

TECHNICAL UNIVERSITY OF CRETE

DOCTORAL THESIS

---

**Development of Methodologies for  
Automatic Thermal Model Generation  
using Building Information Models**

---

*Author:*

Georgios GIANNAKIS

*Supervisor:*

Dr. Dimitrios ROVAS

*Submitted to the School of Production Engineering and Management  
in partial fulfillment of the requirements for the degree of  
Doctor of Philosophy at the Technical University of Crete*

July 4, 2015



# *Abstract*

School of Production Engineering and Management

Doctor of Philosophy

## **Development of Methodologies for Automatic Thermal Model Generation using Building Information Models**

by Georgios GIANNAKIS

Following the recent requirement for efficient allocation of energy resources in the building sector, the use of Building Energy Performance (BEP) simulations became more and more frequent during the design as well as the operation phases. However, BEP simulation models' generation require significant effort for set-up, limiting the potential utilization of modeling in both phases. The substantial effort stems from the difficulty to collect and appropriately define the input data for accurate BEP simulation modeling, which are categorized as follows: static data, containing information on the geometry, construction and actuating energy systems; and dynamic data, including factual (sensed) data, along with forecasts for pertinent parameters (e.g. weather, occupancy), used to bridge the “simulated” and “real” worlds, reducing or even mitigating uncertainties.

Concerning the static data, a methodology for semi-automated BEP (known also as thermal) simulation model creation could make the BEP simulation modeling process much more expedient and as such lower to threshold for the use of such models. Building Information Models (BIMs) are information-rich repositories that could be used to streamline and expedite the collection of such information. The Industry Foundation Classes (IFC) BIM schema provide static building information that include geometric configuration and material properties, but in a form that might not be directly usable for the generation of thermal simulation models due to the absence of 2<sup>nd</sup>-level space boundaries information. Even if 2<sup>nd</sup>-level space boundaries information exists, an IFC to a specific BEP simulation engine's input data mapping must be performed. Towards modeling an automatic data mapping process, the modeler must have expert knowledge of the IFC and the specific BEP simulation engine input data structures. A precise data mapping can lead to a BEP model's geometry of high detail. Here, another recurring theme arises that the modeler has to address during the BEP simulation models' geometry data generation: the decision on how accurate the model should be. For Control

Design tasks, the accuracy requirements are markedly different, compared to the simulation model for energy auditing purposes. Control-Design processes require a model that is able to capture the sensitivities and trends but no high accuracy is necessary. Concerning the dynamic data, for any BEP simulation related task, the availability of sensed dynamic data in a building presents an opportunity to eliminate with minimal effort the uncertainties of the BEP model. Moreover, in-situ weather measurements contribute to the development of an accurate BEP model without increasing the computational complexity. These measured weather data are forwarded to the model through specifically formatted weather files. However, it is quite common that weather data do not include information concerning solar radiation, but only a percentage of cloud coverage. Even if global radiation data are available, a crucial input in the simulation of building's energy performance is the availability of both diffuse and direct solar radiation data. Nevertheless, in most cases, measured data of diffuse and direct radiation are not available.

In this thesis, initially a methodology is presented for (semi-) automated generation of thermal simulation models' static data, including: a query on the building data model — embedded in the relative IFC file — requesting geometry-related information; a processing of the acquired data by a 2<sup>nd</sup>-level space boundary identification algorithm, the Common Boundary Intersection Projection (CBIP) algorithm; and a transformation process that converts the geometry information of IFC, along with the data obtained from the CBIP algorithm, to an EnergyPlus and/or TRNSYS — widely used BEP simulation engines — input file. Next, commonly the BEP simulation model's geometry derived by applying the aforementioned methodology, is of high detail due to numerous surfaces and thermal zones, increasing the simulation runtime, and as such not suitable for a Control Design process. Hence, simulation speed-up approaches are investigated, that focus on geometry and/or zoning reduction, in a way that reduces complexity while maintaining features of the simulation. Establishing a link between the BEP simulation static data and the building's sensed measurements so that the dynamic schedules can be incorporated in the simulation, is of paramount importance. This link, especially desirable also for the testing and design of control strategies, is adopted, following a widely used co-simulation framework, the Buildings Control Virtual Test Bed (BCVTB). Finally, among a plethora of methods for estimating the diffuse (or direct) fraction of global solar radiation, the question of the best method has been not fully settled, a task that this thesis tries to address.

The methodology for (semi-) automated generation of thermal simulation models' static data is applied to real buildings and its results are presented, highlighting the ability in handling non-convex geometries and generating all the possible BEP simulation model's geometry data. The methodology facilitates significantly the overall process of energy

simulation model creation from IFC geometric data. Next, the simulation speed-up approaches are evaluated with respect to accuracy and computational effort, using three test buildings. Moreover, among a plethora of existing experiments on real buildings, one is chosen to be presented highlighting the necessity of the co-simulation setup towards eliminating the building's uncertainties. Concerning the solar radiation models' efficiency, results of our investigation are presented, indicating that there is no method largely better than the others.

## Περίληψη

Τα τελευταία χρόνια ιδιαίτερο ενδιαφέρον έχει παρουσιαστεί στον τομέα της ενεργειακής απόδοσης και εξοικονόμησης κτηρίων, καθώς η ταχεία αύξηση της ενεργειακής κατανάλωσης έχει ήδη εγείρει ανησυχίες για την εξάντληση των συμβατικών ενεργειακών πόρων. Προς αυτή την κατεύθυνση, η χρήση θερμικών μοντέλων λεπτομερούς προσομοίωσης κτηρίων (Building Energy Performance simulation — BEPs) είναι ιδιαίτερα ενδιαφέροντα και προσδίδει σημαντικές δυνατότητες κατά τις φάσεις σχεδιασμού και λειτουργίας, καθώς τα μοντέλα αυτά δύναται να προβλέψουν με υψηλή ακρίβεια την ενεργειακή συμπεριφορά των κτηρίων. Τα κτήρια αντιμετωπίζονται ως πολύπλοκα συστήματα και μια λεπτομερής προσομοίωση απαιτεί να ληφθούν υπόψη τα πραγματικά κλιματικά δεδομένα, η γεωμετρία, τα υλικά του κτηρίου, τα συστήματα θέρμανσης — αερισμού — κλιματισμού και η συμπεριφορά του χρήστη. Τα δεδομένα αυτά μπορούν να κατηγοριοποιηθούν ως εξής: στατικά δεδομένα, που περιέχουν πληροφορίες για τη γεωμετρία, τα υλικά κατασκευής και τα συστήματα θέρμανσης — αερισμού — κλιματισμού, και δυναμικά δεδομένα, τα οποία περιγράφουν τα χρονοδιαγράμματα λειτουργίας διαφόρων παραμέτρων του κτηρίου καθώς και τα ισχύοντα κλιματικά δεδομένα. Συνεπώς, στοχεύοντας στην ανάπτυξη ενός λεπτομερούς μοντέλου θερμικής προσομοίωσης απαιτείται η λεπτομερής καταγραφή και προώθηση των δεδομένων αυτών σε ένα λογισμικό προσομοίωσης, ορίζοντας τα δεδομένα εισόδου του εν λόγω λογισμικού. Απόρροια της δυσκολίας να συλλεγούν και να οριστούν καταλλήλως τα δεδομένα εισόδου αποτελεί ο υψηλός φόρτος εργασίας για την ανάπτυξη μοντέλων BEPs.

