the following pages. Due to the simulations' testing of both scenarios—without blinds and with louver shades at an angle of 30 degrees—the results obtained from their performance are varied. The various representational styles provide for a better understanding of the contrast between a window opening with an impediment in front of it and one without any shading.

### **4.2.1. DAYLIGHT ANALYSIS RESULTS**

Numerous critical performance indicators for daylight were assessed using the building performance simulation tools available in Grasshopper, delivered by Honeybee and Ladybug plugins. The results are presented in the section that follows this chapter, commencing with the simulation of the daylight analysis on the internal surfaces of the model with the desired parameters. It's vital to note that the sole difference between the two situations was whether or not the shading devices were used. The purpose of testing louver shades with an inclination of 30 degrees, was to attempt to see the differences of the results and if there are any changes in the daylight analysis. The simulation components used the information from the generated louver blinds and various outputs for the different daylight analysis indicators.

The difference from the single zone model was that there were multiple zones on which the simulations were performed. All simulations were ran once for all rooms and then the results could be retrieved separately for each room. The sensor grid was inputted for all rooms, which resulted with 8819 sensor points in total and a 10 cm offset from the wall was inputted. The height of the grid was 90 centimeters above the ground. All results are presenting an annual average value for each of the 10 zones in the multi-zone model. Lastly an overall average value was calculated in order to see the whole building performance.

What is important is that the simulation results could be shown separately for each room in order to see how to improve either the glazing size or the properties of the glass material. All this is mentioned for further analysis or additional observation of the results for the simulations.

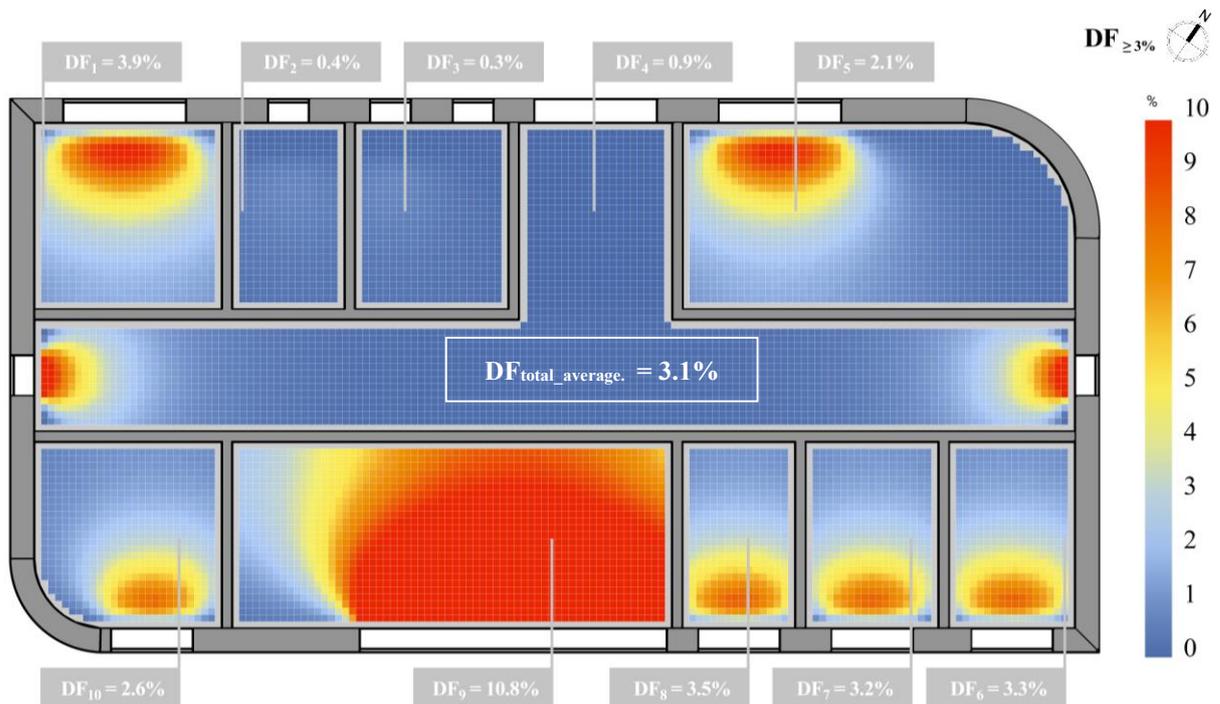
Just one key performance indicator, which is the DGP was calculated for two rooms and the results are shown with images that present the colored surfaces within the two rooms for 2 different dates and seasons during the year, at three different time intervals.

- *Daylight Factor (DF)*

As in the first case study, where the single zone model was tested, in the multi-zone model the daylight factor was evaluated within the building without blinds. From the overall simulation, each room produced a heat map, where the outputted colors indicate the percentage of daylight factor on the imported sensor grids. An average value was obtained for all building, also the rooms had their average value for daylight factor, retrieved separately.

The glazing properties have the same values for each window, just their size is different and also the orientation on the facades. The g value is 35% and the U value of the glass is 0.4., with a solar transmittance of 28%. It is noticeable that the ones which are on the south west façade receive more light and also the ratio between indoors and outdoors illuminance is higher.

In the following heat map seen on *Figure 4.24* all rooms' results are represented. The average value for the whole building daylight factor was 3.1% and the room with highest average value is the Conference Room (10.8%), which is number with DF<sub>9</sub>. This is due to the curtain wall which has the biggest size compared to the other window openings in the building.



*Figure 4.24 Daylight Factor (DF) heatmap for the multi-zone model*

With lowest average daylight factor value are the Utility room and Restroom with 0.3%-0.4%. The Entrance also has a low value, just because in this project a wooden door is used and the two windows that are on opposite facades, North West and South East, resulted with a daylight factor value of 0.9%.

More on this matter is mentioned in the following pages, where comments on the results are given and there is an explanation in detail on how the performance of the glass or its size can change the daylight factor value.

- *Daylight Autonomy (DA)*

For both scenarios, the Daylight Autonomy (DA) results in the offices were outputted as a percentage value and a heat map. A year's worth of data from the rooms' sensor grid was analyzed to get the results. The average values for the first and second case scenarios were determined from all the data. According to *Figure 4.25*, an average value of 28.1% of the occupied hours in the whole building had illumination that was equal to or more than the specified threshold of 500 Lux. The average values of the rooms facing south-east and south-west are much higher than the others, which means that the values are between 30% and 60%.

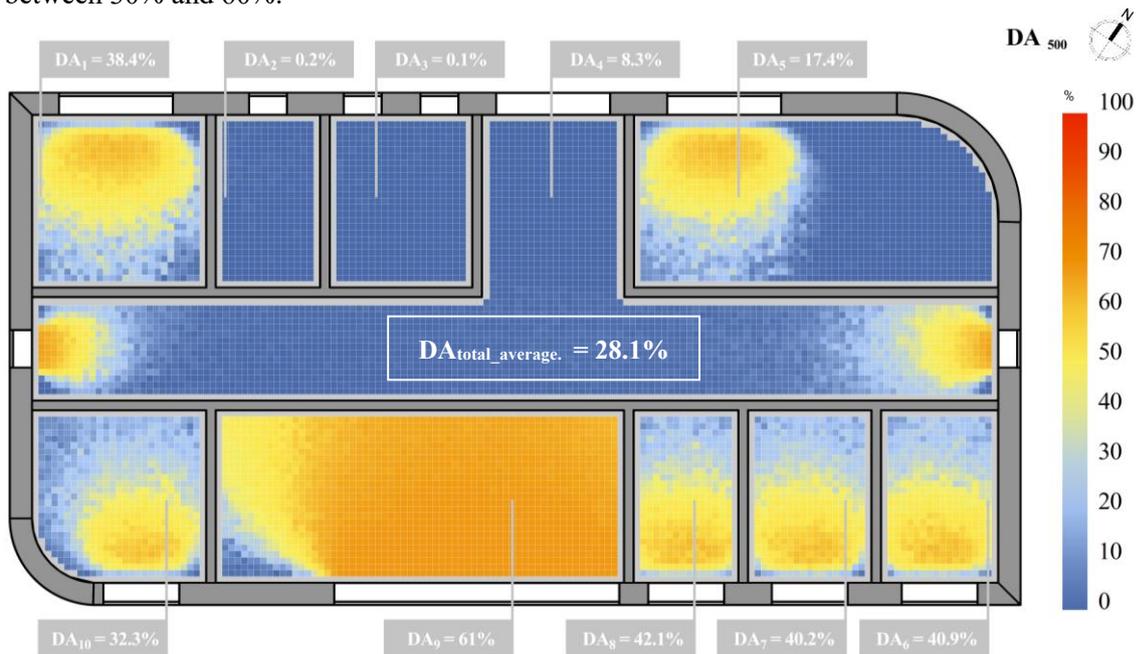


Figure 4.25 Daylight Autonomy (DA) heatmap for the multi-zone model, no blinds

A value of 3.1% daylight autonomy was obtained on average in the second case study with the louver blinds. Applying the louver blinds reduces the offices' daylight autonomy by more than 80% when compared to the case when there are no shades (*Figure 4.26*).

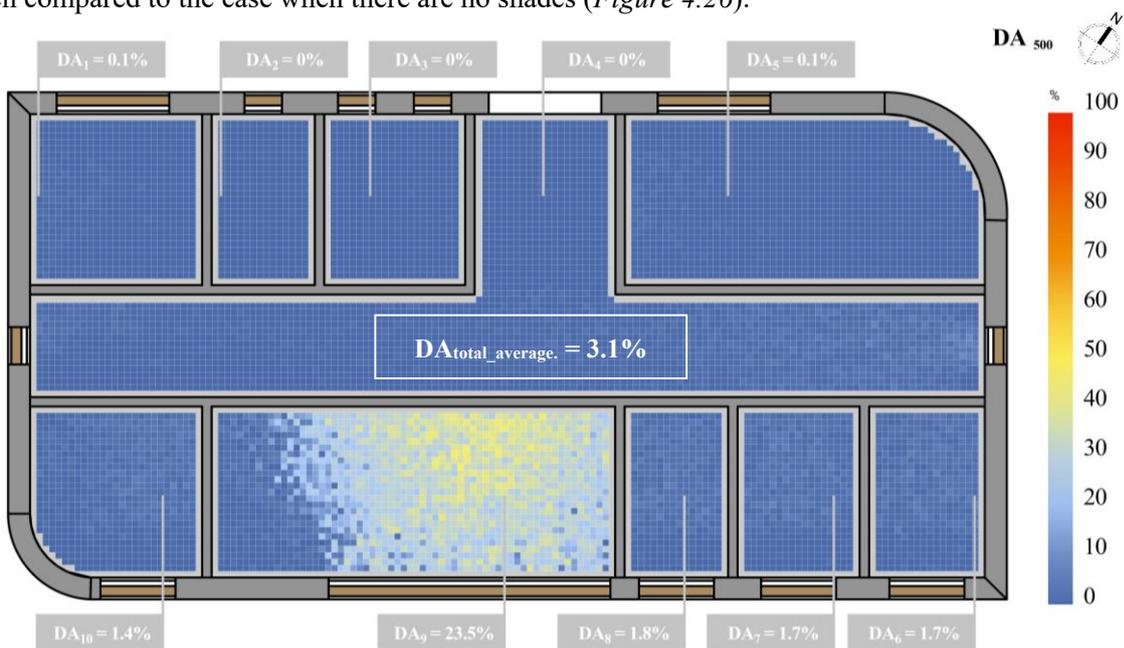


Figure 4.26 Daylight Autonomy (DA) heatmap for the multi-zone model, louver shades

- *Continuous Daylight Autonomy (cDA)*

In order to better understand the data provided from the daylight metrics component's following indicator, Continuous Daylight Autonomy (cDA), a heat map was created. *Figure 4.27* in this instance displays the results for the multi zone model without blinds, yielding an average cDA of 40.8%.

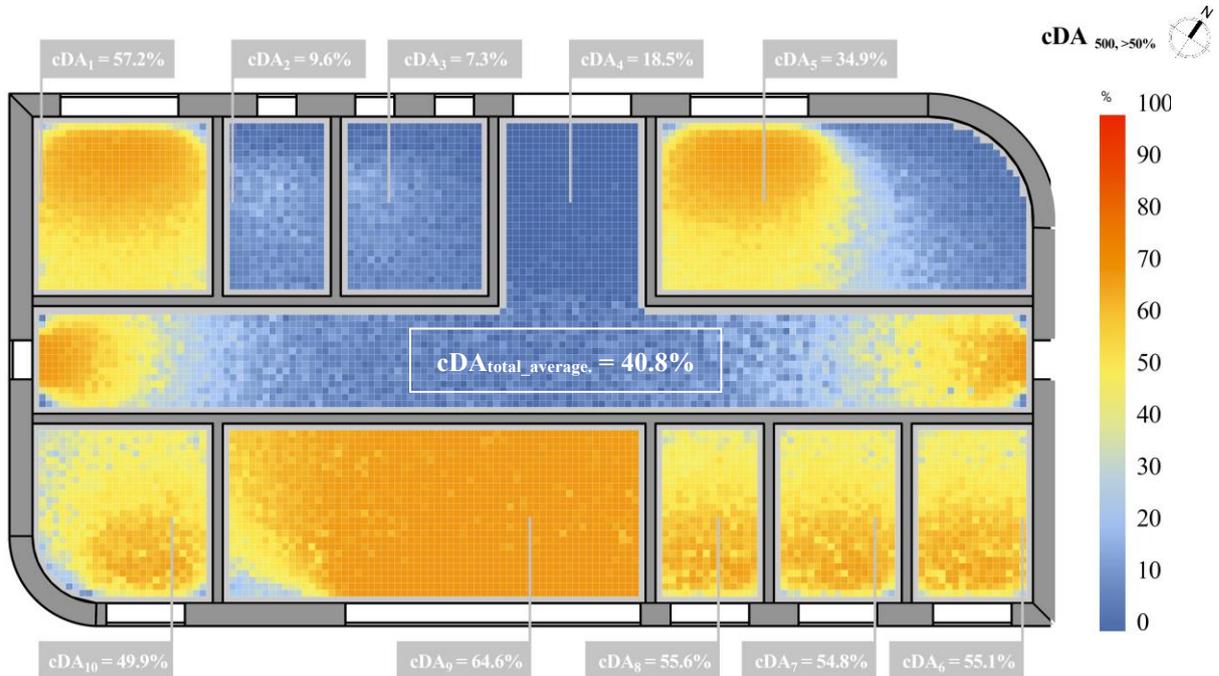


Figure 4.27 Continuous Daylight Autonomy (cDA) heatmap for the multi-zone model, no blinds

*Figure 4.28* displays a map with a colored grid mesh once more, with an average cDA of 11.1% for the whole model with louver blinds. It is obvious to say that the blinds function in order to simulate a lower value for continuous daylight autonomy for all rooms inside the building.

- *Spatial Daylight Autonomy (sDA)*

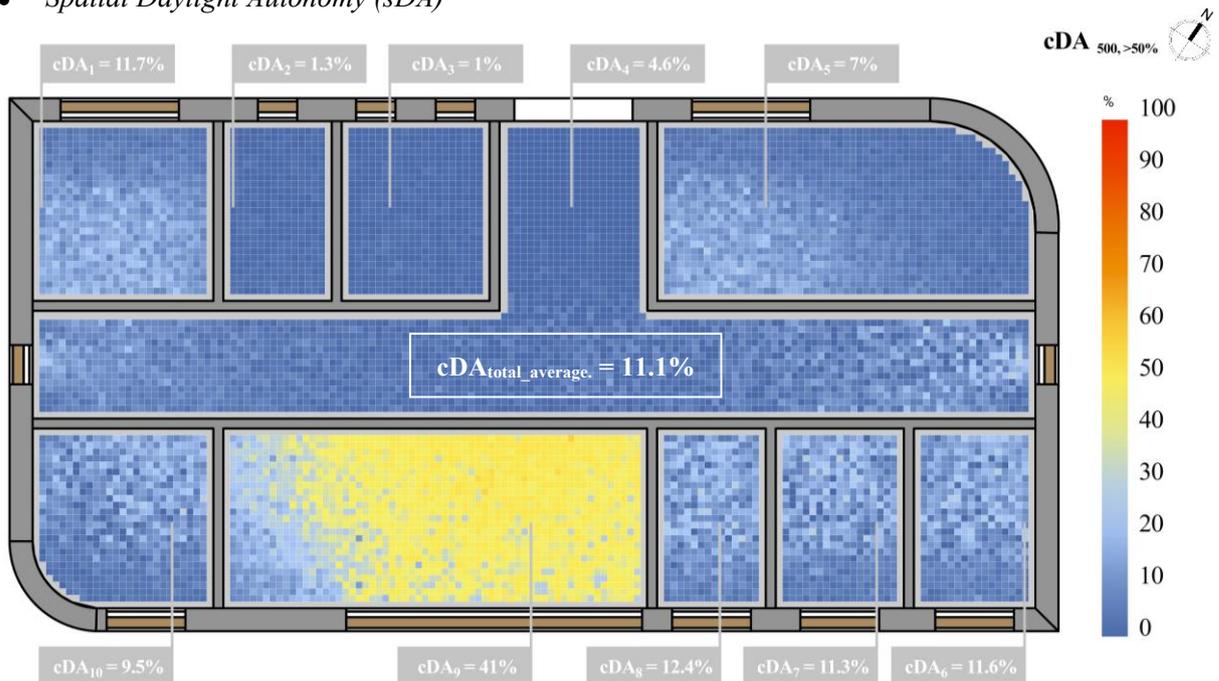


Figure 4.28 Continuous Daylight Autonomy (cDA) heatmap for the multi-zone model, louver shades

The annual sufficiency of ambient daylight levels in the interior environment of the multi zone model was determined using spatial daylight autonomy (sDA). A specified proportion of the analysis area was outputted (sensor grid from each room). The colorful heat map displays two colors for each heat map representation: green for grid points that met the criterion and grey for areas that fell short of the required level of daylight illumination. *Figure 4.29* conclusion depicts a 27.4% green region that, for 50% of the operating hours per year, achieves a minimum of 500 lux of daylight illumination.

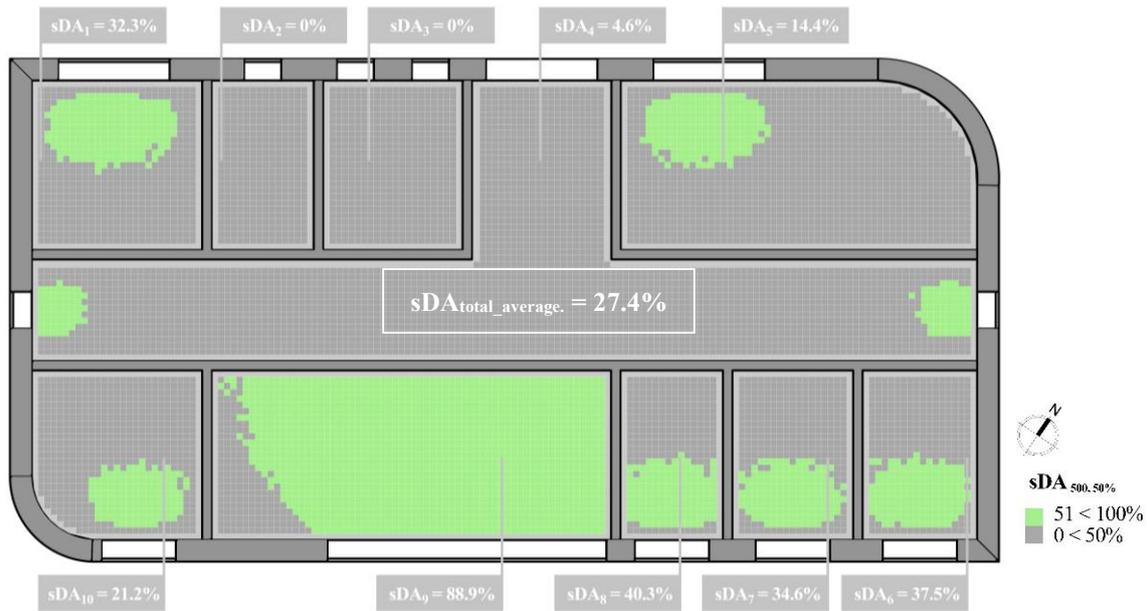


Figure 4.29 Spatial Daylight Autonomy (sDA) heatmap for the multi-zone model, no blinds

The values for the spatial daylight autonomy for the multi zone model with louver shades resulted as 0% due to the high coverage achieved with the 30 degrees inclination of the louver elements (*Figure 4.30*).

- *Annual Sunlight Exposure (ASE)*

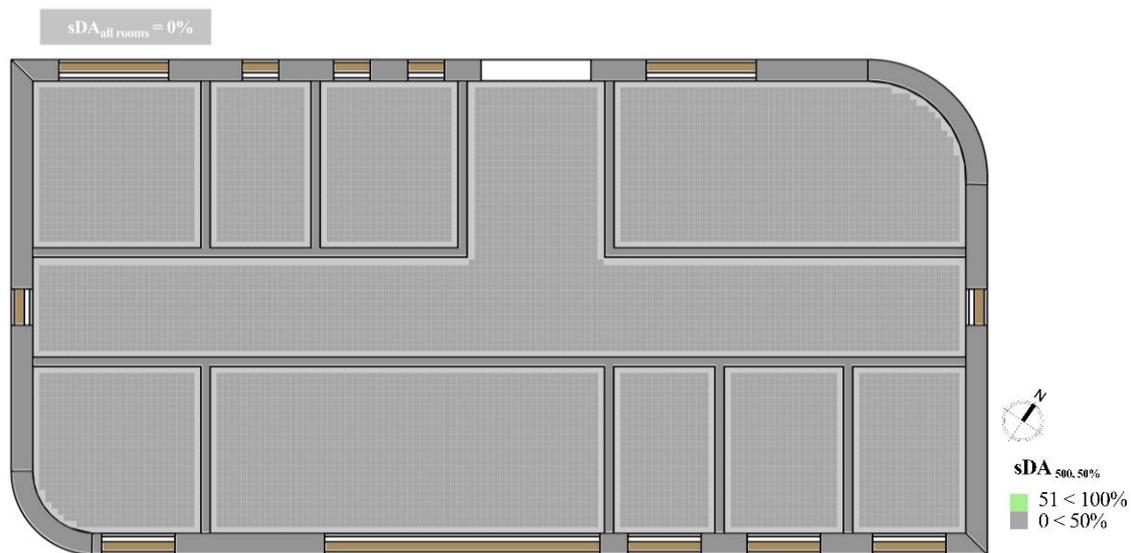


Figure 4.30 Spatial Daylight Autonomy (sDA) heatmap for the multi-zone model, louver shades

The percentage of space that receives 1000 lux for more than 250 hours per year was lastly determined using the sDA. The possibility of setting a threshold for direct sunlight with the help of the offered algorithm ultimately produced the desired results.

The values of Daylight Autonomy were then multiplied by the number of occupied hours, 5992 in total, in accordance with the specified schedule for small offices and the analysis period for a full year, after which they were divided by 100. The points of the assessing surface were then divided by all the values, 1652 in total that had met the threshold for more than 250 hours.

The heat map shown (*Figure 4.31*) depicts the 16.7% yellow-colored region where grid points received 1000 lux for more than 250 hours per year.

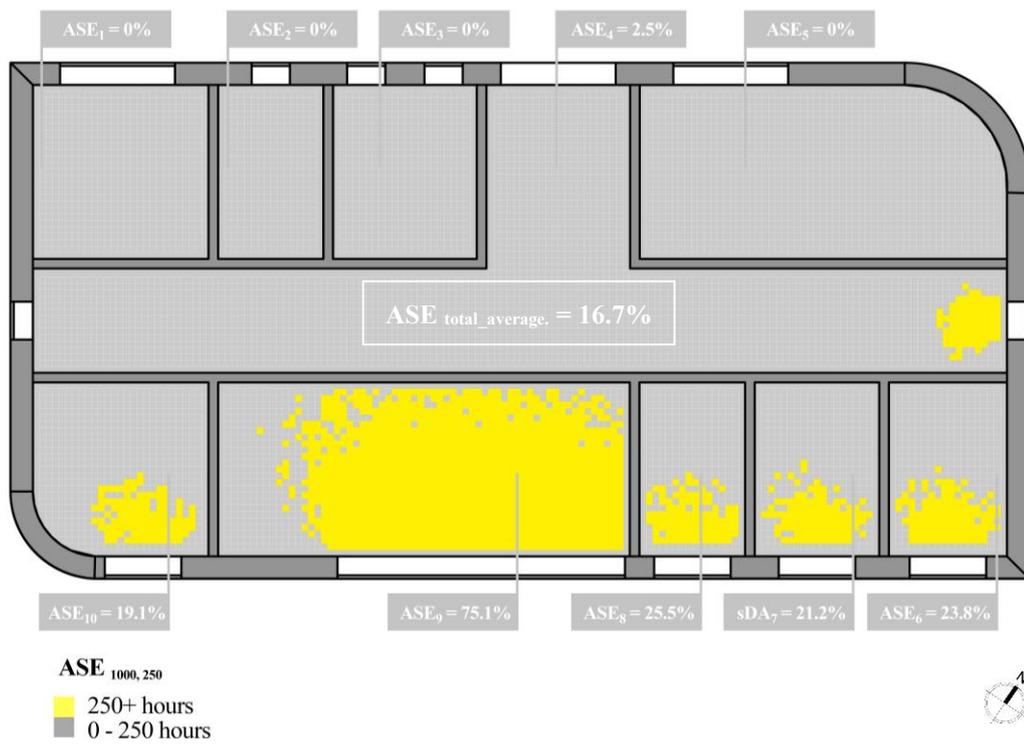


Figure 4.31 Annual Sunlight Exposure (ASE) heatmap for the multi-zone model, no blinds

Also the rooms which were oriented on the North West façade didn't receive 1000 lux for more than 250 hours per year. That's why on the heatmap the testing grid points are colored in grey.

Lastly the results were processed in finding out whether the LEED requirement was passed or no. It resulted in passing the LEED with an additional explanation for glare mitigation strategies. This means that with the shading devices this can be controlled. Adding the louver shades on the exterior part of the windows, helped controlling the annual solar exposure in producing disturbing glare inside the building and effecting the occupants' comfort in the internal area of the offices.

- Useful Daylight Illuminance (UDI) fell short (0-100 lux)

The Useful Daylight Illuminance (UDI) fell short implies that the average figure is the percentage of time when the illuminance dropped between 0 and 100 lux. The highlighted area is represented by the colorful heat maps as seen on *Figures 4.32* and *Figure 4.33*, with a variety of colors designating the various percentages of time. The average UDI (0-100 lux) in the multi zone model without blinds is 51% and 80.7% with louver blinds. The second scenario (louver blinds) clearly contains a bigger area with illumination of 0-100 lux due to the shading elements obstruction, which is how these results can be interpreted.

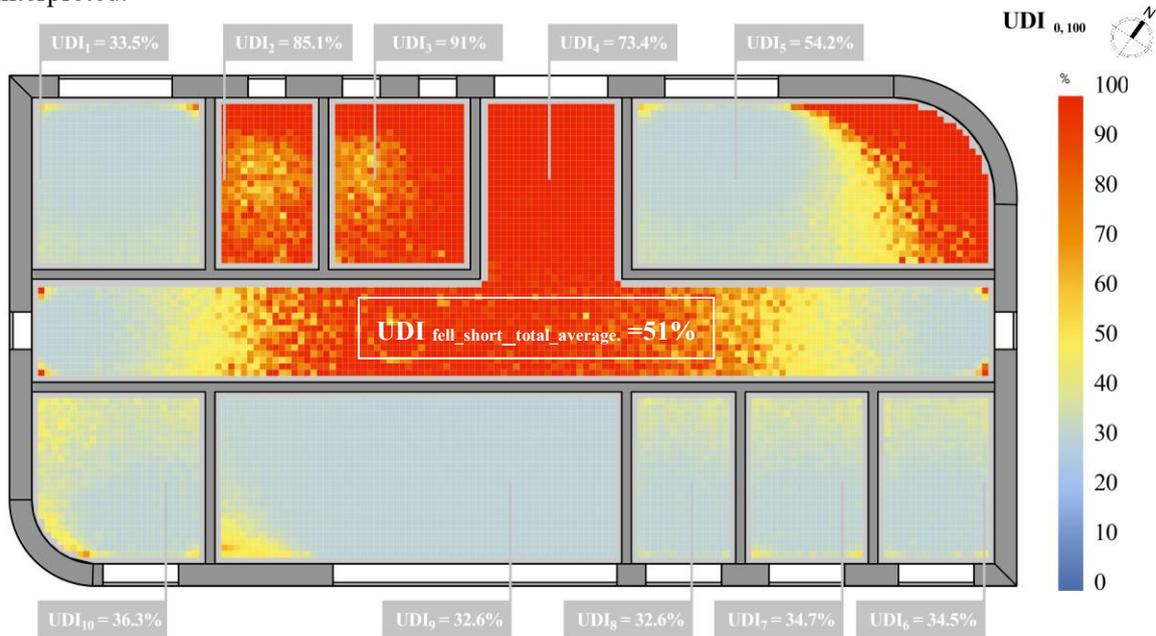


Figure 4.32 Useful Daylight Illuminance (UDI) 0-100 lux, heatmap for the multi-zone model, no blinds

The results are showing that again in the rooms facing the north the UDI fell short is having a large percentage of the average values. On the other hand the rooms facing the south-east and south-west are having a lower percentage of UDI that is between 0-100 lux.

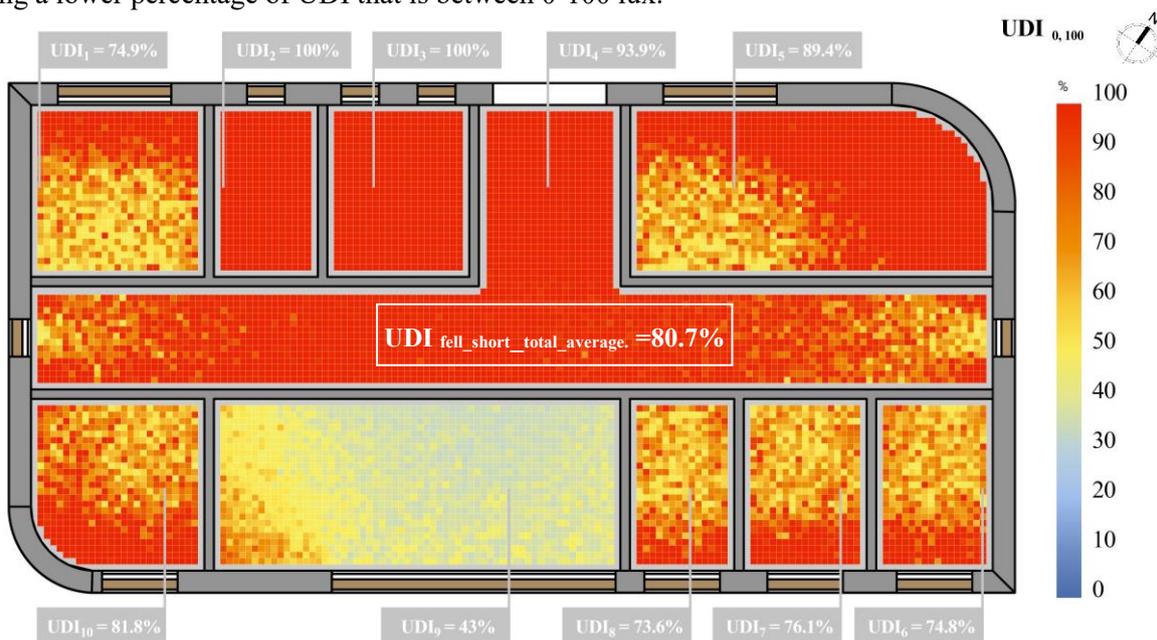


Figure 4.33 Useful Daylight Illuminance (UDI) 0-100 lux, heatmap for the multi-zone model, louver shades

- *Useful Daylight Illuminance (UDI) supplementary (100-500 lux)*

The supplementary type of Useful Daylight Illuminance (UDI), which is depicted in *Figures 4.34* and *Figure 4.35*, has lower and upper thresholds of 100–500 lux of illumination. The average result of the supplementary UDI in the first case without blinds is 21.1% of the examined area, while in the second case it is 16.1%.

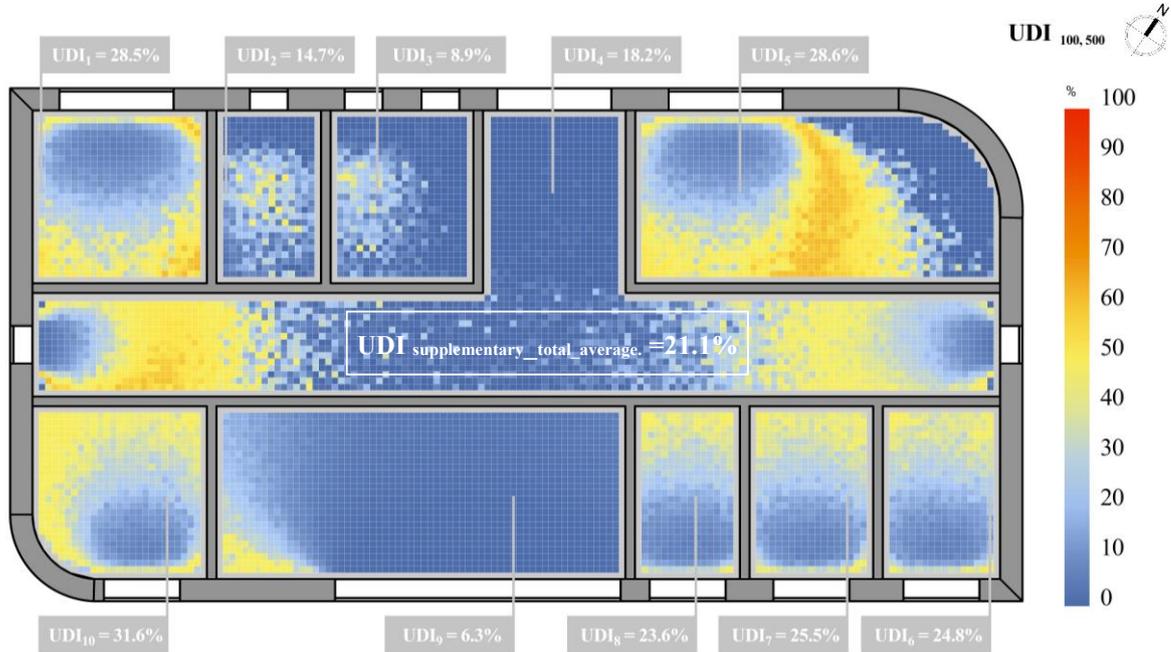


Figure 4.34 Useful Daylight Illuminance (UDI) 100-500 lux, heatmap for the multi-zone model, no blinds

Moreover it is noticeable that the place where there is no need for artificial lights is the conference room with an average value of 33.6%.