Όσον αφορά στα στατικά δεδομένα, η ανάπτυξη μίας μεθοδολογίας για ημι-αυτόματη δημιουργία μοντέλων BEPs θα μπορούσε να επιταχύνει τη διαδικασία μοντελοποίησης. Προς αυτή την κατεύθυνση, τα Κτηριακά Πληροφοριακά Μοντέλα (Building Information Models – BIM) είναι πλούσια σε πληροφορία αρχεία που θα μπορούσαν να χρησιμοποιηθούν για την επίσπευση της διαδικασίας συλλογής των στατικών δεδομένων. Αναφερόμενοι στη γεωμετρία, το ανοικτό πρότυπο τυποποιημένων κλάσεων Industry Foundation Classes (IFC) μπορεί να χρησιμοποιηθεί για την καταγραφή της στατικής αυτής πληροφορίας, αλλά σε μορφή που συνήθως δεν δύναται άμεσα να αξιοποιηθεί για την παραγωγή των μοντέλων θερμικής προσομοίωσης λόγω της απουσίας των 2<sup>ου</sup> επιπέδου χωρικών ορίων (2<sup>nd</sup> level space boundaries). Αν και το πρότυπο IFC υποστηρίζει την καταγραφή των 2<sup>ου</sup> επιπέδου χωρικών ορίων, οι τρέχουσες εκδόσεις εργαλείων που εξάγουν τα αρχεία IFC αδυνατούν να περιγράψουν ορθώς τα όρια αυτά. Σε περιπτώσεις όπου η πληροφορία των 2<sup>ου</sup> επιπέδου χωρικών ορίων είναι διαθέσιμη, απαιτείται μια αντιστοίχιση των δεδομένων του IFC σε δεδομένα εισόδου για ένα συγκεκριμένο λογισμικό προσομοίωσης. Μια ακριβής αντιστοίχιση των δεδομένων αυτών μπορεί να οδηγήσει σε μια υψηλής ακρίβειας γεωμετρία του μοντέλου BEPs. Σε αυτό το σημείο, ένα ερώτημα το οποίο ο δημιουργός του μοντέλου θα κληθεί να απαντήσει είναι το πόσο ακριβές πρέπει να είναι το μοντέλο θερμικής

προσομοίωσης. Η ακρίβεια ενός μοντέλου διαφοροποιείται ανάλογα με τον σκοπό για τον οποίο χρησιμοποιείται. Για παράδειγμα, οι απαιτήσεις ακρίβειας των θερμικών μοντέλων που χρησιμοποιούνται κατά την φάση ανάπτυξης προηγμένων στρατηγικών ελέγχου των ενεργειακών στοιχείων του κτηρίου (όπως για παράδειγμα των συστημάτων κλιματισμού και θέρμανσης, καθώς και συστήματων σκίασης και φυσικού αερισμού) διαφέρουν σημαντικά σε σχέση με τις απαιτήσεις των μοντέλων προσομοίωσης που χρησιμοποιούνται κατά τις ενεργειακές επιθεωρήσεις. Κατά την φάση ανάπτυξης προηγμένων στρατηγικών ελέγχου, απαιτείται ένα μοντέλο χαμηλής υπολογιστικής πολυπλοκότητας, που είναι σε θέση να συλλάβει τις ευαισθησίες και τις τάσεις κατά τη μεταβολή των συνθηκών, αλλά η υψηλή ακρίβεια δεν είναι απαραίτητη. Όσον αφορά στα δυναμικά δεδομένα, η διαθεσιμότητα μετρούμενων από αισθητήρες δυναμικών δεδομένων σε ένα κτήριο αποτελεί μια ευκαιρία για την εξάλειψη αβεβαιοτήτων του μοντέλου BEPs χωρίς να αυξάνεται η υπολογιστική πολυπλοκότητα. Αναφερόμενοι στις καιρικές συνθήκες, τα δεδομένα αυτά είναι σύνηθες να μην περιλαμβάνουν πληροφορία σχετικά με την ηλιακή ακτινοβολία, παρά μόνο το ποσοστό κάλυψης του ουρανού από σύννεφα. Ακόμα και αν υπάρχουν διαθέσιμα στοιχεία της συνολικής ηλιακής ακτινοβολίας, ένα κρίσιμο στοιχείο για την προσομοίωση της ενεργειακής απόδοσης των κτηρίων είναι η διαθεσιμότητα δεδομένων τόσο της διάχυτης όσο και της άμεσης ηλιακής ακτινοβολίας, δεδομένα τα οποία επί το πλείστον δεν είναι διαθέσιμα.

Σε αυτή την διατριβή, αρχικά παρουσιάζεται μια μεθοδολογία για την ημι-αυτόματη δημιουργία των στατικών δεδομένων μοντέλων θερμικής προσομοίωσης, αποτελούμενη από τα εξής βήματα: 1) συλλογή δεδομένων που περιγράφουν τη γεωμετρία του κτηρίου από το σχετικό αρχείο IFC, 2) επεξεργασία των δεδομένων που αποκτήθηκαν από έναν αλγόριθμο υπολογισμού των 2<sup>ου</sup> επιπέδου χωρικών ορίων (αλγόριθμος CBIP), και 3) εφαρμογή μιας διαδικασίας μετασχηματισμού που μετατρέπει την πληροφορία της γεωμετρίας του αρχείου IFC και τα αποτελέσματα του αλγορίθμου CBIP σε δεδομένα εισόδου των λογισμικών θερμικής προσομοίωσης EnergyPlus και TRNSYS. Στη συνέχεια, η γεωμετρία του μοντέλου BEPs που προκύπτει εφαρμόζοντας την προαναφερθείσα μεθοδολογία είναι υψηλής λεπτομέρειας λόγω υψηλού αριθμού επιφανειών και θερμικών ζωνών, αυξάνοντας το χρόνο εκτέλεσης της προσομοίωσης και αποκλείοντας τη χρήση τους κατά τη φάση ανάπτυξης προηγμένων στρατηγικών ελέγχου των ενεργειακών στοιχείων του κτηρίου. Συνεπώς, διερευνώνται μεθοδολογίες επιτάχυνσης της προσομοίωσης που εστιάζουν στη μείωση του πλήθους των επιφανειών ή / και των θερμικών ζωνών έτσι ώστε να μειωθεί η πολυπλοκότητα του μοντέλου. Έπειτα, ο καθορισμός μιας σύνδεσης του μοντέλου BEPs με μετρήσεις αισθητήρων του κτηρίου είναι υψίστης σημασίας. Η σύνδεση αυτή, ιδιαίτερα επιθυμητή επίσης για τη δοκιμή και το σχεδιασμό των στρατηγικών ελέγχου, επιτυγχάνεται χρησιμοποιώντας ένα ευρέως γνωστό λογισμικό ανταλλαγής δεδομένων, το Building Controls Virtual Test Bed (BCVTB). Τέλος, μεταξύ μιας πληθώρας μεθόδων για την εκτίμηση της διάχυτης και

της άμεσης ηλιακής ακτινοβολίας, το ερώτημα της καταλληλότερης μεθόδου δεν έχει πλήρως διευθετηθεί, ερώτημα στο οποίο η διατριβή αυτή προσπαθεί να απαντήσει.

Η μεθοδολογία για την ημι-αυτόματη δημιουργία μοντέλων θερμικής προσομοίωσης εφαρμόστηκε σε πραγματικά κτήρια και τα αποτελέσματά τους παρουσιάζονται, αναδεικνύοντας την ικανότητα της μεθοδολογίας στο χειρισμό πολύπλοκων (μη-κυρτών) γεωμετριών και στην παραγωγή όλων των δεδομένων της γεωμετρίας που απαιτούνται από το μοντέλο θερμικής προσομοίωσης. Η μεθοδολογία επιταχύνει σημαντικά τη συνολική διαδικασία δημιουργίας μοντέλων θερμικής προσομοίωσης, χρησιμοποιώντας τη γεωμετρική πληροφορία που περιλαμβάνεται στο αρχείο IFC. Στη συνέχεια, οι μεθοδολογίες επιτάχυνσης της προσομοίωσης αξιολογήθηκαν με βάση την ακρίβεια και την υπολογιστική πολυπλοκότητα σε πραγματικά κτήρια. Επιπλέον, από ένα πλήθος πειραμάτων που πραγματοποιήθηκαν στα κτήρια αυτά, ένα επιλέχθηκε να παρουσιαστεί, τονίζοντας την αναγκαιότητα της σύνδεσης των μετρήσεων αισθητήρων με το θερμικό μοντέλο για την εξάλειψη των αβεβαιοτήτων του μοντέλου θερμικής προσομοίωσης. Τέλος, οι διάφορες μέθοδοι εκτίμησης της διάχυτης και άμεσης ακτινοβολίας αξιολογήθηκαν, υποδεικνύοντας ότι δεν υπάρχει μία μέθοδος αισθητά καλύτερη από τις υπόλοιπες.