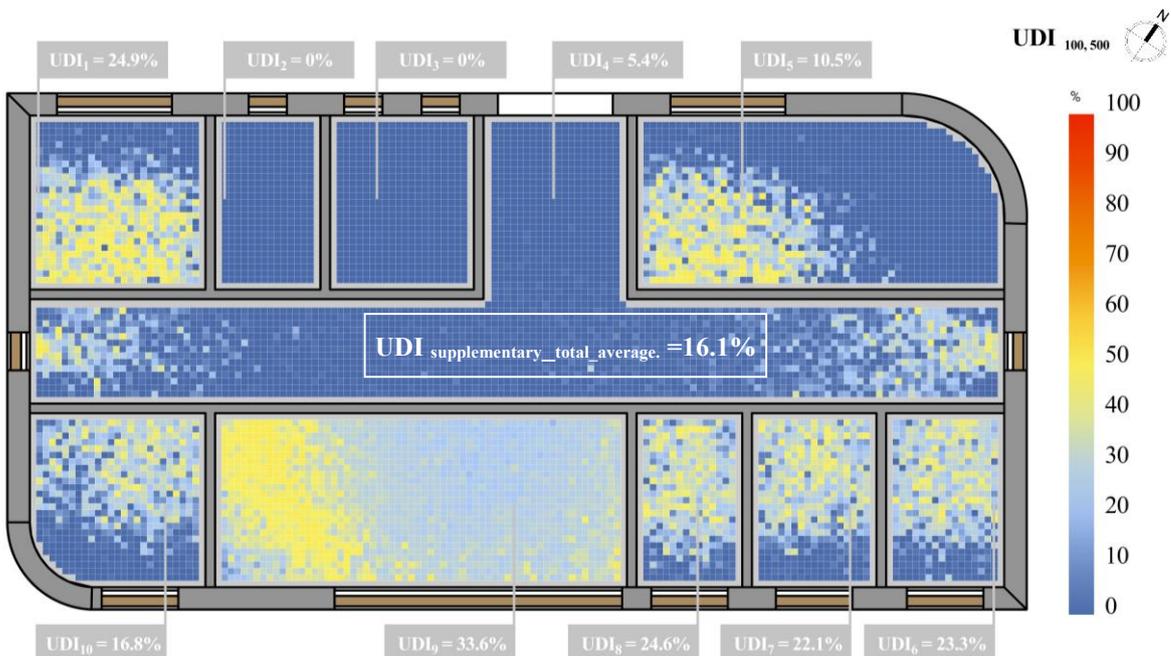


Figure 4.35 Useful Daylight Illuminance (UDI) 100-500 lux, heatmap for the multi-zone model, louver shades

- Useful Daylight Illuminance (UDI) autonomous (500-3000 lux)

The upper and lower thresholds for the third category of Useful Daylight Illuminance (UDI) autonomous were set at 500 and 3000 lux, respectively. The average values of 22.6% (Figure 4.36) and 2.9% (Figure 4.37) show how large an area is where illuminance values between 500 and 3000 lux are met.

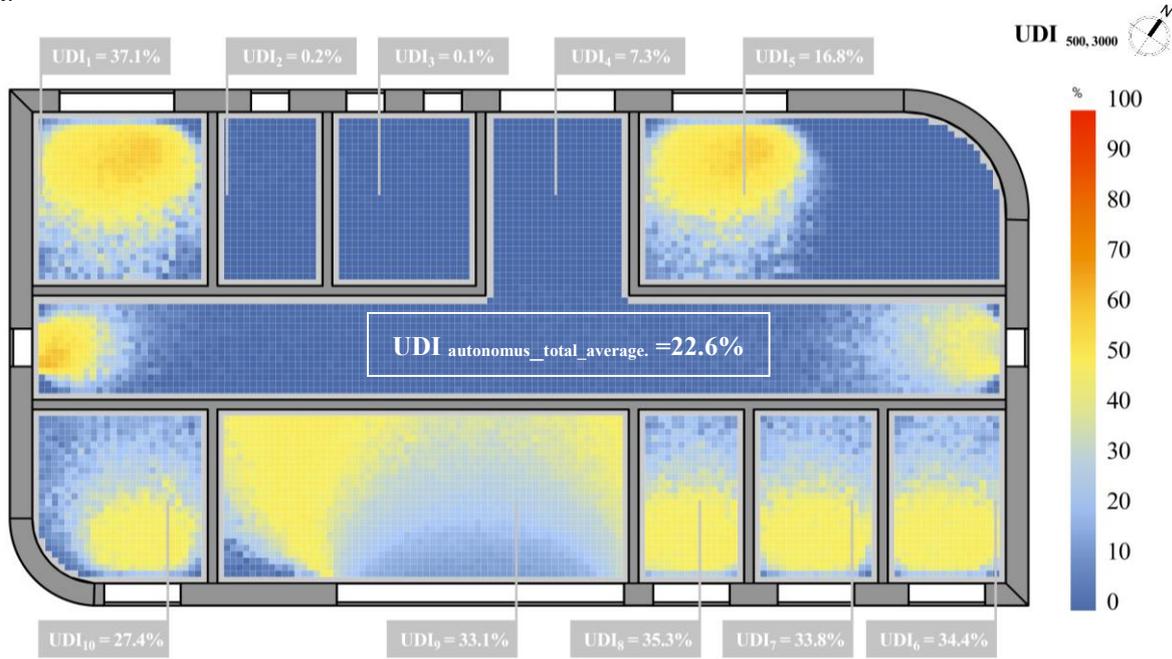


Figure 4.36 Useful Daylight Illuminance (UDI) 500-3000 lux, heatmap for the multi-zone model, no blinds

The conference room is having a much higher average value for the UDI of 500-3000 lux. It can be seen that all the rooms on the North East façade are not resulting not even close to the targeted value, so that means that when the louver shades are on there is a need of artificial light.

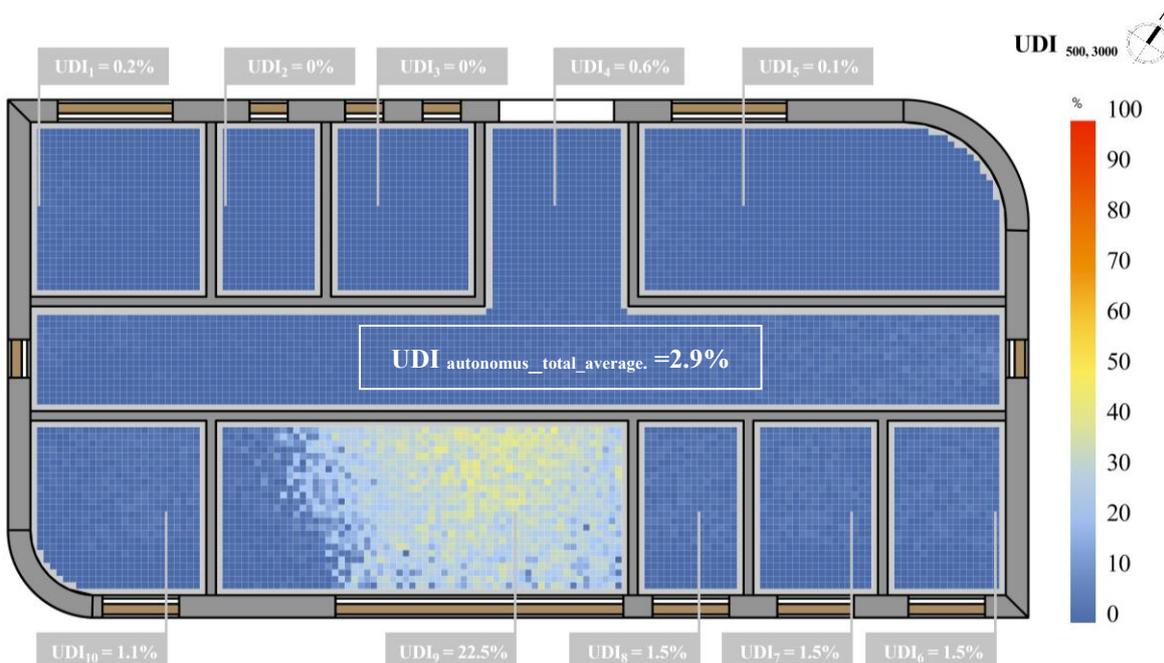


Figure 4.37 Useful Daylight Illuminance (UDI) 500-3000 lux, heatmap for the multi-zone model, louver shades

- Useful Daylight Illuminance (UDI) exceeded ( $>3000$  lux)

The results where the illuminance above 3000 lux were the last ones for the Useful Daylight Illuminance (UDI) computations. Figure 4.38 displays the average value of 5.4%, and Figure 4.39 displays 0.2% of the region where the illumination is greater than 3000 lux for the louver blinds scenario.

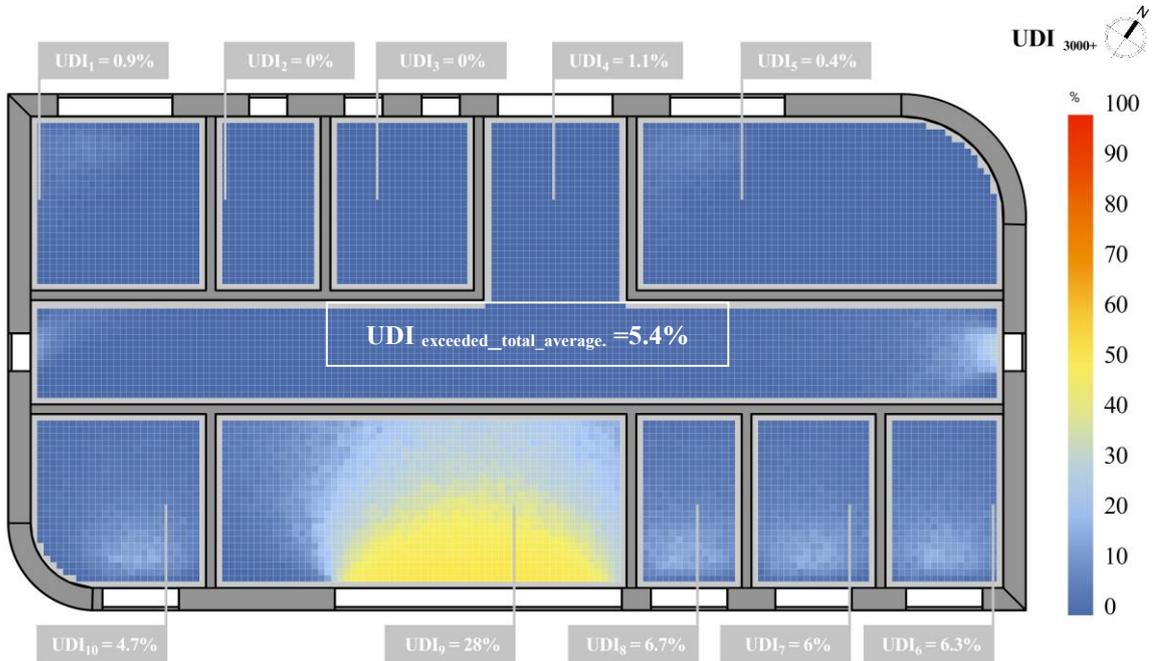


Figure 4.38 Useful Daylight Illuminance (UDI)  $>3000$  lux, heatmap for the multi-zone model, no blinds

The values for obtained UDI over 3000 lux are really low in both situations, with and without blinds.

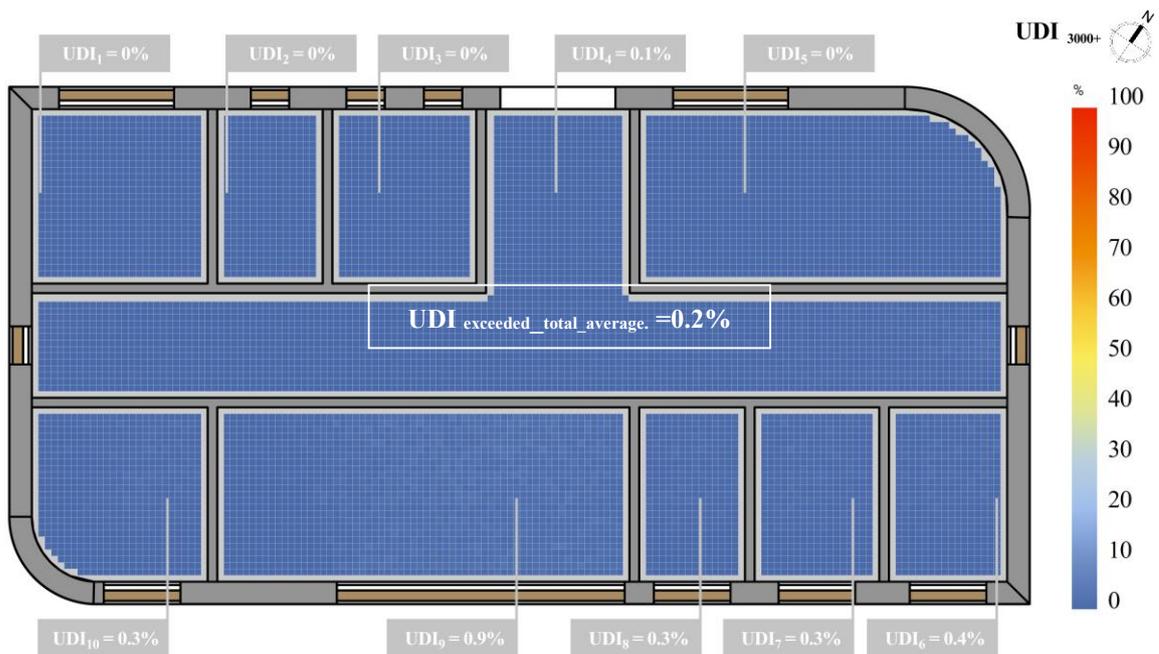


Figure 4.39 Useful Daylight Illuminance (UDI)  $>3000$  lux, heatmap for the multi-zone model, louver shades

## 4.2.2. GLARE ANALYSIS RESULTS

The multi-zone model without blinds is first simulated, and then the model with the louver blinds is simulated. The results will be displayed in the pages that follow, with heat maps and average values for each sensor grid point in all rooms, together with the overall average value. In a second glare analysis, the Daylight Glare Probability (DGP) for two rooms with a point in the middle of the room facing the window opening was determined. The results were represented with values and images.

- *Glare Autonomy (GA)*

Glare Autonomy (GA) simulations were carried out with a threshold of 0.4 DGP above which conditions are regarded to cause glare, and the results are presented as the percentage of occupied hours that are free of glare. The outcomes are displayed in *Figures 4.40* for the multi-zone model without blinds and *Figure 4.41* for the model with a louver blinds.