# *Acknowledgements*

Reaching the end of this thesis I realized that it was the fruit of personal struggle but at the same time, assistance and support of many people to whom I would like to express my sincere gratitude.

First, I would like to thank my thesis advisor Professor Dimitrios V. Rovas, whose expertise, motivation and encouragement added considerably to my experience. He provided me with guidance and support to the world of research, introducing me to the astonishing field of Building Energy Performance Simulation.

I would like also to thank Professor Ioannis K. Nikolos and Professor Evangelos Th. Grigoroudis for their valuable suggestions and efforts as co-advisors. Moreover, I would like to thank all the members of the examination committee for their valuable comments and suggestions, which greatly improved the quality of the final document.

I could not miss referring to the contribution of my colleagues. My sincere gratitude to my friend Dr. Georgios N. Lilis, for all the work we did together, especially concerning the 2<sup>nd</sup>-level space boundary generation algorithm, and to my best friend Mr. Georgios D. Kontes for his invaluable help through stimulating discussions around computationally demanding tasks and for his great sense of humor.

I would like to express my love to my parents, Ioannis and Dimitra, and my brother Costas, for the support they provided me through my entire life. Last but certainly not least, many thanks to Efi without whose love and encouragement, I would not have finished this thesis. This thesis is dedicated to them.

In conclusion, I gratefully acknowledge for a part of this work the financial assistance of PEBBLE and BaaS projects. My thanks to colleagues from the Cartif research centre and the Fraunhofer Institute for Building Physics for the thorough information provided concerning two real buildings, and to Professor Denia Kolokotsa, who did not hesitate to provide us with a TRNSYS17 license.

# Contents

<b>Abstract</b>	<b>ii</b>
<b>Acknowledgements</b>	<b>viii</b>
<b>Contents</b>	<b>ix</b>
<b>List of Figures</b>	<b>xiii</b>
<b>List of Tables</b>	<b>xvi</b>
<b>Abbreviations</b>	<b>xviii</b>
<b>Symbols</b>	<b>xx</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Objectives . . . . .	10
1.2 Outline . . . . .	11
1.2.1 Buildings used for Testing and Application of Proposed Methodologies . . . . .	14
<b>2 Background</b>	<b>16</b>
2.1 BEP Simulation Methodologies . . . . .	16
2.1.1 Quasi-static Calculation Methodologies . . . . .	17
2.1.2 Transient Calculation Methodologies . . . . .	20
2.2 BEP Simulation Engines . . . . .	23
2.2.1 EnergyPlus . . . . .	26
2.2.2 TRNSYS . . . . .	27
2.2.3 Modelica . . . . .	28
2.3 BEP Simulation Data . . . . .	29
2.3.1 Static Data – Building Geometry . . . . .	30
2.3.1.1 First and Second Level Space Boundaries . . . . .	33
2.3.2 Static Data – Building Materials . . . . .	36
2.3.3 Static Data – Systems . . . . .	38
2.3.4 Dynamic Data – Schedules . . . . .	39
2.3.5 Other Simulation Parameters . . . . .	40
2.3.6 Dynamic Data – Weather Data . . . . .	41

2.3.6.1	Solar Radiation Models . . . . .	42
2.4	Simulation Speed-Up . . . . .	43
2.5	Co-Simulation . . . . .	44
<b>3</b>	<b>CBIP Algorithm</b>	<b>46</b>
3.1	IFC Data Quality . . . . .	46
3.1.1	Geometric Error Classification . . . . .	47
3.2	Geometric Defintions . . . . .	49
3.2.1	Boundary representation . . . . .	49
3.2.2	Boundary Space Partitioning tree representation . . . . .	50
3.3	Geometric Operations . . . . .	51
3.3.1	Polygon clipping operators . . . . .	51
3.3.2	Polygon set partition function . . . . .	52
3.3.3	Polygon set clipping functions . . . . .	54
3.4	CBIP algorithm stages . . . . .	56
3.4.1	Identification . . . . .	57
3.4.2	Boundary Surface Extraction . . . . .	58
3.4.2.1	Opening Construction Subtraction process . . . . .	60
3.4.3	Common Boundary Intersection . . . . .	61
3.4.3.1	Primary types of common boundary surfaces . . . . .	62
3.4.3.2	Derived types of surfaces (Environment surfaces) . . . . .	62
3.4.4	Boundary Intersection Projection . . . . .	64
3.4.4.1	Second level boundaries of type 2a . . . . .	66
3.4.4.2	Second level boundaries of type 2b and 2c . . . . .	67
3.4.4.3	Connectivity information . . . . .	67
3.5	Design requirements . . . . .	68
3.5.1	Building site and spaces . . . . .	68
3.5.2	Curved solids . . . . .	68
3.6	Design recommendations . . . . .	69
3.6.1	Nonzero volume intersections . . . . .	69
3.6.2	Space-environment surfaces . . . . .	69
3.6.3	Curtain walls . . . . .	69
3.6.4	Suspended ceilings . . . . .	70
3.6.5	Orientation of boundary surfaces . . . . .	70
3.7	Demonstration Examples . . . . .	71
<b>4</b>	<b>BEP Model Generation</b>	<b>73</b>
4.1	Data Requirements . . . . .	73
4.2	IFC Data Refinement . . . . .	77
4.3	EnergyPlus IDD-classes . . . . .	81
4.4	CBIPtoIDF – EnergyPlus . . . . .	82
4.4.1	Opaque and Glazing Materials . . . . .	82
4.4.2	Constructions’ Material Layer Bedding . . . . .	83
4.4.3	Thermal Zones . . . . .	83
4.4.4	Opaque Building Surfaces . . . . .	84
4.4.5	Fenestration Surfaces . . . . .	85
4.5	CBIPtoIDF – TRNSYS3d . . . . .	86

4.6	Illustrative Example . . . . .	87
<b>5</b>	<b>Simulation Speed-up Approaches</b>	<b>89</b>
5.1	Geometry Reduction Approaches . . . . .	89
5.1.1	FIBP Building – Leveraging Geometry’s Periodicity . . . . .	91
5.1.2	TUC Building – Co-Simulation Approach . . . . .	92
5.1.3	Simulation Results . . . . .	94
5.1.3.1	FIBP Building . . . . .	95
5.1.3.2	TUC Building . . . . .	98
5.2	Zoning Reduction Approaches . . . . .	100
5.2.1	Hierarchical Clustering Analysis . . . . .	100
5.2.2	Koopman Mode Analysis . . . . .	103
5.2.3	Automatic Generation of Models . . . . .	105
5.2.4	Simulation Results . . . . .	110
<b>6</b>	<b>Dynamic Data to BEP simulation</b>	<b>115</b>
6.1	Co-Simulation . . . . .	115
6.1.1	EnergyPlus – BCVTB data exchange . . . . .	118
6.1.1.1	IDF file . . . . .	118
6.1.1.2	XML file . . . . .	119
6.1.1.3	Simulator actor – EnergyPlus . . . . .	120
6.1.2	Matlab – BCVTB data exchange . . . . .	121
6.1.2.1	M file . . . . .	121
6.1.2.2	Simulator actor – Matlab . . . . .	122
6.1.3	TRNSYS - BCVTB data exchange . . . . .	123
6.1.3.1	BCVTB component and dck file . . . . .	123
6.1.3.2	Simulator actor – TRNSYS . . . . .	123
6.1.4	Experiment . . . . .	124
6.2	GSR-Virtual Sensor: Global Solar Radiation prediction based on percent- age of cloud coverage data . . . . .	126
6.3	DDSR-Virtual Sensor: Diffuse fraction of global solar radiation models . . . . .	129
6.4	DDSR-Virtual Sensor: Direct Irradiance models . . . . .	130
6.5	Solar Radiation Models: Comparative studies . . . . .	132
6.6	Solar Radiation Model’s Impact on BEP simulation . . . . .	138
6.6.1	Building’s Geometry and Zoning . . . . .	138
6.6.2	Climate Data . . . . .	140
6.6.3	Shading . . . . .	141
6.7	Solar Radiation Models Comparison and Results . . . . .	141
<b>7</b>	<b>Conclusions and Future Work</b>	<b>144</b>
<b>A</b>	<b>CBIP Aglorithm Results in XML-format Files</b>	<b>148</b>
<b>B</b>	<b>Solar Radiation Data</b>	<b>151</b>
B.1	Databases Providing Solar Radiation Data . . . . .	151
B.2	Data Quality Control . . . . .	154