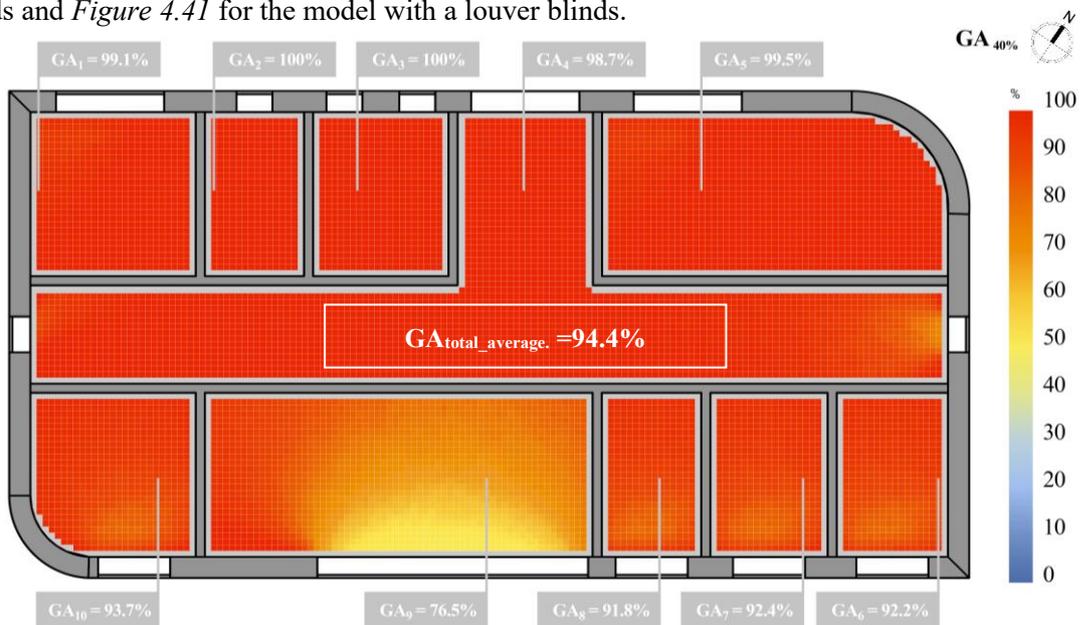


Figure 4.40 Glare Autonomy (GA) heatmap for the multi-zone model, no blinds

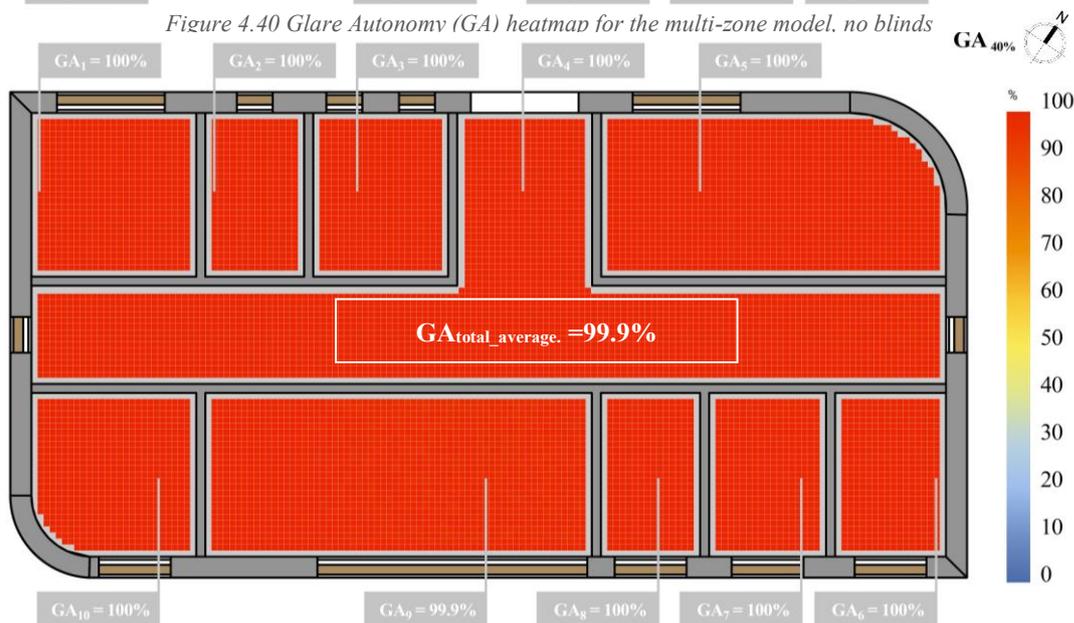


Figure 4.41 Glare Autonomy (GA) heatmap for the multi-zone model, louver shades

- *Daylight Glare Probability (DGP)*

The DGP results were simulated from two rooms in the multi zone model. One being Office 5 and the other one is the Conference room 9. A point for the view was in the middle of the room and the view towards the window opening. All results can be observed in the following figures. The evaluated dates are June 20<sup>th</sup> and December 21<sup>st</sup>. Three times a day, between the hours of 9am, 12pm and 3pm, they are represented for both solstice and equinox.

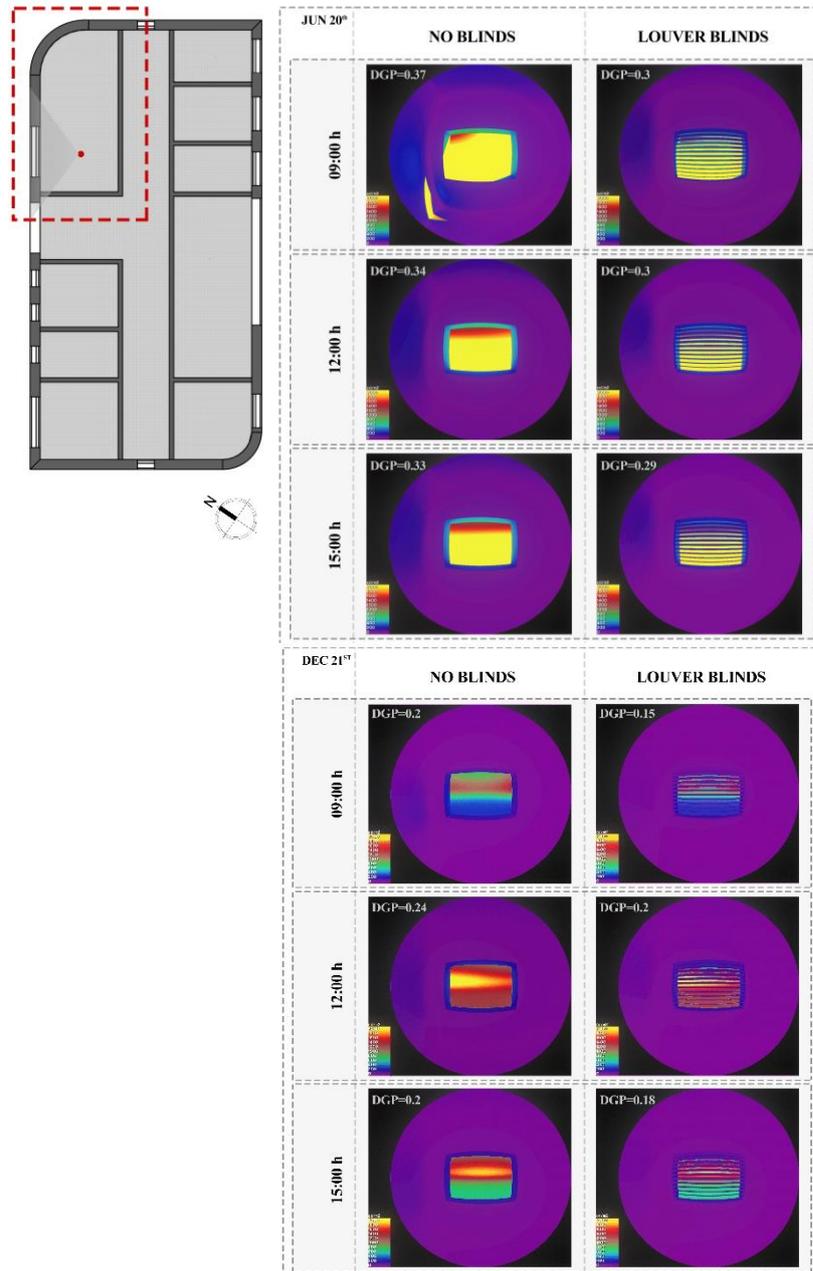


Figure 4.42 DGP results for multi-zone model, Office 5

Hereby in the *Figure 4.42* are shown the results from the calculation for the DGP in the selected zone, Office 5. The obtained results could be observed and they represent values and images which are showing how the glare is almost in each room imperceptible or even not validated. It is noticeable that the blinds decreased the DGP in the room. But not to an extreme level, due to the small openings that may occur between the louver blinds' elements and also the used materials.

The obtained results in the second room that was tested, are shown in *Figure 4.43*. In this case due to the size of the glazing of the curtain wall, which is larger than the other window openings, the DGP values were higher. There was always some type of daylight glare probability, which could be perceptible or even disturbing on a high level. For example during December, on the selected date, there was multiple chances of disturbing glare. Also for summer there was an excessive amount of direct light coming into the room and that is again due to the size of the opening compared to the floor area 90% (window to floor ratio is greater than 1/6).

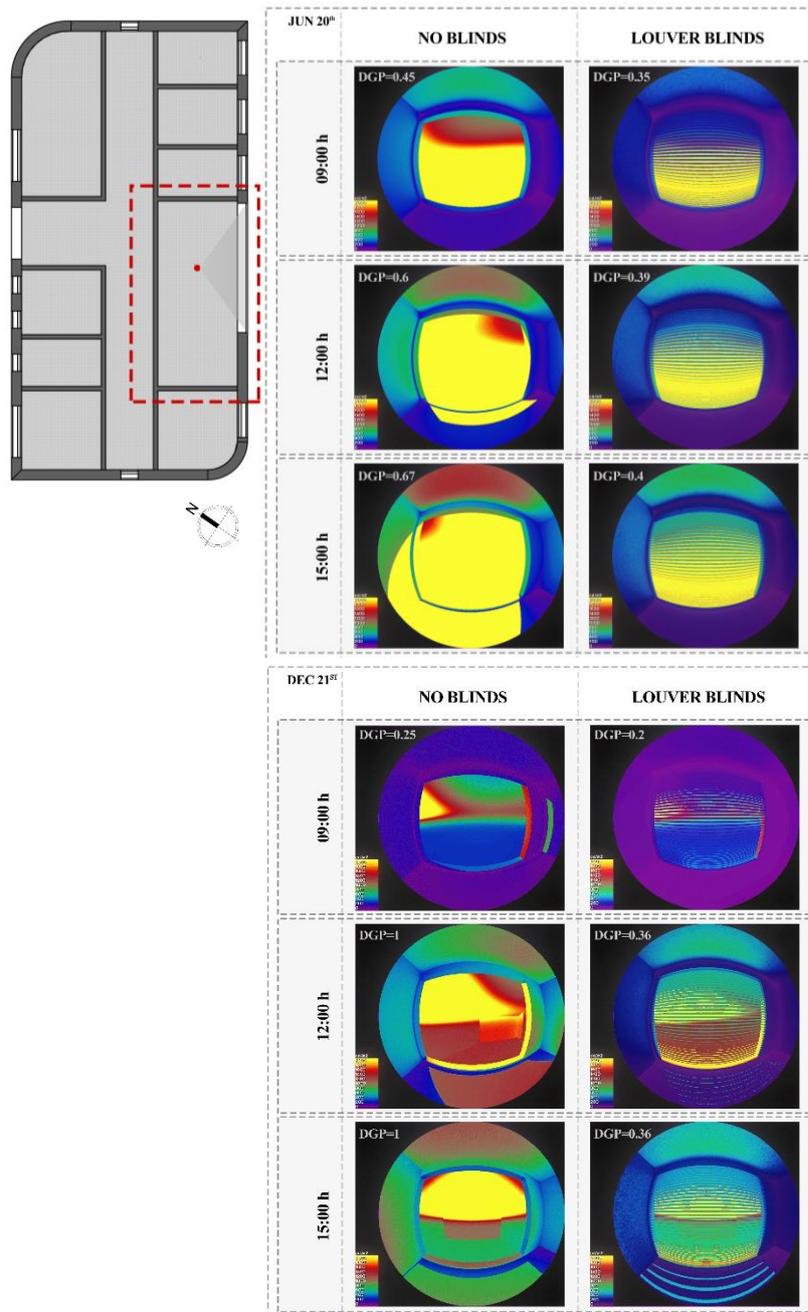
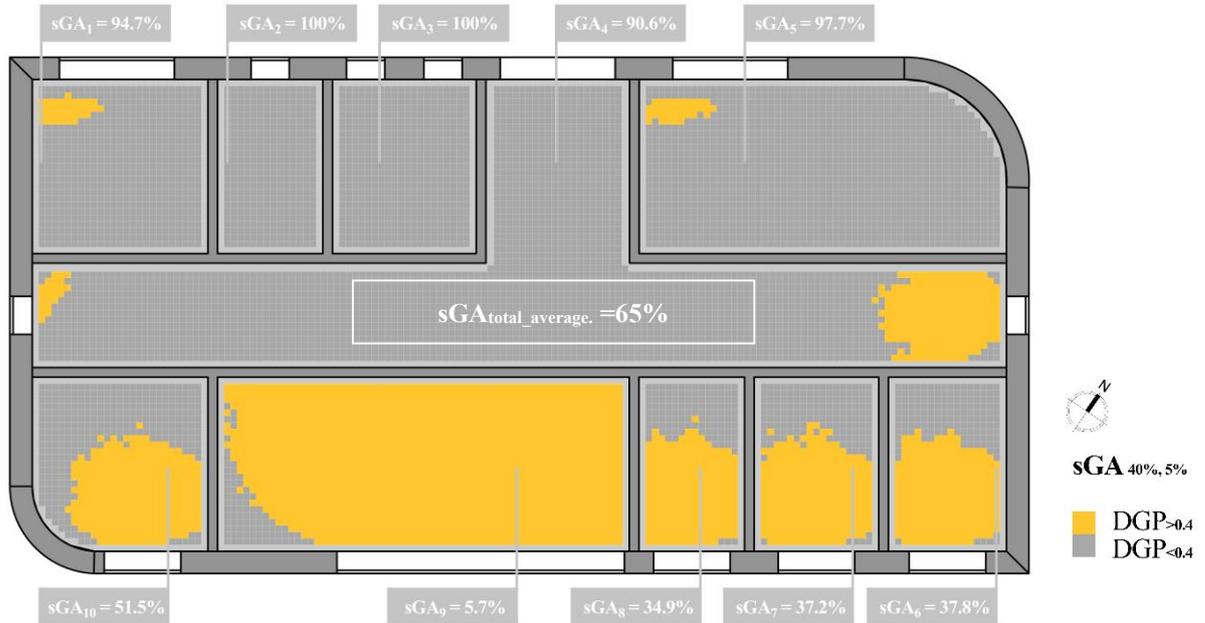


Figure 4.43 DGP results for multi-zone model, Conference Room 9

Having the louver shades with a 30° inclination enabled a situation where the DGP could be lowered and it could reach a value that is belonging to the group of imperceptible or perceptible glares. The parameters that were used for all DGP analysis are again with low quality due to the time consuming of the simulations.

- *Spatial Glare Autonomy (sGA)*

Finally, the outputs from the Spatial Glare Autonomy (sGA) were compiled and displayed as a heatmap. *Figure 4.44* displays the multi-zone model without any shading components and yields a result of 65%, indicating that a majority of the tested grid points fall below the DGP threshold of 0.4. It signifies that the region is free from glare for at least a portion of the targeted 5% of the time.



*Figure 4.44 Spatial Glare Autonomy (sGA) heatmap for the multi-zone model, no blinds*

The multi-zone model with louver blinds is shown by the second heatmap on *Figure 4.45*. It is clear that the percentage is greater and that the louver blinds are blocking the direct sun rays that would otherwise cause glare in excess of 0.4 from entering the room. If the spatial glare autonomy is 100%, then the entire room will be free of glare for at least 5% of the time that it is occupied.



*Figure 4.45 Spatial Glare Autonomy (sGA) heatmap for the multi-zone model, louver shades*

### 4.2.3. COMMENTS ON THE RESULTS

In the following pages comments on the results are presented, acknowledging the previously mentioned outputs from the simulations performed for all KPIs. It is important to take a look at *Table 4.2* where all the mean values for each room and each KPI is introduced in percentage, for both case scenarios, without shades and with louver blinds.

Room type	Shading element type	Key Performance Indicators (KPIs) (Mean value) [%]										
		DF	DA	cDA	sDA	ASE	UDI <sub>fell-short</sub> (0-100 lux)	UDI <sub>supplementary</sub> (100-500 lux)	UDI <sub>autonomous</sub> (500-3000 lux)	UDI <sub>exceeded</sub> (>3000 lux)	GA	sGA
Office 1	No shading element	3.9	38.4	57.2	32.3	0	33.5	28.5	37.1	0.9	99.1	94.7
	Louver blinds	/	0.1	11.7	0	/	74.9	24.9	0.2	0	100	100
Utility 2	No shading element	0.4	0.2	9.6	0	0	85.1	14.7	0.2	0	100	100
	Louver blinds	/	0	1.3	0	/	100	0	0	0	100	100
WC 3	No shading element	0.3	0.1	7.3	0	0	91.0	8.9	0.1	0	100	100
	Louver blinds	/	0	1	0	/	100	0	0	0	100	100
Entrance 4	No shading element	0.9	8.3	18.5	4.6	2.5	73.4	18.2	7.3	1.1	98.7	90.6
	Louver blinds	/	0.7	4.6	0	/	93.9	5.4	0.6	0.1	100	100
Office 5	No shading element	2.1	17.4	34.9	14.4	0	54.2	28.6	16.8	0.4	99.5	97.7
	Louver blinds	/	0.1	7	0	/	89.4	10.5	0.1	0	100	100
Office 6	No shading element	3.3	40.9	55.1	37.5	23.8	34.5	24.8	34.4	6.3	92.2	37.8
	Louver blinds	/	1.7	11.6	0	/	74.8	23.3	1.5	0.4	100	100
Office 7	No shading element	3.2	40.2	54.8	34.6	21.2	34.7	25.5	33.8	6.0	92.4	37.2
	Louver blinds	/	1.7	11.3	0	/	76.1	22.1	1.5	0.3	100	100
Office 8	No shading element	3.5	42.1	55.6	40.3	25.5	34.4	23.6	35.3	6.7	91.8	34.9
	Louver blinds	/	1.8	12.4	0	/	73.6	24.6	1.5	0.3	100	100
Conference Room 9	No shading element	10.8	61	64.6	88.9	75.1	32.6	6.3	33.1	28.0	76.5	5.7
	Louver blinds	/	23.5	41	0	/	43.0	33.6	22.5	0.9	99.9	100
Office 10	No shading element	2.6	32.3	49.9	21.2	19.1	36.3	31.6	27.4	4.7	93.7	51.5
	Louver blinds	/	1.4	9.5	0	/	81.8	16.8	1.1	0.3	100	100

*Table 4.2 KPIs Results for the Multi-zone model, both without shading element and louver blinds*

To begin with the whole building performance must to be taken into consideration. According to this, the average values for all KPIs calculated for whole building in total are presented in the following pages in *Table 4.3*.

As in the previous case study, the analysis started with the daylight factor. It can be seen that the larger number of the rooms have daylight factor that is around 2-3%, which reaches the threshold for offices. Rooms with the lowest daylight factor are the Utility and the Restroom resulting with almost 1% of DF. The overall average value received as a result, was 3.1% which probably got that high just because of the higher daylight factor in the Conference Room. This of course passes the target value, but it can be improved with larger window openings or changing the glass properties, which could be an important factor in the future analysis. But this study was done in order to see if the results could be retrieved for the performed simulations and divided according to the room types, also how they affect the overall building performance.

About the other indicators, it is sure that all the rooms gained some kind of daylight, except the ones that are on the NE façade of the building and more precisely the utility and the restroom. Their windows have a high parapet, distance from floor of 160cm. Maybe the case was not that those rooms didn't resulted with higher values because of their size, location, etc, but the sensor grid for the simulations was situated on 90cm distance from floor. So there is a high chance of daylight passing through the window, but could be detected only if the sensor grids are set higher, closer to the windows.

Getting the results for daylight autonomy for both case scenarios, it can be seen that for the first scenario where there are no louver shades, the Offices 1,6,7,8 and 10 had around 30-40% of the occupied hours that had the illuminance equal or above 500 lux, which resulted in not passing the target value. Only the Conference Room 10, exceeded the 50%, with an average value for all points on the grid of 61%. The area where there is the least amount of illuminance above 500 lux is the Corridor 4, where most of the space is dark. The overall average value for daylight autonomy resulted in 28.1%.

On the other hand the daylight autonomy for the case scenario with louver shades resulted with almost 0% of the occupied hours receiving illuminance above 500 lux. Again this is due to the size and orientation of the windows, but mainly the obstruction made from the inclination of the louver shades. For the multi-zone model with louver blinds the whole building performance average value for DA is 3.1%.

When comparing the results from cDA inputted out from the analysis on the multi-zone model with and without louver blinds, the requirement is met only in the first case scenario, where there are no louver blinds. Of course the differences in the results between the model without blinds and with louver shades can be seen with comparing the two average values for the overall building performance. First case scenario resulted with 40.8% and the second 11.1%.

The sDA for an annual calculation resulted as 0% for the model containing louver blinds. The average value of 27.4% for the overall building performance of the multi-zone model without shading device, represents a low number which shows that the highlighted area of that amount, 50% of the operating hours per year is receiving minimum of 500 lux of daylight illumination.

Same for the annual sunlight exposure, some of the rooms didn't get any direct sun rays, due to the fact that they are facing NE direction. But that is not a problem because the measurements of the windows and the properties of the glass surfaces can be always changed. The outcome of the room where there is a curtain wall, shows that 75.1% of the area received 1000 lux for more than 250 hours per year. This signifies that this room if there aren't any shading elements could have an increased value of DGP and it needs a shading device.

Shading element type	Key Performance Indicators (KPIs) (Mean value) [%]										
	DF	DA	cDA	sDA	ASE	UDI <sub>fell-short</sub> (0-100 lux)	UDI <sub>supplementary</sub> (100-500 lux)	UDI <sub>autonomous</sub> (500-3000 lux)	UDI <sub>exceeded</sub> (>3000 lux)	GA	sGA
No shading element	1.4	12.4	26.1	7.8	9	60.1	27.4	9.8	2.7	94.4	80.3
Dynamic shading element	/	5	13.5	0.6	/	81.6	13.4	4.5	0.5	99.5	98.1

Table 4.3 KPIs Results for the Multi-zone model, both without shading element and louver blinds, overall building

Finalizing the daylight analysis, UDI gave different outputs for the different categorized thresholds. Figure 4.46 shows a chart where all UDI values for each room are stacked and they are compared between the model without and with the louver blinds.

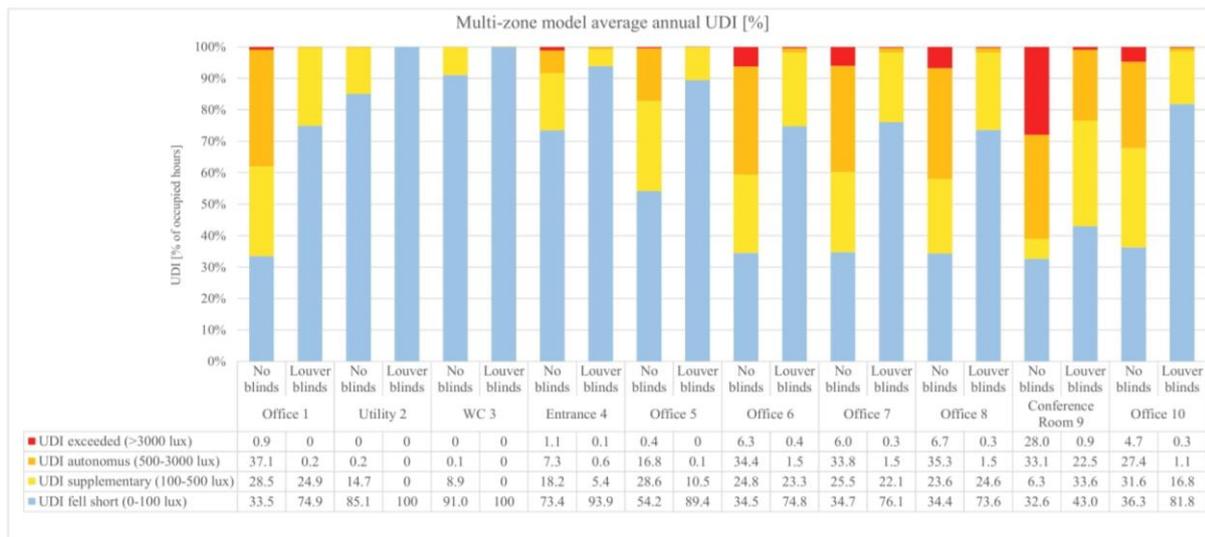


Figure 4.46 Multi-zone model average annual UDI

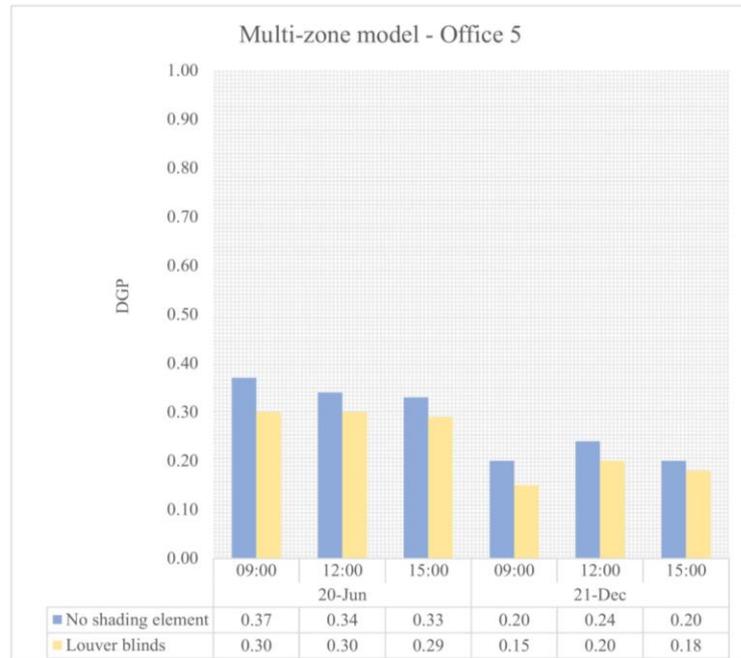
Starting with the office 1 which has a window which is on the NE façade, it is important to point out that it has over 60% of time with UDI<sub>100-3000lux</sub>. Continuing with the other rooms which have their windows on the NE façade, it can be seen that the value for UDI<sub>0-100lux</sub> is bigger, also because they have smaller windows. Next the rooms on the SW façade show higher values for UDI<sub>100-3000lux</sub>, but having lower values for UDI<sub>0-100lux</sub>. For example the room with lowest percentage of occupied hours that have illuminance up to 100 lux is the Conference Room 9 and the one with highest average value is the Corridor. This means that the corridor needs another source of light, which could result in application of artificial lights or changing the design of the model.

Furthermore from looking at the chart, the building overall has a high percentage of UDI fell-short, but that is due to the average values from the smaller rooms, such as utility and restroom, where the windows are small. Looking at the results for UDI exceeded, it only happens in the rooms on the SW façade, mainly in the Conference room, when there are no blinds.

Lastly from the chart can be concluded that the louver shades made quite large difference for the overall building performance, especially the percentage of areas that got useful daylight illuminance which is over 500 lux.

Glare Autonomy simulations were simulated in all rooms, without blinds and with louver blinds. The difference can be seen mostly in the conference room. The curtain wall gives the opportunity of DGP to be over 0.4 and an occurrence of disturbing glare. On the other hand the model with louver blinds completely blocks the glare and the resulted values are lower. These results show that the model needs to be improved in order to achieve the desired outcome of the overall building performance. Lastly spatial glare autonomy resulted in quite high values and from the results it can be seen that the glare is mostly disturbing or intolerable in the conference room and not valid or imperceptible in the NE oriented rooms.

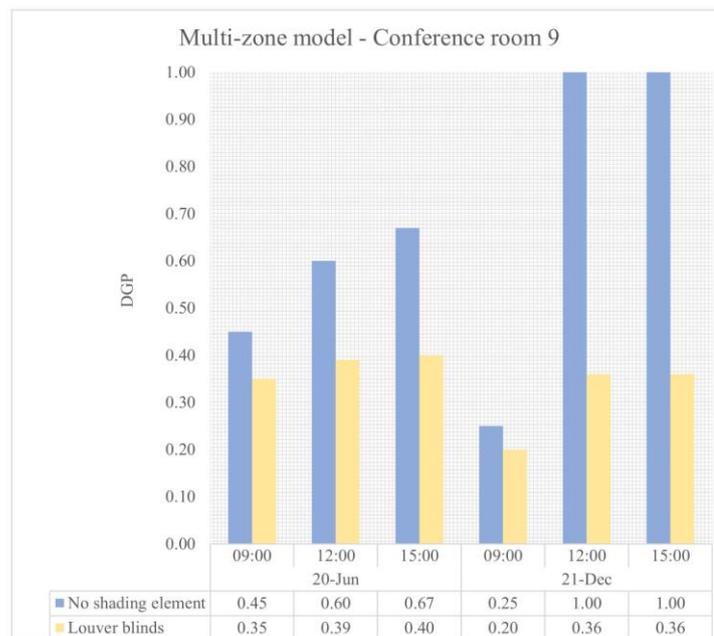
However the last tests that were made using point in time DGP analysis, were evaluated in Office 5 and Conference room 9. *Figure 4.47* is representing a chart where the results for both case scenarios for office 5 are shown.



*Figure 4.47 Multi-zone model DGP for Office 5, June 20th and December 21st, at 9pm, 12pm and 3pm*

The differences were noticeable, during both seasonal periods, but still they are not that different and most importantly the glare is under control.

On *Figure 4.48* the results from Conference room 9 are presented in a chart. Expectedly the DGP had such high values in the conference room, during winter. Due to the orientation of the windows and also the view that was directly hit by the sun rays. Other than that it can be concluded that the louver blinds obstructed most of the glare that could be present in both of the rooms.



*Figure 4.48 Multi-zone model DGP for Conference room 9, June 20th and December 21st, at 9pm, 12pm and 3pm*

## **5. DISCUSSION**

At the beginning the following chapter focuses on the strengths and limitations of the performed research. It is acknowledging the possibilities that are obtained while using the BIM programs and the BPSTs mentioned and utilized during the examination of the workflows. Also it addresses the lack of abilities, which are part of the software programs. Next it elaborates how the gathered information from both literature and practice could be used in an implication of another user for another experiment. The opportunities of future analysis following this study are also mentioned in the following pages. Lastly it was worth mentioning the backwards interoperability, as a consequence of using Rhino Inside Revit as a plug-in for the interoperability between Revit and Grasshopper. Due to the fact that the plug-in can perform smooth interoperability between the above mentioned software programs, the backwards interoperability is a workflow that could be examined in the future.

### **5.1. STRENGTHS AND LIMITATIONS OF THE RESEARCH**

At the beginning of the research an enormous attention was paid to the already existing information about the interoperability between BIM software programs and BPSTs. The collected data from articles, papers, books, sites, was carefully observed and analyzed in order to gain as much knowledge as possible before the practical use of the tools. During the collective process a lot advantages and disadvantages came across, which were mentioned either by experienced users of the software programs or people who are professional architects, engineers, etc. and on the other hand enthusiastic users, which are new to the building performance simulations.

When talking about all the information gathered at the beginning of the research, it can be said that it was really helpful to find multiple tools that were available for assessing the performance of a building envelope from a visual comfort point of view. Addressing the possibilities and opportunities of various daylight analysis simulation tools, provided an open path to explore the ones that are connected with parametric modeling. The assembled knowledge answered the first research question, which is strictly connected with the visual comfort analysis done on account of BPST's. To complete the opinion about their importance of their use for analyzing the daylight and glare in a building, the research had to proceed with two case studies, one being a single-zone model and the other one a multi-zone model.

In order to elaborate the strengths and limitations of a software a workflow was constructed, according to the complexity of the models. At first it was noticed that constructing a 3D model in a BIM environment, such as Revit, is quite easy. Mostly because, when providing a 3D model for this type of analysis, it is important to build it correctly, with not many needed configurations in the future experimentation process. Revit gives an opportunity to construct a model of a different type and supply the properties of the model with all needed information for further evaluation. But to correctly prepare a model which would be later exported to one of the simulation tools, a step by step guide is easier to be followed. Within the research this workflow was elaborated and one of the strengths found within

Revit modeling, was the accessibility to a range of libraries consisting model geometries, types of materials and also a chance to modify them in the preferable way. Moreover there are multiple plug-ins that could work within Revit's interface and offer a choice of analyzing the model within the program. Due to the evaluation on data transfer between Revit and a VPL program software, a connection for smooth interoperability was used. Rhino Inside Revit was introduced as a plug-in for Revit, that gave a possibility to directly open the Rhino UI from Revit and then get access to Grasshopper. Regarding the exportation of the 3D model into Grasshopper, the total extraction process of the model elements was performed with the help of Revit components from the existing toolbar within Grasshopper. Another advantage of using Rhino Inside Revit was the possibility to extract all elements that are existent in Revit, their material properties and many more configurations of the model. Also, if any change was made on the model in Revit, the information was immediately transferred to Grasshopper.

Concerning the fact that Grasshopper is a tool in Rhino, it supplies numerous plug-ins that could be installed within its interface and later used for desired purposes. It exists as a very capable and compelling tool, which offers a wide selection of activities to be performed from a basic to high level of expertise. The model elements' extraction operation for the single-zone model was quickly finished, but the only thing that was a problem connected to the geometry was the extraction of curved walls, while performing the second type of model extraction (exportation of rooms). This could be a possible limit of the program, which results with a time consuming problem. Another limitation of the interoperability performed with Rhino Inside Revit is the data transfer of specific material properties to Grasshopper. They could be extracted, but it goes to a point where the unit type stays different and the only way to use the information about the material is to manually convert the units within Grasshopper. On the other hand all steps together did not result as inefficient workflow, it resulted with quite the opposite. It is worth to mention that a limitation of the Grasshopper components when building a material are the limited inputs that could be plugged in in order to perform a simulation, which could result with slightly inaccurate results. In spite of that the second research question could be proclaimed as answered, because the used steps for achieving a smooth interoperability between Revit and Grasshopper, using Rhino Inside Revit, was achieved on an exceptional extent.