---

<b>C Solar radiation models' Results</b>	<b>156</b>
--	------------

<b>Bibliography</b>	<b>177</b>
---------------------	------------

# List of Figures

1.1	Amalgamation of discussion in this Section . . . . .	3
1.2	BEP simulation preparation – Current Practice, Expert Knowledge . . . .	4
1.3	BEP simulation preparation – Leveraging BIM and Sensed Data . . . . .	7
1.4	Generating Simulation Models for Control-Design Tasks . . . . .	9
1.5	Static Data – Transformation of an IFC file’s geometry to a BEP simulation model’s geometry for energy auditing or control design purposes . . .	12
1.6	Dynamic Data – Improve the BEP simulation model utilizing in situ weather data (and/or weather predictions and a co-simulation set-up . . .	13
1.7	Demonstration Buildings of the proposed methodologies . . . . .	14
1.8	TUC building – full-scale simulation model . . . . .	15
1.9	FIBP building – full-scale simulation model . . . . .	15
2.1	Schematic of the ISO 13790:2008 calculation methodology [38] . . . . .	19
2.2	5R1C Representation of a zone [38] . . . . .	20
2.3	CFD calculations for the TUC building [80] . . . . .	22
2.4	Utilization share of major simulation programs in building optimization [70] . . . . .	25
2.5	From gbXML to BEP simulation models — attempts for an automated data translation . . . . .	31
2.6	From IFC to BEP simulation models — attempts for an automated data translation . . . . .	32
2.7	Cases where building elements are subdivided . . . . .	34
2.8	Special types of 2 <sup>nd</sup> -level space boundaries . . . . .	36
2.9	Example of a building’s element construction . . . . .	37
3.1	Design error examples – A. Wall-Slab and B: Space-Wall . . . . .	47
3.2	Space incorrect and correct definitions example . . . . .	48
3.3	B-reps of a slab with correctly and incorrectly oriented surfaces . . . . .	48
3.4	B-rep of a building entity by a set of boundary surfaces . . . . .	49
3.5	Cross section and BSP-tree representation for a set of polygon boundary surfaces forming a non-convex region . . . . .	50
3.6	Illustration of polygon clipping operators $c_1$ and $c_2$ on polygon $A_i$ by polygon $B_j$ . . . . .	52
3.7	Illustration of the polygon set partition function . . . . .	53
3.8	Results of clipping functions – $\mathcal{A}$ is the polygon set of a clipped polyhedron and $\mathbf{B}$ is the clipping polyhedron . . . . .	54
3.9	Illustration of OCS process applied on a rectangular, wall containing door and window openings . . . . .	60

3.10	Part of IFC Coordination View 2.0 EXPRESS-G schema [15] containing the required classes for CBIP algorithm. Sub-classes which require calculations (performed by BSE) are indicated by dashed blocks. . . . .	61
3.11	Primary types of common boundary surfaces referring to building constructions (A cases) and Derived environment surfaces (B cases) . . . . .	63
3.12	Geometrical illustration of CBI process on two polyhedrons (Top) – Plan view of the resulted Common Boundaries (CB) for a three space building (Bottom) . . . . .	64
3.13	Geometrical illustration of BIP process on two polyhedrons (Top) – Illustration of: C-V and O-V CBIP surface pairs, V-V Common Boundaries (CB) and extracted 2b space boundary surface types, for a building space (Bottom) . . . . .	65
3.14	Design requirements for curtain walls (Left) and suspended ceilings (Right). . . . .	70
3.15	Categories of surface pairs of CARTIF building, obtained from CBIP. . . . .	71
3.16	Categories of surface pairs of TUC building, obtained from CBIP. . . . .	72
4.1	Example of IFC4 classes referring to second level boundaries and their relating building entities . . . . .	78
4.2	Simulation model element examples . . . . .	80
4.3	CARTIF building – BIM in Revit 2014 . . . . .	87
4.4	CARTIF building – BEP simulation model geometry . . . . .	88
4.5	Illustrative examples of transformation rules' correctness – CARTIF building . . . . .	88
5.1	FIBP Building – tower simulation model . . . . .	91
5.2	Dividing the whole building to 3 sub-buildings . . . . .	93
5.3	Data exchange between the sub-building through BCVTB . . . . .	93
5.4	Upscaling of the simulation runtime for increasing number of thermal zones in TRNSYS Type 56 — base interval is one year. Left axis: general building model, right axis: fullscale FIBP & tower building model . . . . .	96
5.5	FIBP tower building model – different boundary conditions, for a simulation interval of a summer month . . . . .	96
5.6	FIBP tower building model – different boundary conditions, for a simulation interval of a winter month . . . . .	97
5.7	Simulated Air Temperature values in an office room (office 11) — whole building model and sub-building:1 model (Co-sim) . . . . .	98
5.8	Hierarchical tree – the point to cut the hierarchical tree into clusters is defined as the point where the zones' pair with the minimum distance are linked . . . . .	103
5.9	An overview of the Automatic Process for Generating Speed-up Model based on Zoning Reduction Approaches . . . . .	106
5.10	Stage 1 – Select objects — output variables of EnergyPlus — and run a simulation to receive simulated results . . . . .	107
5.11	Stage 2 – Choose a method for Zoning Reduction to receive Zones of initial IDF that are merged . . . . .	108
5.12	Stage 3 – Create new objects of appropriate IDD classes and write them to a new IDF of Grouped Zones . . . . .	109

5.13	Zoning approximations for the first two steps of Hierarchical Clustering Approach and the respective $\epsilon_{1,2}$ values of Koopman Modes Approach in CARTIF Building, using as data-set variables the zones' operative temperatures . . . . .	110
5.14	H/C energy demands' error in prediction as the number of zones increases	112
6.1	Architecture of the connection between EnergyPlus and the BCVTB and the connection between MATLAB and BCVTB during Control Design . .	117
6.2	Integration of the Control Design and the Simplified Simulation Models within BCVTB . . . . .	118
6.3	Simulator actor – EnergyPlus and its parameters . . . . .	120
6.4	Simulator actor – Matlab and its parameters . . . . .	122
6.5	Simulator actor – TRNSYS and its parameters . . . . .	124
6.6	Floor plan of TUC building and test zone O4 (office 4) . . . . .	125
6.7	Warming up period and validation period: Comparison of simulated inside temperatures and measured inside temperatures for zone O4 . . . . .	125
6.8	Control Design results – TUC building . . . . .	126
6.9	The hourly diffuse fraction vs clearness index for different polynomial correlations in Cyprus: (a) first-order; (b) second–third order; and (c) fourth-order polynomial correlations [41] . . . . .	134
6.10	Geometry of the sample building created in OpenStudio plugin for Google SketchUp . . . . .	139
6.11	Zoning of the sample building . . . . .	139
6.12	Glazing area in the front, right, back and left side of the building . . . . .	140
6.13	Locations of the weather station and the sample building at the University Campus . . . . .	140
6.14	Blinds orientations for representative blind angles [22] . . . . .	141
A.1	XSD schema of Materials.xml file . . . . .	149
A.2	XSD schema of SB.xml file . . . . .	150

# List of Tables

2.1	Describing the differences between 1 <sup>st</sup> (left) and 2 <sup>nd</sup> (right) level space boundaries . . . . .	35
3.1	Connectivity information of thermal elements. Ci. = Construction index, Isi = Internal space index, Ei = Environment index . . . . .	68
4.1	Simulation model elements and respective space boundary pairs information of thermal elements – PHY = PHYSICAL, VIR. = VIRTUAL, INT = INTERNAL, EXT = EXTERNAL . . . . .	79
5.1	Mean and standard deviation results for rooms R207, R107 and R007 of the FIBP tower building model . . . . .	97
5.2	Mean and standard deviation results for each office room of TUC building . . . . .	99
5.3	Simulation runtimes for each sub-building (Sub:1, Sub:2 and Sub:3), the parallel simulation (Co-sim) and the whole building (Full) . . . . .	99
5.4	Number of zones, walls and windows for each sub-building (Sub:1, Sub:2 and Sub:3) and the whole building (Full-scale) . . . . .	100
5.5	Results of Zoning Reduction Approaches — Different Koopman modes tolerances $\epsilon_{1,2}$ result to different H/C energy demands' error and simulation runtimes . . . . .	112
5.6	Error in model predictive capability of zones' operative temperature as the number of zones is reduced, in terms of the <i>rmse</i> and the $\delta_{\max}$ . . . . .	114
6.1	ASHRAE clear sky model coefficients [2] . . . . .	127
6.2	Monthly values of Linke turbidity factor for TUC Building location . . . . .	128
6.3	Diffuse fraction-polynomial models coefficients and their second range of clearness index . . . . .	129
6.4	Results of the Jacovides et al. hourly model's comparison study [41] . . . . .	133
6.5	Results of the Miguel et al. hourly model's comparison study [19] . . . . .	135
6.6	Results of the Torres et al. hourly model's comparison study [99] . . . . .	136
6.7	Results of the Notton et al. hourly model's comparison study [71] . . . . .	137
6.8	Results (%) with corresponding maximum relative error . . . . .	138
6.9	Thermal Zones - Area and Height . . . . .	139
6.10	Statistical errors (%) of eight different solar radiation models for ten different locations . . . . .	142
B.1	Summary of the meteorological databases providing hourly solar radiation data . . . . .	151
B.2	Climatic environment of each selected data set [78] . . . . .	154

---

C.1	Statistical errors of ten variables for two solar radiation models – Albany	157
C.2	Statistical errors of ten variables for two solar radiation models – Albuquerque . . . . .	159
C.3	Statistical errors of ten variables for two solar radiation models – Bismarck	161
C.4	Statistical errors of ten variables for two solar radiation models – Brownsville	163
C.5	Statistical errors of ten variables for two solar radiation models – Chicago	165
C.6	Statistical errors of ten variables for two solar radiation models – Glens Falls	167
C.7	Statistical errors of ten variables for two solar radiation models – Los Angeles . . . . .	169
C.8	Statistical errors of ten variables for two solar radiation models – Miami .	171
C.9	Statistical errors of ten variables for two solar radiation models – New York	173
C.10	Statistical errors of ten variables for two solar radiation models – Phoenix	175

# Abbreviations

BCVTB	Building Controls Virtual Test Bed
BEM	Building Energy Management
BEP	Building Energy Performance
BIM	Building Information Model
BIP	Boundary Intersection Projection
B-rep	Boundary representation
BSE	Boundary Surface Extraction
BSP	Boundary Space Partitioning
CBI	Common Boundary Intersection
CBIP	Common Boundary Intersection Projection
CFD	Computational Fluid Dynamics
CTF	Conduction Transfer Function
DDSR	Direct and Diffuse Solar Radiation
DLL	Dynamic Link Library
EAE	External Air Element
EOE	External Opening Element
EPBD	Energy Performance of Buildings Directive
EPW	EnergyPlus Weather file format
ESH	External Shading Element
ESRA	European Solar Radiation Atlas
ETE	External Thermal Element
FMI	Functional Mockup Interface
FMU	Functional Mockup Unit
gbXML	green building XML
GSR	Global Solar Radiation

---

GST	Geometry Simplification Tool
HVAC	Heating, Venting and Air Conditioning
IAE	Internal Air Element
ID	Identification
IDD	Input Data Directory
IDF	Input Data File
IFC	Industry Foundation Classes
ISH	Internal Shading Element
ISO	International Organization for Standardization
ITE	Internal Thermal Element
IWEC	International Weather for Energy Calculation
PID	Proportional Integral Derivative
SDF	Synchronous Data Flow
TMY	Typical Meteorological Year
TRY	Test Reference Year
WRDC	World Radiation Data Center
XML	eXensible Markup Language

# Symbols

$\partial \mathbf{A}$	B-rep of a polyhedron $\mathbf{A}$
$T_{\mathbf{A}}$	BSP-tree representation of a polyhedron $\mathbf{A}$
$\hat{n}_{A_i}$	normal vector of a boundary polygon $A_i$
$P(\mathcal{B}, \mathcal{A})$	polygon set partition function: partition of a polygon set $\mathcal{A}$ , intersected by a partition set $\mathcal{B}$ with the same outward normal vector $\hat{n}_B$
$\mathcal{A}_{ins}$	subsets of the set $\mathcal{A}$ containing polygons lying in the half-space pointed by $-\hat{n}_B$
$\mathcal{A}_{out}$	subsets of the set $\mathcal{A}$ containing polygons lying in the half-space pointed by $\hat{n}_B$
$\mathcal{A}_{cop}$	polygons coplanar with the polygons in $\mathcal{B}$ , which have opposite direction normals with $\hat{n}_B$
$\mathcal{A}_{csd}$	polygons coplanar with the polygons in $\mathcal{B}$ , which have same direction normals with $\hat{n}_B$
$F_{ins}(T_{\mathbf{B}}, \mathcal{A})$	parts of $\mathcal{A}$ , which are inside polyhedron $\mathbf{B}$
$F_{out}(T_{\mathbf{B}}, \mathcal{A})$	parts of $\mathcal{A}$ , which are outside polyhedron $\mathbf{B}$
$F_{cod}(T_{\mathbf{B}}, \mathcal{A})$	parts of $\mathcal{A}$ , which are coplanar with the surfaces of $\partial \mathbf{B}$ and have opposite outward normal vectors
$\mathcal{CB}_{AB}$	common boundaries of $\partial \mathbf{A}$ and $\partial \mathbf{B}$
$C_{az}$	air capacitance of zone $z$
$T_{az}$	mean air temperature of zone $z$
$q_i$	convective internal load $i$
$\dot{m}_i$	air mass flow rate from exterior surface $i$
$C_p$	air capacitance of zone $z$
$\dot{m}_{\infty}$	outdoor air mass flow rate
$\dot{m}_{sys}$	system supply air mass flow rate

---

$T_{sup}$	system supply air temperature
$T_{\infty}$	ambient air temperature
$T_s$	surface temperature
$q_{lw}$	sum of the longwave radiant exchange transfer between zone surfaces and the longwave radiation transfer from equipment in zone
$q_{sw}$	shortwave radiation transfer to surface from lights
$q_{sol}$	transmitted solar radiation flux absorbed at surface
$q_f$	conduction transfer through the wall
$q_c$	convective heat transfer to the zone air
$q_{CE}$	convective parts of internal loads
$q_{IV}$	sensible load caused by infiltration and ventilation air
$q_{sys}$	heat transfer from Heating Ventilating and Air Conditioning system
$f_d$	hourly diffuse fraction of global solar radiation
$k_t$	hourly clearness index
$I$	hourly global solar radiation
$I_o$	hourly extraterrestrial solar radiation
$I_{clear}$	clear sky global irradiance
$I_{b,clear}$	beam clear sky irradiance
$I_{d,clear}$	diffuse clear sky irradiance
$SE$	solar elevation angle
$Z$	solar zenith angle
$m_a$	optical air mass
$TL$	Linke turbidity factor
$p/p_o$	atmospheric pressure ratio
$alt$	elevation above the sea level
$\phi$	local latitude
$D$	solar declination
$\omega$	hour angle
$AST$	apparent solar time
$K_t$	daily clearness index
$\psi$	persistence index
$G$	monthly average global irradiance
$k'_t$	correction of the clearness index to make it independent on the sun's position

---

$W$	precipitable water in the atmosphere
$rmse$	root mean square error
$mbe$	mean bias error

# Chapter 1

## Introduction

The utilization of building thermal simulation tools stems from their capability to accurately predict the thermal performance of a building and the thermal comfort of its occupants under specific conditions. A well-defined BEP simulation model acts as surrogate of the real building, since it assimilates precisely how the building operates under certain conditions, and as such enable comparisons of different design alternatives and/or actions of building's controllable elements without being performed in reality. Following the recent requirement for efficient allocation of energy resources in the building sector, the use of Building Energy Performance (BEP) simulations became more and more frequent, during the design [16, 81] as well as the operation phases [29, 52, 53, 79]. BEP simulations and their respective calculation methodologies can be used to accomplish a variety of different tasks, and few of them are listed below:

- Energy performance estimation – In this task the energy performance of the whole building is estimated. Energy performance refers to the total energy needs, including energy used for conditioning the spaces. In detailed calculation methodologies, thermal comfort parameters can also be computed.
- Energy performance forecasting – The goal of energy performance forecasting is to estimate building energy needs in order for comfort conditions to be preserved in building spaces, during a finite future time horizon. The use of forecast data obtained from various sources is necessary in this case. As can be expected the validity of the forecasting process depends strongly on the quality of the forecasts.
- Model calibration – Although models are designed to predict the real behaviour of buildings and their systems as accurately as possible, their predictions may differ from real sensor measurements, because of a different number of reasons including: sensor measurement errors, modelling insufficiencies, or incorrect model parameter

value's estimations. Model calibration tasks rely on past sensor measurements in order to change the model parameter values and bridge the above gap.