As mentioned previously few of the provided plug-ins for Grasshopper were used in order to perform the daylight and glare simulations, among which Honeybee components as simulation performers and Ladybug components as visualization tools were used. A list with desired KPIs was constructed at the beginning of the implementation of Honeybee components for the visual comfort analysis. All of them could be examined with the use of the provided components. The only issue when performing a daylight and glare simulation is if the use of surfaces is wrong. This means that the internal surfaces of the rooms needed to be correctly inputted in order to have the most accurate results. When performing the simulations for the single-zone model the operation didn't took a lot of time, even when using a high level of radiance parameters. The only problem that occurred was while performing the DGP simulation for point-in-time. With a detailed level of the outcome represented as an image with false color, the simulation could take up to 2-3 hours. Due to the complexity of the multi-zone model the simulations took more minutes to be performed, but no errors occurred. The process of retrieving

the results, was relieved with the use of Ladybug components for visualization. Even though in the multi-zone model there were 10 rooms not being evaluated separately, the results could be processed individually, using components which split up the data according to the room itself. Lastly the observation of the results from the daylight and glare analysis simulations, gave an opportunity to develop and upgrade the model in Revit or perform simulations under other conditions of the building design.

## 5.2. POSSIBILITY OF BACKWARDS INTEROPERABILITY

During the whole study, it was understood that Rhino Inside Revit gives the best opportunity of transferring data back and forth from Revit to Rhino and vice versa. Within the framework of the research this realization was gained while exploring the Grasshopper interface and its commands supplied by Revit. On account of geometry modifications, this ability is available with few components. They enable the user to perform some changes within Grasshopper, such as modifying the size, location and shape of elements, or even adding a new geometry on the existing one and then observe all changes on the Revit UI. About the construction of a new material, consisting specified properties, there was no exploration regarding that action. Lastly, an important thing connected to this part is that after performing the simulations and gathering the results, the only thing that could positively influence the design of the building is to optimize the data and later implement the desired changes to the model.

As a result of this finding a small experiment was performed, in order to see what could possibly happen if a parameter is changed from Revit UI. The experiment included only the single-zone model, testing one KPI using the same algorithm for simulating the Daylight Factor in order to compare if there is a difference in the results, if the interoperability works, how long it will take to transfer the information on the existing model in Rhino and perform the simulation on Grasshopper. Since the simulations are for visual comfort the change that was made was according to the window measurements, more precisely the size of the glazing surface. The window size changes were made within Revit UI and the new ones are: 220cm height and 110cm width, located on the SW façade (*Figure 5.1.*).

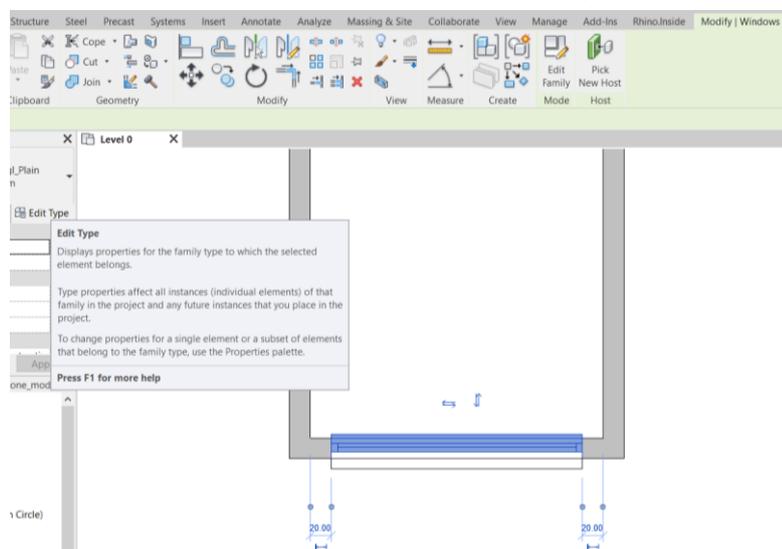
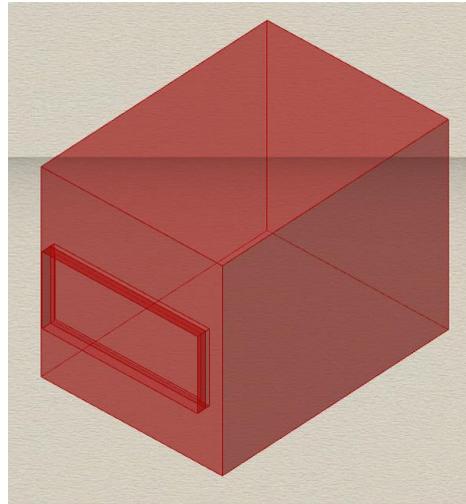


Figure 5.1 Screenshot from Revit UI, window measurement changes

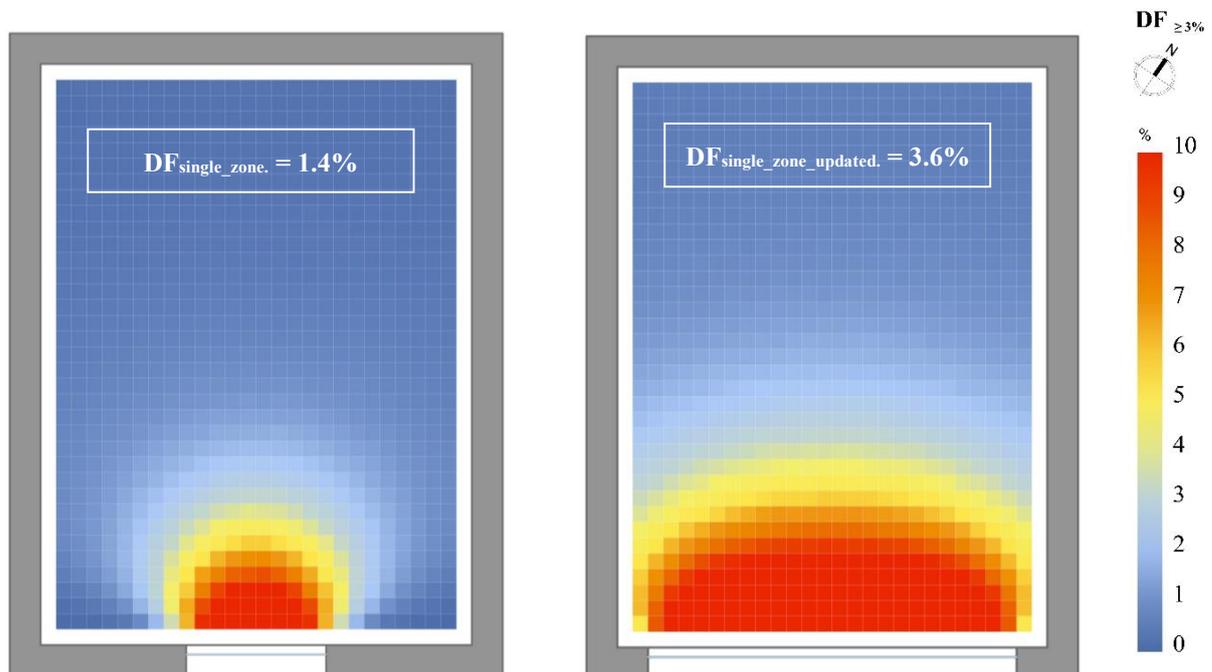
After the changes made in Revit, the Rhino UI can be opened immediately. Within few seconds the data is directly transmitted to Grasshopper and the model can be seen on the Rhino UI with the already made modifications (*Figure 5.2*).



*Figure 5.2 Screenshot from Rhino UI, 3D model with the window measurement changes*

There is no need of any algorithm modifications within the Grasshopper UI. Honeybee accepts the model as closed boundary and the simulation is ready to be performed.

Finally a heatmap with the new results for daylight factor was produced. This could be compared to the previous analysis and results on the single-zone model with the old window measurements (*Figure 5.3*).



*Figure 5.3 Daylight Factor (DF) heatmap for the single-zone model compared with the updated version*

The results show how the size of the window can change the outputted values from the simulation. As expected the new result has a higher value for the DF, 3.6% which is more than twice from the previous result and now the requirements for DF in an office are met.

Since the backwards interoperability is left for a future analysis, the idea of this experiment was to see if there are design changes into the model, how they can be translated on to the model for simulations. Also to see that after getting the results from the first simulations, the designer can change the design of the building or make some geometry modifications in order to run the simulations again and compare results with the variations of the design. This is really important as conclusion for the interoperability process using Rhino Inside Revit. The design changes can be made during the whole design process and it is easy to make modifications only on the model in Revit and then perform the simulations in GH. Only problem could be if there is a more complex model, but if all Revit elements previously were extracted correctly in GH, the changes won't bug the program and won't be any kind of geometry modification made within GH algorithm.

### **5.3. PRACTICE IMPLICATIONS AND FUTURE RESEARCH**

Since the beginning of the research, the lack of information about the topic of interoperability between BIM and BPSTs, was playing a big role into the development of this study. Not a lot of architects and engineers are interested in creating a co-simulation environment with a workflow for assessing their building design.

Concerning the fact that the evolution of technology is progressing rapidly in the last 20 years, the use of simulation engines will be essential for any type of building envelope evaluation. Hence this matter, this research focused on combining all steps that could be later used by any person who wants to perform daylight and glare analysis. With the variety of graphical representation of the parametric workflow, from building information models to building performance simulation, the information can be implicated practically from users.

Moreover it enables the opportunity to further explore this field and upgrade the existing step by step guide applied on more complex situations. For example the use of dynamic motion of building elements could be brought to another level. Additionally more compounded high performance buildings could be evaluated. Even other limits could be tested for both programs (Revit and Rhino (Grasshopper)). Also this research could help to identify all mistakes and data loss from the beginning in order not to receive any errors that could happen trough out the use of Rhino Inside Revit.

Lastly, using multiple test simulations on different types of 3D modelled buildings could result into a more efficient outputs and getting the algorithm to an excessively developed smooth workflow for comfort analysis.

As a conclusion to this chapter, the third research question could be answered positively, meaning that the unobstructed work eases the transfer and accuracy of data, which is reducing time and errors when performing an evaluation on a building envelope performance. Wrapping up Chapter 4 the overall view of this research is that it succeeded in a guide which could help users to improve their knowledge about future studies and encourage new users to implement this type of solutions for their analysis on user's comfort and a building envelope performance.

## 6. CONCLUSION

Finalizing the work analyzed in all chapters, it could be confirmed that during time the research got to an ultimate progress. Starting from evaluating the programs, finding the right visual comfort indicators to analyze, constructing two types of case studies, transferring the data from BIM environment to a VPL environment, using Rhino Inside Revit, then performing simulations on both models, to producing results for daylight and glare analysis, put together the path that brought this research to the desired outcome. It is worth to mention that even though this research started with an idea to assess a parametric workflow with a smooth interoperability, it has also addressed and evaluated the different results of the performed simulations, as part of the data transfer process. Moreover an attention was not only paid on the abilities or incapacities of the above mentioned software programs, but to the precision of the carried out simulations.

As a conclusion to the analyzed data from the existing information about BPSTs, due to user-friendly interfaces, building performance analysis are not accessible to everyone; in order to generate and comprehend realistic and reliable simulation results, one must be aware of the programs' limitations and have a fundamental understanding of processes. The graphical user interfaces also have diverse functions and frequently don't utilize all the capabilities of the corresponding engine. Typically, these tools are made to be used throughout the design phase of a structure's life cycle. However, the use of the three main software programs, mentioned above, offered the ability to construct a parametric workflow, which could eventually help future analysis and studies, and could also attract architects and engineers to make an effort of learning it.

Moreover it could be addressed that the amount of transferred data from Revit to Grasshopper resulted in sufficient outcome for the daylight and glare analysis. While extracting all the geometries for both case study I and case study II, the produced algorithm is easily understandable and doesn't take much time to be applied to other constructed 3D models within Revit. The only thing that needs to be done is to open the Revit file, then access Rhino through Revit and finally open Grasshopper where the algorithm could be inputted and adjusted to the existing model.

Another thing, connected to extracting the surfaces in case study I could be the modification of the geometry within Grasshopper. The construction of sills and stiles, could take more time if there are multiple windows. Luckily this method was tested only on the single zone model, so the sills and stiles were built only for one window opening. In the case of the multi-zone model, the modification on the geometry in which there is a curved wall, the different approaches that could be considered, could also result in more time consuming action. However, following the script produced in this research, the construction of the incorrect surfaces, should be simplified.

Following up the modifications on the geometry are the materials' properties. The performances of the used material's layers could also increase the time spent on modifications, before running the simulations. This means that it is better to try and collect the material properties on the side and use the thermal assets plugging them into Grasshopper components, than modify the properties in Revit and then find a unit conversion for the unit type, before plugging them in the material construction components.

Constructing the algorithm for creating a Honeybee Model, in both case studies seems to be correct and accurate. There was no difference in the script of the Grasshopper algorithm. Most importantly before running the simulations is to check all of the script components and to pay attention on input parameters, especially radiance parameters for the daylight and glare analysis. Moreover the sensor grid should be applied correctly in order to obtain accurate results at the end. Continuing with the simulation components, for the single zone model a simulation on one zone was performed. When performing a simulation with high radiance parameters, could take time, but only if there are multiple zones. For example all rooms of the multi-zone model ended in one component of Honeybee Model, after which the simulation was performed on the whole building. This allowed the optimization of the results to be performed on each room or on the whole building, which in practice could influence the production of results positively. Additionally it took more time for the simulations to finish, but the best option is to check all the inputs twice and then run the simulations, in order not to have any mistakes and to be obligated to run the simulation all over again.

The end results that were retrieved from this research, regarding visual comfort analysis, are representing the ability of the Grasshopper tool to compute such analysis for simpler and more complex building type. Due to the fact that some of the results didn't reach multiple thresholds, there is always room to go back to Revit, change the size of the window opening and then compute the simulations again. As mentioned previously, the quick experiment shows exactly how it would work if there is a need of changing a parameter for the glazing surfaces. This was the reason why those results were represented in a way of giving an option to think about backwards interoperability and using it in the future for real projects. Wrapping Chapter 6, the last research question is answered.

## **7. COMMON DISCUSSION WITH COLABORATOR IVA LAZARESKA**

As mentioned previously, the research was firstly started as a collaboration with colleague Iva Lazareska, with exploring existing BIM software programs and BPSTs. Continuing with the collection of KPIs, then resulted in dividing the research into two paths. The other research followed the analysis related to energy and thermal comfort, using different types of exportation for the 3D model from Revit to Rhino (Grasshopper). It is important to mention that both researches analyze the same models, single-zone and multi-zone model.

In the following page there is a graphical representation (*Figure 7.1*) of both approaches from the beginning until the end of the whole research, where it is shown how they start as a common groundwork, exploring data that was useful in both cases, continuing with separated workflows and combining their discussion and conclusion about pros and cons at the end.

The brief depiction of the used workflows is simplified into mainly three narrowed paths, where the panels with orange color represent the workflow followed by the collaborator, with blue color the workflow done regarding this research and with grey color is the common information. It is divided into few steps following the structure of the thesis which is the same for both researches.

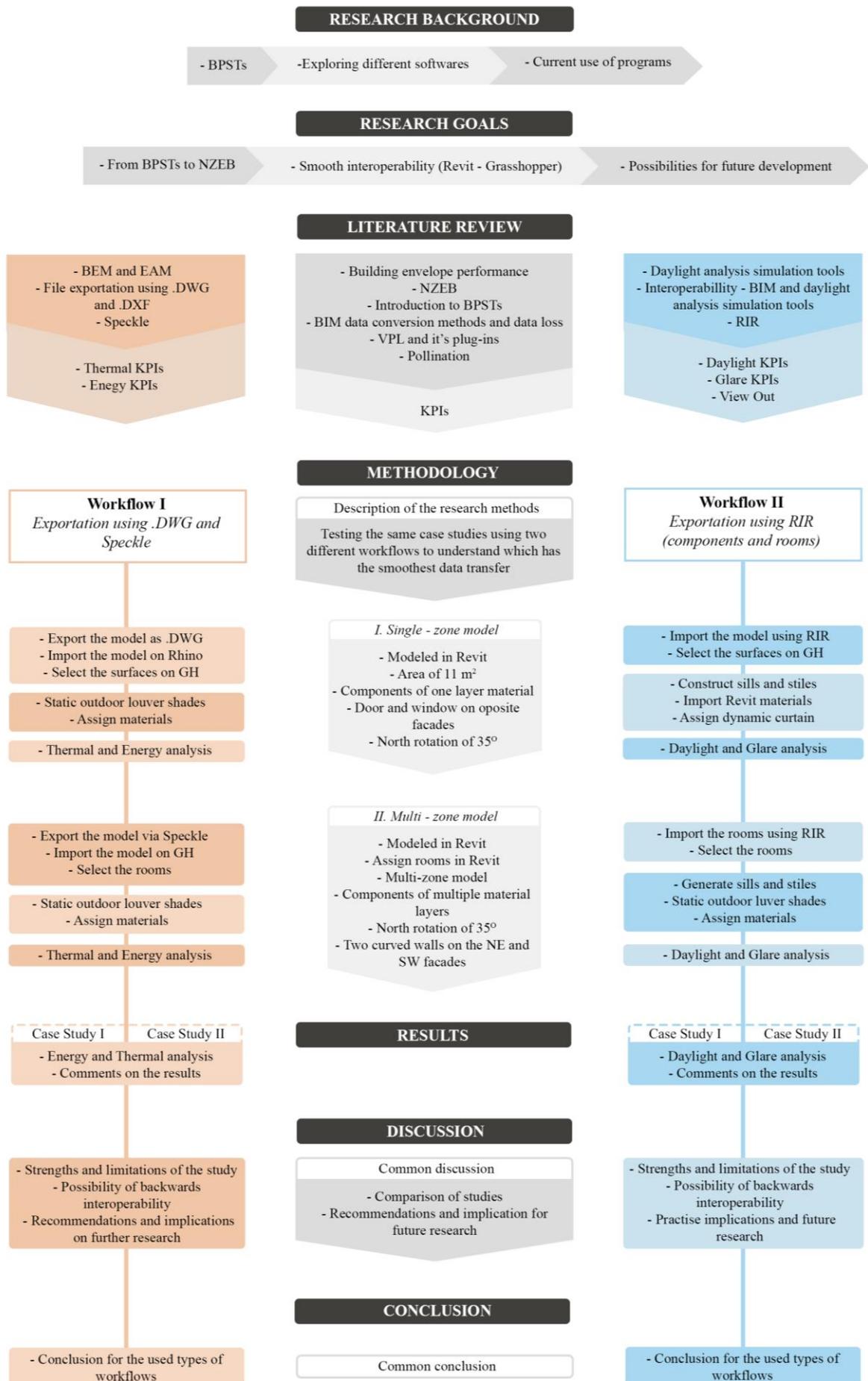


Figure 7.1 Vertical flowchart following the comparison of the two studies

## 7.1. COMPARISON OF STUDIES

Following the first exportation type, the other research has applied a traditional .DWG exportation method using the Revit program. This type of exportation was used for the single zone model. When using this exportation the importation is directly through Rhino, which means that the Rhino interface is opened separately and the .DWG file is opened there. On the other hand this is not the case when using Rhino Inside Revit. After the model was exported in Rhino, it accepts it as a mesh with triangular shapes, but that could be easily changed and transformed into rectangular surfaces. The other solution is by modifying the exportation setup within Rhino. Other action that could possibly be taken into consideration is the modification on AutoCAD. After exporting the model from Revit, it could be opened in AutoCAD and later modified.