- Virtual sensors – In many cases there are building locations where sensors cannot be placed or they are not present. In such occasions, the missing sensor data can be replaced by simulation value estimates by performing a task called “Virtual sensors”. Assuming a well conducted simulation, the computed values – “virtual” sensors – can yield acceptable approximation of real parameters.
- Components validation – System performance can degrade over time, leading to out-of specification operation. This can have adverse effects with respect to energy performance and thermal comfort. Anomaly detection and identification using simulation-based methodologies can be one of the ways, to identify such events.
- Control design – The general purpose of (supervisory) control design is to design controller that given state parameter values will return operation schedules and commands of controllable building elements. In model-based control design the calculation methodology (here synonymous to “model”) is used in combination with model-predictive control algorithms to generate such strategies.
- Control design optimization – The generated control actions, using simplified models, can have poor performance when applied to the real system. For this reason, the resulting controllers can be improved using more “accurate” building models, by performing a second optimization step. Uses of calculation methodologies can be an invaluable asset in fine-tuning/optimizing controller parameters.

The calculation methodologies used can range from “simple” quasi-steady-methods [38] to dynamic, implemented in energy-performance 3D zonal-type simulation software like EnergyPlus [18] and TRNSYS [49]. Each calculation method supports different use cases and as such the modeling assumptions and the associated inputs can vary greatly in the levels of detail and information that has to be provided. Beyond the zonal-type approaches, models developed specifically for a particular purpose (e.g. control design) exist. For model-based control design purposes, typically state-space models adhering to certain mathematical constraints are required. The accuracy of such models may not be of essence but rather their ability to correctly capture dynamics and sensitivities of the system that is being modeled, as this is the critical quality needed for control design. Still in some cases, looking at finer-than-zone scales can be justified. Such cases include: thermal comfort studies where the temperature distribution and its variations within a room or a zone must be known; or in ventilation studies, where the age of air can be an important parameter; or even for the determination of the placement of temperature sensors so that a good reading, representative of the average temperature in

the zone, can be ensured. In all these cases, the granularity offered by considering zonally averaged parameters is just not enough. It is exactly for these cases, that the use of Computational Fluid Dynamics (CFD) calculation methodologies can come handy. CFD for whole-building simulation can be helpful as a quasi-qualitative tool for understanding fundamental flow structures, but with little hope of matching real operation situations in great detail. It is for this reason, that the use of CFD calculation methodologies is mostly restricted in the design phases of a building life-cycle.

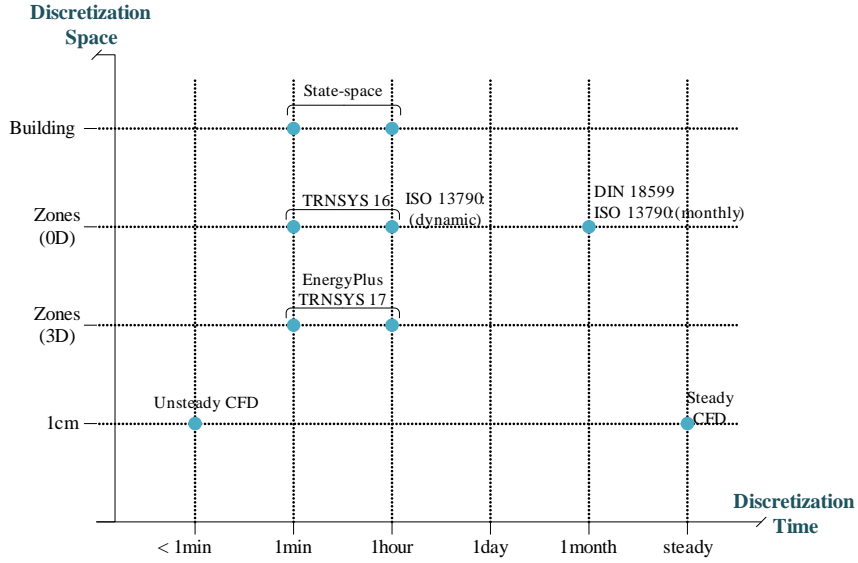


FIGURE 1.1: Amalgamation of discussion in this Section

Shown in Figure 1.1 are the calculation methodologies briefly discussed above, on a diagram with the spatial and temporal discretization on its two axes. The classification based on the spatial discretization is important as it determines the level of modeling detail and the amount of information that has to be prescribed as input, when defining the geometry and other related information. The temporal discretization dimension is also important as it determines the integration time step, and consequently the granularity in which dynamically changing data (occupancy, weather, etc.) should be defined.

In this thesis we intend to provide a more active role of simulation and such calculation methodologies in the building design and operation phases. Quasi-static and CFD calculation methodologies are primarily useful in the design phase, either due to the resolution of their predictions (annual basis for quasi-static) or due to the inherent assumptions and modeling detail required – as such, they are of lesser importance here. The use of time-steps in the range of a minute to one hour allows to account for the dynamics of active climate control systems, but also to incorporate control strategies that use state

measurements as inputs to compute actuation commands. The desire to use simulation as a forecasting tool, also suggests that a “small” time step might be warranted. In view of the comments above, the 3D zonal approximation is acceptable for many envisaged and practical use scenarios, as it manages to strike a balance between accuracy and computational complexity.

The accuracy of a 3D zonal-type BEP simulation result is determined by its input data, mainly comprising the building geometry, internal loads, HVAC systems and components, weather data, operating strategies and schedules, and simulation specific parameters. These data can be further categorized as follows: static or dynamic data. Static data include the building geometry, construction materials, glazing information, systems used in the building, etc., while dynamic data consist of all time-dependent data such as user-actions (e.g. opening and closing the windows), occupancy schedules in each of the building zones, use of equipment, weather predictions, etc., commonly being in-building sensed measurements.

In current practice, to develop a 3D zonal-type BEP simulation model, modelers gather and combine 2D drawings such as Architectural and Mechanical Electrical Plumbing (MEP) plan views, material data and other information, and manually transform them into the specific input data, required by the respective BEP simulation engine. As Figure 1.2 depicts, BEP simulation model preparation consists of the following steps:

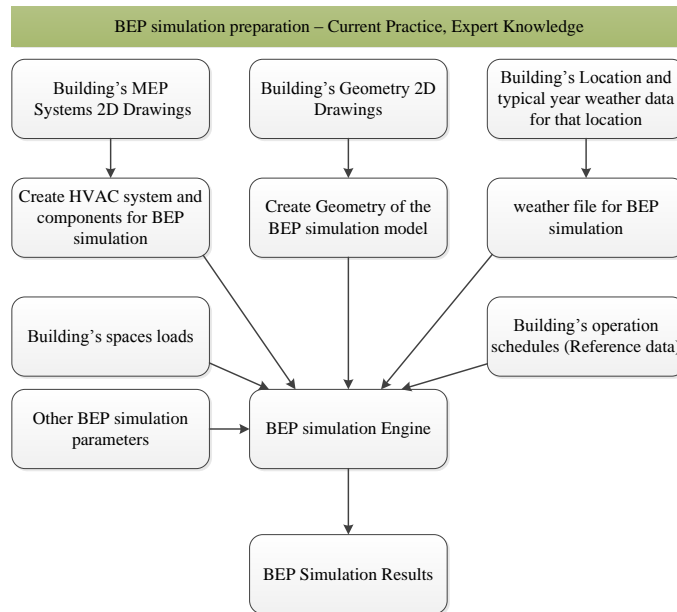


FIGURE 1.2: BEP simulation preparation – Current Practice, Expert Knowledge

1. Determination of the Building's location and typical year weather data for that location (dynamic data);
2. Definition of the building's geometry, constructions and spaces according to 2D architectural drawings (static data);
3. Definition of space loads — like electric equipment, lighting, etc.(static data);
4. Definition of the HVAC system and its components according to 2D Mechanical Electrical Pumping (MEP) drawings (static data);
5. Determination of simulation other parameters — such as numerical tolerances, begin and end time of the simulation (dynamic data);
6. Determination of reference data for the building's operation schedules (dynamic data).