Secondly, for the multi-zone model is used a similar extraction of the model volume as within this research. The exportation of the surfaces are same as the ones produced by Revit. This is done with the help of Speckle, which is a program that could be downloaded on Revit. The Speckle connector is opened on the Revit UI, where all desired information could be send to the cloud. When streaming, the model elements could be exported to the online cloud stream.

After this the exported elements in Grasshopper, could be used for later evaluation of the model. The few geometry modifications are present also in the other research, which means that both have the same measurements that need to be taken into considerations before continuing with other steps. In the case of the collaborator's research methods, the only information that is transferred from Revit to Grasshopper, is the geometry. This means that the interoperability is limited. If there is a need of backwards interoperability, the way of performing it, should be achieved by using the baking tool in Grasshopper. After performing this, the model could be inputted in Revit again. Due to repeating the transfer of data, being just the geometry, this type of interoperability wasn't tested in the research.

One more thing that needs to be mentioned is the time consuming and the efficiency of both methods. When running the energy and thermal simulations, the collaborator's algorithm seemed to be more efficient. Components from Honeybee Energy toolbar were implemented in the algorithm, where with a use of a connector component called "From Honeybee to OSM (Open Studio Model)" and Energy Plus as an engine the simulations were run. The mentioned efficiency derives from the use of only one component to run the simulations and then retrieve all needed results for later evaluation and visualization. On the other hand this was not the case within the research where daylight and glare simulations were performed. Even though the running process of the simulations was fast, approximately 1-2 minutes per each component, it seemed less efficient due to the use of few components from Honeybee Radiance in order to perform all needed simulations. Also it is worth to mention that in order to have more accurate results the radiance parameters need to be modified and this could add up to the running time.

Following the findings of both researches the different approaches allow the user to choose between multiple workflows which could be best suiting for the case scenario that is tested. Most importantly all exportation processes (Traditional - .DWG, Speckle and RIR) have their pros and cons.

## 7.2. RECOMMENDATIONS AND IMPLICATION FOR FUTURE RESEARCH

During the process of the researches there were few things, connected to modeling and actual performance of the simulations, which were found in common, both on Revit and Grasshopper (Rhino). As addressed previously the recommended way of modeling the 3D model of a design project, is strictly paying attention to all elements' construction. Most importantly if using the traditional exportation process or components exportation with RIR, the adjustments to all elements, such as walls, floor and roof, should be performed correctly in Revit. This means that it is better to correct the joint of all these building components within Revit UI in order to smoothly transfer the data to Grasshopper. On the other hand if having a more complex model the best way of exporting the model is with putting the rooms in Revit and extracting them as room volumes in Grasshopper, and this option is given both by Speckle and RIR. Since Honeybee on Grasshopper asks for a closed boundary surface, there is one more thing that needs to be counted as recommendation for better accuracy of the model geometry. While still being on Revit UI, if the model has multiple zones, it is better to try and use the room separator, so that all of the zones could share one surface and won't have a duplication of internal walls, which would result with a double thickness when adjusting the Honeybee Rooms for the Honeybee Model on Grasshopper. The same strategy for using only room separator could be used with both Speckle and RIR when extracting the model on Grasshopper.

Furthermore when extracting the 3D model the user should be aware of incorrectly imported surfaces, mostly when having a curved face. A suggestion is to try and find the most efficient way of constructing the algorithm for that one (or more) surface, with simplifying the use of components provided from Grasshopper toolbar.

Regarding the next steps, before performing any type of simulation, if exporting the material properties or adjusting them on Grasshopper, the units must be double checked, because it could result in inaccurate results. Both researches focused on different KPIs, but each of the performed simulations asks for correctly inputted information and if that's not the case it could give an error. Even if the design could be changed anytime, with the use of RIR, all model modifications and changes on the simulation inputs could be implemented multiple times with their corrections. But as a result of the high time-consuming it is important not to rely on this ability of the software, but to use it coherently.

Moreover when using components for running various simulations, all inputs possibilities should be checked, because on Grasshopper there are components that allow to detail the inputted data that is needed to simulate a desired analysis.

As a conclusion to this, both researches allow the users to adjust and improve the algorithms based on their needs. Speaking of future implications, both algorithms were designed in a way to analyze different case scenarios, resulting into an open door to explore all design possibilities which could range from simplified to intricate design technologies.

Lastly, the chance to take the two researches on a next level, would be to explore the backwards interoperability, where the user could have the fully workflow of interoperability between software programs such as BIM and VPL.

## 8. COMMON CONCLUSION WITH COLLABORATOR IVA LAZARESKA

After finishing both researches, a short chronological retrospective was made on the two studies, gathering all the information that they retrieved from the literature review and what they acknowledged from it. Next it continued with the way of constructing the workflow, what results were outputted from the simulations and what errors and data loss was resulted as a final comment. After they were both commented and discussed, a final conclusion was made in order to understand what are the pros and cons and how they can be used in the most correspondent way to the project design.

This common conclusion, which was acknowledged together with the collaborator, is explained in three separated stages, more precisely named as before, during and after the researches. Even though they were depicted on three graphical representations, they still connect and they are in reciprocal proportion, due to the fact that each following step depended on few previous ones.

Firstly it all starts with the period before the methodology part and the actual analysis on the case studies. Both researches have in common that all information that was gather from data providers, such as articles, books, forums, web-sites, users' experience, etc, gave a starting point, which motivated the researchers to explore how this information could help creating a base line followed by testing the limits of the possibilities and inabilities during the evaluating stage. *Figure 8.1* shows this initial cognition step by step.

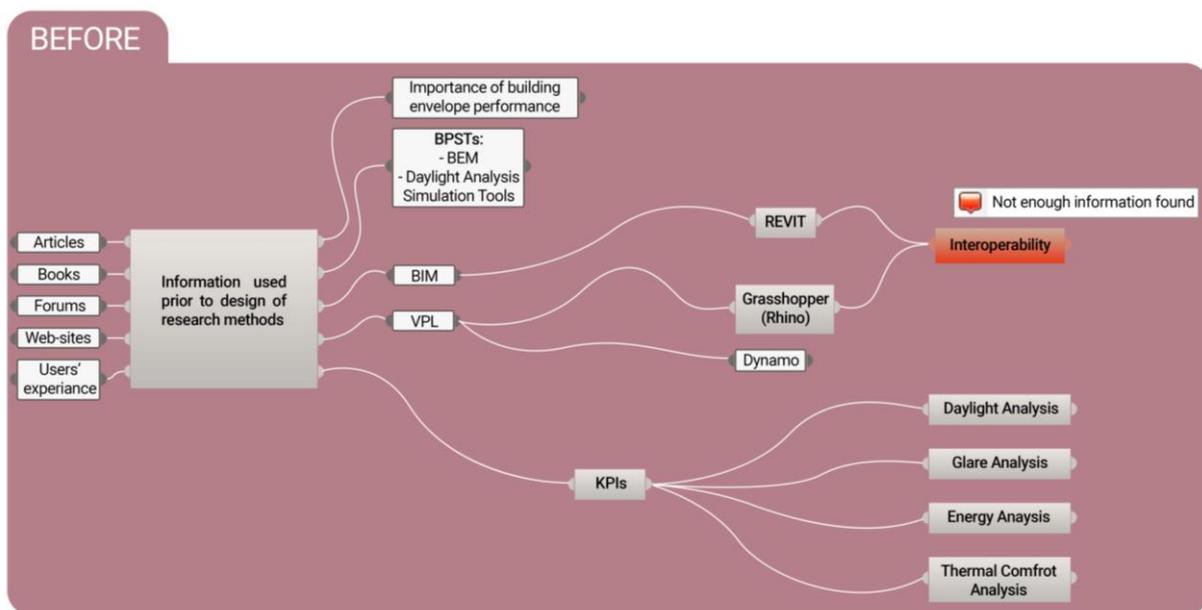
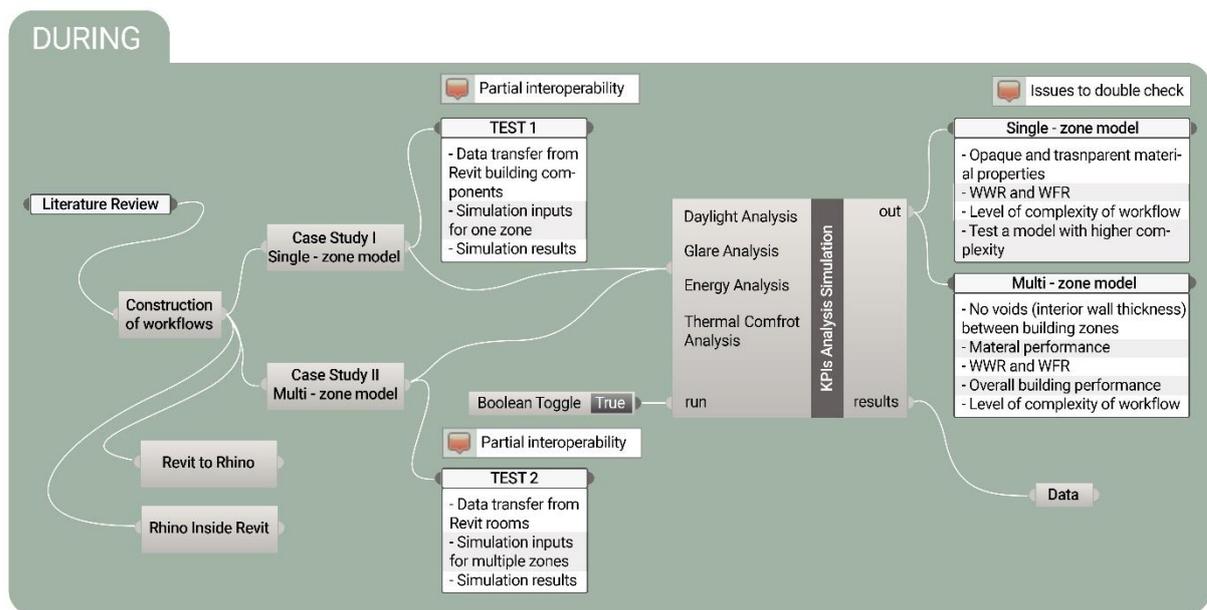


Figure 8.1 Before practice implication of case studies

Finding out that the building envelope performance is a crucial point in designing a nZEB, led us to explore what key performance indicators could be assessed in order to evaluate the overall building performance. But to perform a computational calculation for these KPIs, multiple simulation tools were assessed. The provided information was mostly derived from users' experience and forums, because most of the users solved their issues with a help of professionals that worked directly in constructing the scripts for previously mentioned simulation tools. Discovering Revit as a BIM software program, helped the researches explore more about it since it was the most used among AEC. This applies also for

Grasshopper (as VPL software program). The main problem was that for both researches there was a lack of information on how these two software programs interact with each other and what type of information is exchanged. And this is the biggest issue for users when deciding whether to include this type of analysis into their building performance evaluation or not. Moreover the second biggest problem is not only the complexity of some of the tools, but the lack of information about their use, which also applied on both researches.

Since this only gave a base line for the upcoming research phase, where there is an actual practice implication of the case studies, the more detailed exploration began right away in the second stage. *Figure 8.2* shows the steps of what came as a conclusion after performing each step during the actual construction of the workflows, the testing phase of interoperability between Revit and Grasshopper, the simulations done within Grasshopper UI (using its selected engines) and finally issues that need to be double checked.



*Figure 8.2 During practice implication of case studies*

Both researches resulted in partial interoperability within the used workflows. As previously mentioned one workflow was Revit to Rhino (with .DWG and Speckle exportation of the model) and the other one (used in current research: exportation of the model using RIR). After they both explored the way of extracting a simple geometry on Grasshopper (Case Study I: Single-zone model), the outcomes were that the main attention needs to be paid on constructing a closed surface boundary and having the glazing element directly aligned with the wall surface in order to perform the desired simulations. The partial interoperability also includes the material properties exportation from Revit to Grasshopper. While using RIR it was possible to export some of the properties of the materials, but with wrong units, that didn't correspond with the provided by Grasshopper components. On the other hand with the traditional exportation nor with Speckle, this option was not given. Even though RIR allows the user to extract material assets directly to Grasshopper, with the first case study this information

became irrelevant. Overall the algorithms in both researches, constructed within the process of preparing the model for simulation, didn't resulted with bigger issues.

After concluding that tests for only one zone are not exploring all the possibilities of the software programs, a more complex geometry was tested. One of the overall issues and problems with a model where there are multiple zones (Case Study II: Multi-zone model) is connected to the geometry, for example if there are curved surfaces, voids between internal surfaces of the zones and incorrect size of window openings according to wall and floor area. Moreover the same issue with the material properties came after testing the multi-zone model, in both research methods. Even though RIR allows material properties extraction, using the room exportation process doesn't allow transferring this type of data between Revit and Grasshopper. Also this wasn't the case with Speckle too.

In order to conclude the “during” stage the results from the simulations made both of the researchers to realize that each time the level of complexity of the inputted model on Grasshopper is higher, the results are more accurate, but on the other hand the simulations are heavier and take more time.

Lastly the stage after both researches were done, was the stage where after the simulations, both research methods' results came accurate and they could be used into developing a full knowledge on how the decisions made on both case studies impacted them and lastly the overall pros and cons of all exportation types used by both software programs, shown through the interoperability between Revit and Grasshopper (Figure 8.3).

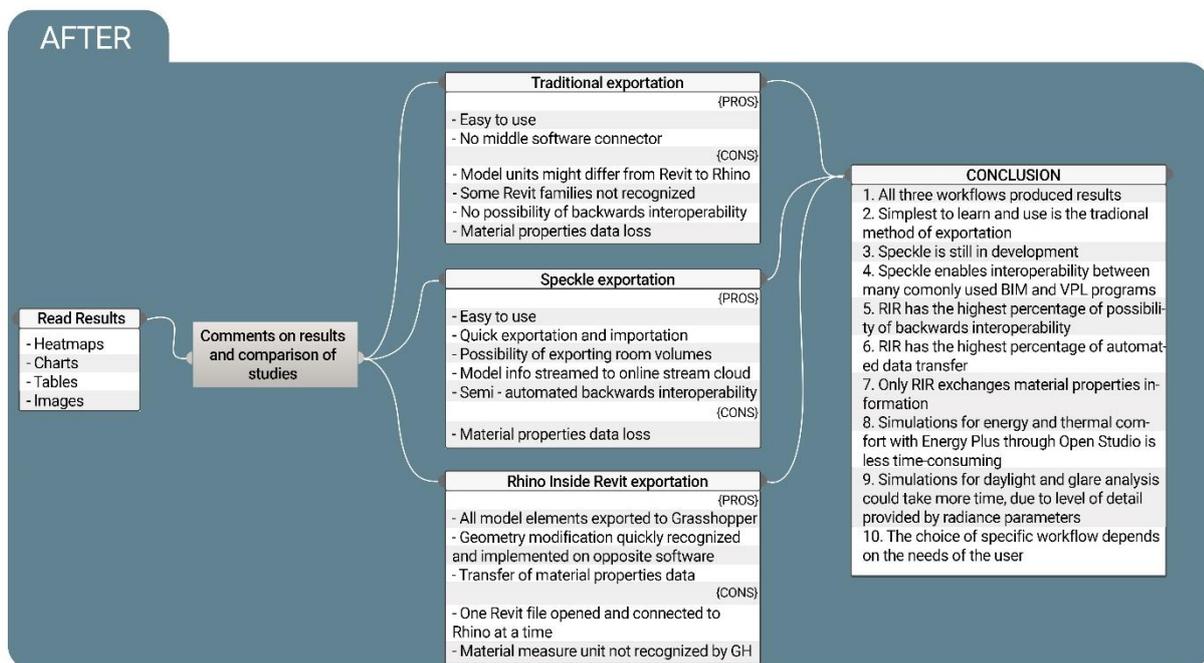


Figure 8.3 After practice implication of case studies

In conclusion to the last stage, positive comments could be given on the overall easy to use exportation types, which are both the traditional and the one with Speckle. On one side the traditional exportation doesn't need a middle software connector between Revit and Grasshopper, but on the other Speckle is used as a middle software connector, which allows quick exportation and importation of the model elements (in the collaborator's case scenario, using room exportation).