This process has two strong weaknesses: a) it is very time-consuming, often requiring more time than is available due to project's deadlines; and b) it is a non-standardized process that produces BEP simulation models whose results can significantly vary from one modeler to another according to their experience [12], even given the same initial building design information. A methodology for semi-automated creation of BEP simulation models could make the BEP simulation modeling process much more expedient and therefore, less vulnerable to modeling errors. Building Information Models (BIMs) are information-rich repositories that could be used to streamline and expedite the collection of such information. Hence, current research studies focus on BIM-based BEP simulation models generation (see Figure 1.3).

BIM is an object-oriented digital representation of a building, which encapsulates the information required for generating a BEP simulation model. Relevant BIM data schemas include the Industry Foundation Classes (IFC) [39] and the green-building XML schema (gbXML).

The current version of IFC (IFC4) is an ISO standard and is the replacement of the previous IFC2X3 version. Its data are physically stored in a STEP file (readable with a text editor) using EXPRESS data definition language. In addition to the IFC-EXPRESS specification an ifcXML specification is published as well, following the XML document structure. The IFC-file usually is generated by the exporter of a BIM-authoring tool. Actually there is no software available implementing an IFC4 export function besides a new tool called “Constructivity”, but still with limited IFC4 support.

The BIM server facilitates the storage of and access to IFC files in a database and furthermore the data management including version control and multi-user support.

The “BiMserver project” provides an online access to their demosever as well as the download of local servers. The server is based on Java-code and open source. The BIM server also provides simple queries of the building model, i.e. target selected information out of the model content. The query language is Java. It is also possible to query the model with a stand alone tool bypassing the user interface of the BIM server. This can be realized by Java code (environment e.g. Eclipse), using the libraries of the BiMserver. The code than gets access to the BiMserver, loads (deserializes) an IFC file into an object variable and can be used to define different logic for certain data request. The use of the BIM libraries and the data model enables a comfortable handling of the data but is limited to the state of completeness of the implantation regarding server and the IFC version. Going one step back, the queries can also be realized using direct access to the IFC file without having a BIM server (or libraries) involved. Since IFC is using the STEP schema, there are several tools available for parsing STEP files (direct access to text/STEP file).

Within the buildingSMART project, a new specification of schema was introduced called Model View Definition (MVD). Since the IFC model contains a significant amount of information covering most functional requirements during the lifetime of a building, a “sorting” process has to proceed, where all the information not required by the simulation model definition task will be excluded from the interface definition. Model View Definitions (MVD) could be developed towards this way. An MVD can be described as an intermediate state of data structure in-between the original IFC file and the result of a certain query. The physical format of the MVD could still be an IFC file or an mvdXML file. The main task of an MVD is to filter the IFC data in order to reduce the amount of information and focus on data related to a defined subject, i.e. structural data, HVAC data, etc. The additional important role of MVD is allowing consistency checks of the data, exception handling and filling of “gaps” in data (as far as possible). MVDs are defined and published using the “ifcDoc” tool. For instance, a MVD that filters the IFC’s building geometry data could be defined. Although IFC files contain information referring to multiple building geometry entities, only some of them are required for building thermal simulations. These building geometry entities can be classified into three categories depending on their role in thermal building simulations: Building Elements, Openings, and Volumes, which are described by objects of *IfcBuildingElement*, *IfcOpeningElement*, *IfcSpace*, and *textitIfcSite* IFC classes.

IFC supports three file formats that facilitate exchange between applications: 1) .ifc – the default IFC exchange format; 2) .ifcXML – using the XML document structure; and 3) .ifcZIP – using the PKzip compression algorithm. GbXML is based on the XML (Extensible Markup Language) format, deployed by Green Building Studio (GBS),

that can potentially represent any building model through translation using appropriate mapping engines.

Many commercial authoring tools (e.g. Revit<sup>TM</sup>, Vasari<sup>TM</sup> and ArchiCad<sup>TM</sup>) support export in one of these two BIM schemas. One of the problems is that this export is often not perfect: unlike what would be expected, the resulting exports are of poor quality and therefore not directly usable.

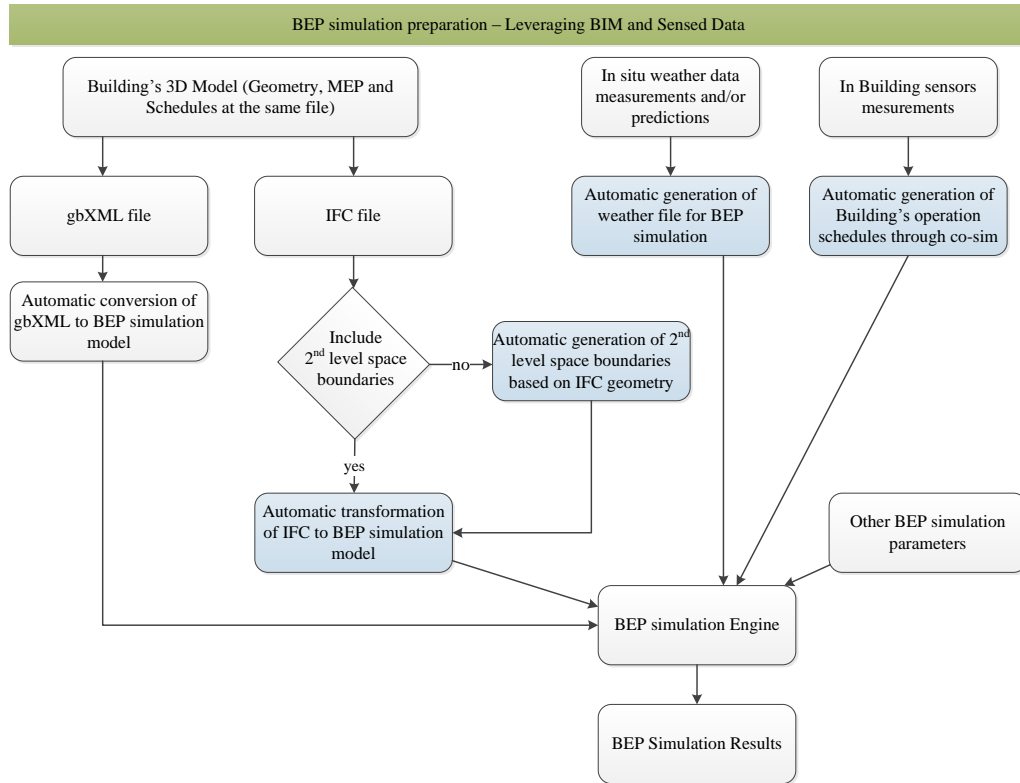


FIGURE 1.3: BEP simulation preparation – Leveraging BIM and Sensed Data

Concerning building geometry, the generic approach of IFC has the ability to represent any shape of geometry, while gbXML only accepts the rectangular shapes. For this reason, a plethora of recent studies focuses on developing a methodology to automatically generate geometry inputs for BEP simulation from IFC files. IFC appears to be a suitable choice as its rich content enables interoperability among different software environments and can easily be updated following a building's life cycle [32]. Although IFC has the necessary classes to support the description of building geometry in a very compact and precise way, it is quite common that the 2<sup>nd</sup>-level space boundary data (see Section 2.3.1.1), necessary for energy simulations, contained in IFC, are missing or are incorrect due to design errors or exporting software imperfections. Hence, a consistent way is

required of transforming the building geometry information contained in the IFC, into 2<sup>nd</sup>-level space boundary information.