Moreover RIR could be mentioned as the best exportation tool. Even though it could be only used on Revit, connecting it to Rhino (Grasshopper), this tool provides the largest collection of components, which allows various types of extraction of the 3D model from the BIM environment.

Another important thing to mention is that with Speckle there is an opportunity to have a semi-automated backwards interoperability, which could be performed thanks to the available online stream cloud. But RIR allows an automated interoperability between Revit and Rhino (Grasshopper), which means that when some changes need to be made they could be directly transferred on the opposite software and vice versa. One thing that is considered as an obstruction is that the material properties couldn't be changed through Grasshopper and transported to Revit, but only the geometry.

Furthermore both traditional and Speckle exportation types allow to have multiple files that are opened from Revit, Rhino and Grasshopper. Which apparently is not the case when using RIR. Rhino Inside Revit allows the user to have one file opened on Revit and through the same file to have the other two files for Rhino and Grasshopper, opened. Again this is not a large regression of the overall capabilities of Rhino Inside Revit.

Finally the 10 common conclusions from both researches are presented in the same graphical representation on *Figure 8.3*, where it is worth to mention that they were constructed based on shared opinion with the collaborator Iva Lazareska. The overall conclusion was that, all exportation types produced results which could be later evaluated and help improve the design of a project. Next as simplest to use was chosen the traditional method, not only because there is no need of middle software connector, but because that type of exportation is used on every day basis even with other software programs, so it doesn't ask for much effort by the user.

Since Speckle is easy to use and is a connector not only for Revit and Grasshopper, but for other software programs, the prediction is that it has a room for development, even passing the abilities of Rhino Inside Revit.

Lastly, Rhino Inside Revit was chosen as the most capable tool with the highest percentage of interoperability between Revit and Grasshopper and an only automated backwards interoperability. It is also worth to mention that if the user wants to extract any type of material property, it could only be done by using Rhino Inside Revit.

Wrapping up Chapter 8, it can be concluded that even though different type of exportation processes were used during both of the researches, the simulations done on energy, thermal comfort, daylight and glare could be performed correctly in any case scenario. They would give the same results, same time demand and same strategy for constructing that part of the algorithm.

As an overall conclusion to Chapter 8 is that each user can decide which specific exportation type is best suiting for them, according to the experience, knowledge and needs of the project design.

## 9. REFERENCES

- Akbarieh, A. (2018). *Systematic Investigation of Interoperability Issues between Building Information Modelling and Building Energy Modelling-BIM-Based BEM Information Exchange Issues*. Master's thesis.
- Aksamija, A. A. (2013). PARAMETRIC CONTROL OF BIM ELEMENTS FOR SUSTAINABLE DESIGN IN. *PERKINS+ WILL RESEARCH JOURNAL*, 3(1), 32.
- Alisherbek, N. (2021). Development of Urban Development in the Territory of Uzbekistan. *CENTRAL ASIAN JOURNAL OF THEORETICAL & APPLIED SCIENCES*, 2(10), 24-26.
- Aryanti, F. N. (2020). Implementation of Building Information Modelling (BIM) Revit to meet the needs of the construction industry. In *IOP Conference Series: Materials Science and Engineering* (Vol. 830, p. 042054). IOP Publishing.
- Aryanti, F. N. (2020). Implementation of Building Information Modelling (BIM) Revit to meet the needs of the construction industry. In *IOP Conference Series: Materials Science and Engineering* (Vol. 830, p. 042054). IOP Publishing.
- Asl, M. R. (2014). BIM-based parametric building energy performance multi-objective optimization.
- Attia, S. H. (2012). Selection criteria for building performance simulation tools: contrasting architects' and engineers' needs. *ournal of building performance simulation*, 5(3), 155-169. doi:<https://doi.org/10.1080/19401493.2010.549573>
- Bastos Porsani, G. D.-O. (2021). Interoperability between building information modelling (BIM) and building energy model (BEM). *Applied Sciences*, 11(5), 2167.
- Bazjanac, V. &. (2007). Reduction, simplification, translation and interpretation in the exchange of model data. In *Cib w* (Vol. 78, pp. 163-168).
- Boarin, P. M.-M.-F. (2020). Understanding students' perception of sustainability in architecture education: A comparison among universities in three different continents. *Journal of Cleaner Production*, 248, 119-237. doi:<https://doi.org/10.1016/j.jclepro.2019.119237>
- Boyce, P. H. (2003). *The benefits of daylight through windows*. Rensselaer Polytechnic Institute. New York: Troy.
- Carlucci, S. C. (2015). A review of indices for assessing visual comfort with a view to their use in optimization processes to support building integrated design. *Renewable and sustainable energy reviews*, 47, 1016-1033.
- Celani, G. &. (2012). CAD scripting and visual programming languages for implementing computational design concepts: A comparison from a pedagogical point of view. *International Journal of Architectural Computing*, 10(1), 121-137.
- Choi, H. H. (2016). Toward the accuracy of prediction for energy savings potential and system performance using the daylight responsive dimming system. *Energy and Buildings*, 133, 271-280.

- ClimateStudio. (2022). *Advanced daylighting, electric lighting and conceptual thermal analysis*. Retrieved from Solemma: <https://www.solemma.com/climatestudio>
- Council, U. G. (2013). LEED reference guide for building design and construction. *US Green Building Council*.
- Day, M. (2010). *Rhino Grasshopper, AEC, Magazine, X3DMedia*. Retrieved from <http://aecmag.com>
- de Lange, P. B.-I. (2017). Socio-technical Challenges in the Digital Gap between Building Information Modeling and Industry 4.0. In *STPIS@ CAiSE* (pp. 33-46).
- Dogrusoy, I. T. (2007). A field study on determination of preferences for windows in office environments. *Building and Environment*, 42(10), 3660-3668. doi:<https://doi.org/10.1016/j.buildenv.2006.09.010>
- Eastman, C. M. (2011). BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors. *John Wiley & Sons*.
- Echenagucia, T. M. (2015). The early design stage of a building envelope: Multi-objective search through heating, cooling and lighting energy performance analysis. *Applied energy*, 154, 577-591. doi:<https://doi.org/10.1016/j.apenergy.2015.04.090>.
- Fernandez-Antolin, M. M.-L. (2022). Building Performance Simulations and Architects against Climate Change and Energy Resource Scarcity. *Earth*, 3(1), 31-44. doi:<https://doi.org/10.3390/earth3010003>
- Fernandez-Antolin, M. M.-R.-L. (2020). The relationship between the use of building performance simulation tools by recent graduate architects and the deficiencies in architectural education. *Energies*, 13(5), 1134. doi:<https://doi.org/10.3390/en13051134>
- Ghofrani, M. S. (2020). Daylight Analysis in Qajar Era Houses of Tabriz, Iran.
- Grimm, N. B. (2008). Global change and the ecology of cities. *science*, 319(5864), 756-760. doi:DOI: 10.1126/science.1150195
- Guglielmetti, R. M. (2011). OpenStudio: an open source integrated analysis platform. *NREL/CP-5500-51836*. Golden, CO, (United States): National Renewable Energy Lab.(NREL).
- Güneralp, B. Z.-V. (2017). Global scenarios of urban density and its impacts on building energy use through 2050. *Proceedings of the National Academy of Sciences*, 114(34), 8945-8950. doi:<https://doi.org/10.1073/pnas.1606035114>
- Gusc, J. &. (2018). “Oh This Learning, What a Thing It Is!”—Putting Sustainability First in Teaching Techniques and in Content. *Sustainability*, 10(8), 2803. doi:<https://doi.org/10.3390/su10082803>
- Hellinga, H. I. (2013). Daylight and view: the influence of windows on the visual quality of indoor spaces.
- Heschong, L. (1999). Heschong, L. (1999). Daylighting in Schools: An Investigation into the Relationship Between Daylighting and Human Performance.
- Heschong, L. (2021). Visual Delight in Architecture: Daylight, Vision, and View. *Routledge*.

- Hetherington, R. L. (2011). Integrated building design, information and simulation modelling: the need for a new hierarchy. Retrieved from <http://www.bs2011.org/>
- Hijazi, M. K. (2015). Bridging the gap: Supporting data transparency from BIM to BEM. *Architectural Research*, 149.
- Jabi, W. (2014). Parametric spatial models for energy analysis in the early design stages.
- Jakubiec, J. A. (2011). DIVA 2.0: Integrating daylight and thermal simulations using Rhinoceros 3D, Daysim and EnergyPlus. In *Proceedings of building simulation* (Vol. 20, pp. 2202-2209).
- Jensen, H. W. (2001). Realistic image synthesis using photon mapping. 364.
- Knaack, U. (2014). Potential for innovative massive building envelope systems - Scenario development towards integrated active systems. In *Journal of Facade Design and Engineering* (pp. 255-268).
- Knowledge Base DIALux Evo. (2021, May).
- Knowlegde Base DIALux Evo. (2021, May). Retrieved from <https://evo.support-en.dial.de/support/solutions/articles/9000121044-daylight>
- Kubba, S. (2012). Handbook of green building design and construction: LEED, BREEAM, and Green Globes. Butterworth-Heinemann.
- Ladybug Tools. (2020, Jan 4). Retrieved from Food4rhino Apps for Rhino and Grasshopper: <http://www.food4rhino.com/app/ladybug-tools>
- Lagios, K. N. (2010). Animated building performance simulation (ABPS)–linking Rhinoceros/Grasshopper with Radiance/Daysim. *Proceedings of SimBuild*, 4(1), 321-327.
- Lee, K. S. (2016). Feasibility study on parametric optimization of daylighting in building shading design. *Sustainability*, 8(12), 1220. doi:<https://doi.org/10.3390/su8121220>
- Leslie, R. P. (2012). Conceptual design metrics for daylighting. *Lighting Research & Technology*, 44(3), 277-290.
- LLC, S. (2022). "What is ALFA?". Retrieved from Solemma: <https://www.solemma.com/alfa>
- Loftness, V. H. (2006, June). Sustainability and health are integral goals for the built environment. *Healthy buildings*, 1, 1-17. Retrieved from <https://www.researchgate.net/publication/282736493>
- Mahdavinejad, M. J. (2011). Horizontal distribution of illuminance with reference to window wall ratio (wwr) in office buildings in hot and dry climate, case of iran, tehran. In *Applied Mechanics and Materials* (Vol. 110, pp. 72-76). Trans Tech Publications Ltd. doi:<https://doi.org/10.4028/www.scientific.net/AMM.110-116.72>
- Maile, T. F. (2007). *Building energy performance simulation tools-a life-cycle and interoperable perspective*. Center for Integrated Facility Engineering (CIFE) Working Paper. Retrieved from [https://www.researchgate.net/profile/Vladimir-Bazjanac/publication/237621385\\_Building\\_Energy\\_Performance\\_Simulation\\_Tools\\_-](https://www.researchgate.net/profile/Vladimir-Bazjanac/publication/237621385_Building_Energy_Performance_Simulation_Tools_-)

- Marsh, A. (2006). The interaction of radiance and ecotect. In *5th International Radiance Scientific Workshop*. Leicester, UK: De Montfort University.
- McNeil, A. &. (2016). Applicability of DGP and DGI for evaluating glare in a brightly daylit space. *Proceedings of SimBuild*, 6(1).
- Mcneil, R. F. (2019). "About Radiance". Retrieved from Radiance-online: <https://www.radiance-online.org/about>
- Merwe, C. v. (2019, Nov 6). *How to do a Daylighting Analysis in Revit 2020 using Autodesk Insight*. Retrieved from Micto Graphics: <https://mgfx.co.za/blog/building-architectural-design/how-to-do-a-daylighting-analysis-in-revit-2020-using-autodesk-insight/>
- Nabil, A. &. (2005). Useful daylight illuminance: a new paradigm for assessing daylight in buildings. *Lighting Research & Technology*, 37(1), 41-57.
- Ochoa, C. E. (2012). Considerations on design optimization criteria for windows providing low energy consumption and high visual comfort. *Applied energy*, 95, 238-245. doi:<https://doi.org/10.1016/j.apenergy.2012.02.042>
- Pinheiro, S. W. (2018). MVD based information exchange between BIM and building energy performance simulation. *Automation in Construction*, 90, 91-103.
- Qudratovich, B. B. (2021). Personnel Issues in the Application of Nanotechnology in Construction and Architecture. *International Journal of Discoveries and Innovations in Applied Sciences*, 1(5), 248-250.
- Rashwan, A. E. (2019). Evaluation of the effect of integrating building envelopes with parametric patterns on daylighting performance in office spaces in hot-dry climate. *58(2)*, 551-557.
- Reinhart, C. &. (2006). Findings from a survey on the current use of daylight simulations in building design. *Energy and buildings*, 38(7), 824-835. doi:<https://doi.org/10.1016/j.enbuild.2006.03.012>
- Reinhart, C. F. (2001). Dynamic RADIANCE-based Daylight Simulations for a full-scale Test Office with outer Venetian Blinds. *Energy & Buildings*, 33(7), 683-697.
- Reinhart, C. F. (2006). Dynamic daylight performance metrics for sustainable building design. *Leukos*, 3(1), 7-31. doi:<https://doi.org/10.1582/LEUKOS.2006.03.01.001>
- Roudsari, M. S. (2022, July 20). "Have you tried Pollination yet?". Retrieved from Ladybug Tools Forum: <https://discourse.ladybug.tools/t/have-you-tried-pollination-yet/18784>
- Rubinstein, G. J. (1988). A New Technique for Computer Simulation of Illuminated Spaces. *ournal of the Illuminating Engineering Society*, 17(1), 80-91. doi:DOI: 10.1080/00994480.1988.10748710
- Ruck, N. A. (2000). Daylight buildings. In *A source book on daylighting systems and components*.

- Seto, K. C. (2017). Sustainability in an urbanizing planet. *Proceedings of the National Academy of Sciences*, 114(34), 8935-8938. doi:<https://doi.org/10.1073/pnas.16060371>
- Souza, S. A. (2019). Performer, consumer or expert? A critical review of building performance simulation training paradigms for building design decision-making. In *Journal of Building Performance Simulation* (12 ed., pp. 289-307).
- Šprah, N. &. (2019). Daylight provision requirements according to EN 17037 as a restriction for sustainable urban planning of residential developments. *Sustainability*, 12(1), 315.
- Tedeschi, A. (2011). *Parametric Architecture with Grasshopper® | Primer* (Le Penseur ed.).
- Thomas, P. P. (2009). Four approaches to teaching with building performance simulation tools in undergraduate architecture and engineering education. In *Journal of Building Performance Simulation* (Vol. 2, pp. 95-114). doi:10.1080/19401490802592798
- Ullah, K. L. (2019). An overview of BIM adoption in the construction industry: Benefits and barriers. In *0th Nordic Conference on Construction Economics and Organization*. Emerald Publishing Limited.
- Vehmaa, A. K. (2018). Building a more sustainable society? A case study on the role of sustainable development in the education and early career of water and environmental engineers. *Sustainability*, 10(8), 2605. doi:<https://doi.org/10.3390/su10082605>
- Vos, J. J. (2003). Reflections on glare. *Lighting Research & Technology*, 35(2), 163-175.
- Ward, G. J. (1994). The RADIANCE lighting simulation and rendering system. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (pp. 459-472). doi:<https://doi.org/10.1145/192161.192286>
- Wetter, M. &. (2004). A comparison of deterministic and probabilistic optimization algorithms for nonsmooth simulation-based optimization. In *Building and Environment* (Vol. 39(8), pp. 989–999.). doi:<https://doi.org/10.1016/j.buildenv.2004.01.022>.
- Yerjanovich, Y. B. (2021). ABOUT THE URBAN PLANNING PRACTICE OF THE URDA FORTRESS OF ANCIENT JIZZAK. *International Journal of Discoveries and Innovations in Applied Sciences*, 1(5), 148-151.