For a 2<sup>nd</sup>-level boundary generation process, the input geometry should be of good quality; however, IFC geometry description classes sometimes contain incorrect data, mainly induced due to two reasons: either the designer has erroneously defined certain building entities, or the IFC-exporter software has flaws and exports incorrectly the geometrical data in the IFC file. Such errors can slow or make that process fail. In [73], errors that affect the creation of properly defined 2<sup>nd</sup>-level space boundaries are presented. Consequently, as a post-processing step, after the IFC file export, such errors should be detected and corrected either automatically or manually by communicating them back to the designer. Commercial software, such as Solibri Model Checker [48], are able to identify such errors, which are communicated back to the BIM software and corrected manually. Towards automatic detection and correction of such inaccuracies, detection and correction algorithms have been developed, though their description is not the topic of this thesis. With an IFC free of design errors and building space incorrect definitions at hand, geometric data can be queried by the 2<sup>nd</sup>-level space boundary generation process. When 2<sup>nd</sup>-level space boundary information is available, a data mapping process of elements between the IFC and the BEP simulation tool is required to eventually generate the BEP simulation model's geometry.

Regarding the rest BEP simulation input data, the space loads are contained in the IFC file and can be translated to the BEP simulation space loads. The HVAC system and its relevant components can be defined either by manual user input or semi-automatically by establishing a data mapping process of the HVAC elements between the IFC and the BEP simulation input data. Moreover, IFC has the proper classes to carry building sensing information; the availability of sensor data in a building presents an opportunity to eliminate with minimal effort the uncertainties of the BEP model. In current practice, due to imperfections of current BIM-authoring tools' exporters, such information is incorrect or missing, while objects of these classes are usually related to another external information layer; hence, instead of sensed, reference data are used. To overcome this drawback, sensed data can be forwarded to BEP simulation model through a co-simulation setup (see Section 2.5). With respect to the weather data availability, IFC does not contain any actual information on weather conditions, such as temperature, solar radiation, wind direction, etc. These data are derived from external data sources and can be used to automatically generate the specifically formatted weather file (e.g. EPW format file for EnergyPlus simulation).

Concluding, following the process illustrated in Figure 1.3, a BEP simulation model can be generated leveraging the BIM and building's sensed data. However, the models derived from this process are elaborate and computationally expensive; a recurring theme that the modeler has to address during the BEP simulation model generation is the decision on how accurate the model should be, the trade-off being between accuracy and computational complexity. For Control-Design tasks, the accuracy requirements are markedly different compared to the simulation model for energy auditing purposes. Control Design processes require a model that is able to capture the sensitivities and trends but no accuracy is necessary, whereas for energy auditing all modeling assumptions have to be appropriate so that the simulated building matches as realistically as possible the real one. In the trade-off between accuracy and computational efficiency the two models lie on opposite edges of the spectrum. For model-based Control-Design purposes, typically state-space models adhering to certain mathematical constraints are required (see Figure 1.4).

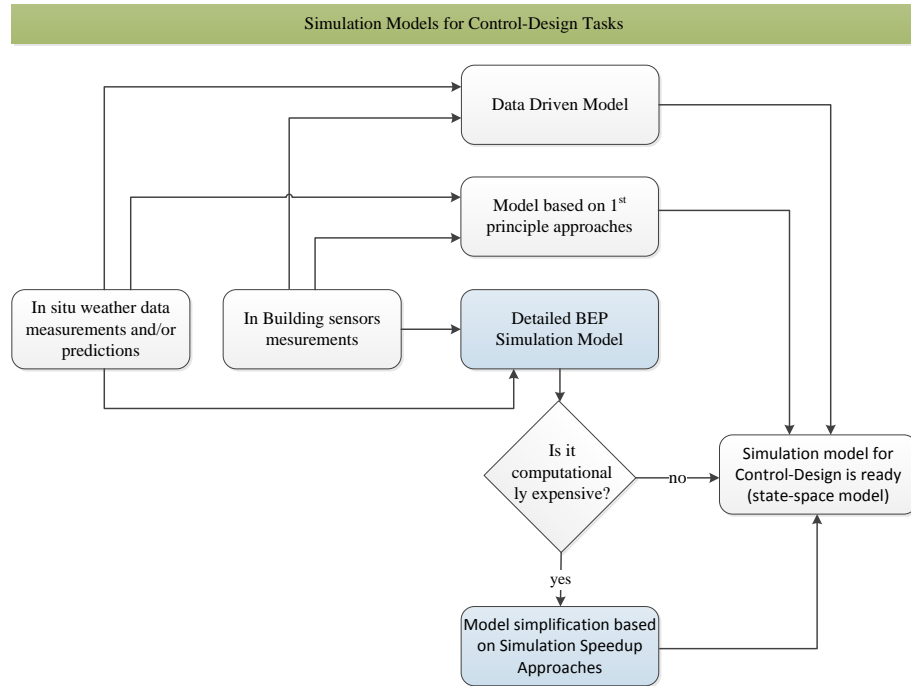


FIGURE 1.4: Generating Simulation Models for Control-Design Tasks

Towards defining such models many approaches are possible. These models can be developed using first principle approaches but for larger buildings their construction is impractical. Data-driven models produced by system identification methods fails when applied to real, occupied buildings, due to under-excitation of system dynamics [100]. In more recent approaches [84] a full scale (detailed), zonal-type, thermal simulation

model of the building is used for the identification phase, which remains computationally expensive. What is particularly attractive in this approach is that the excitation necessary for the identification happens at the simulation level, so it is possible to excite the system in many ways that would be impractical, or even unrealistic, if they were to be applied to the real building. Even though for small buildings, full scale models have been used successfully for model-assisted control design, for large buildings a very detailed model is impractical to simulate, since it is too complex to build and takes too long to run. Simulation is predominantly slowed down by the increasing number of zones, windows and surfaces, and, therefore, simulation speedup techniques must focus on geometry reduction, zoning reduction and/or order reduction approaches.

## 1.1 Objectives

Focusing on the building's envelope thermal simulation, this thesis aims at developing a methodology which will automatically generate Building Energy Performance (BEP) simulation models, in order to be used for either detailed thermal auditing purposes or Control-Design tasks using available data from Building Information Models (BIMs).

Concerning the building geometry, the Industry Foundation Classes (IFC) schema is adopted, providing static building information that includes geometric configuration and material properties, but in a form that might not be directly usable for the generation of thermal simulation models due to the absence of 2<sup>nd</sup>-level boundary information. Hence, in this thesis a 2<sup>nd</sup>-level space boundary generation methodology is presented, applied to generate BEP models of high detail (also noted as full-scale BEP models), appropriate for energy auditing purposes. A transformation process that converts the 2<sup>nd</sup>-level space boundary into a BEP simulation engine's input file is also required; such transformation processes are presented for the EnergyPlus and the TRNSYS simulation engines.

During the BEP simulation preparation for Control-Design tasks, the main objective would be to significantly decrease the simulation run-time of a model derived from the aforementioned methodology. Towards this way, simulation speed-up approaches are presented and their efficiency is investigated.

For either energy auditing or control-design tasks, the availability of sensor data in a building presents an opportunity to eliminate with minimal effort the uncertainties of the BEP model. Establishing a link between the BEP simulation model and the building's sensed measurements, so that the dynamic schedules can be incorporated in

the simulation, is of paramount importance. This functionality, which we refer to as co-simulation, is especially desirable also for the testing and design of control strategies and is presented in this thesis.

Moreover, building's envelope thermal simulation calculations require weather data values in order to be executed. Most of these data are provided by weather files. However, it is quite common that weather data do not include information concerning solar radiation, but only a percentage of cloud coverage. Even if global radiation data are available, a crucial input in the simulation of building's energy performance is the availability of both diffuse and direct radiation data. Nevertheless, in most cases, measured data of diffuse and direct radiation are not available. Among a plethora of methods for estimating the diffuse (or direct) fraction of global solar radiation, the question of the best method is not fully settled, a task that this thesis tries to address.

According to the previous discussion, in this thesis the following goals are defined:

1. Query on the building data model requesting geometry-related information and processing of the acquired data by a 2<sup>nd</sup>-level boundary identification algorithm, the Common Boundary Intersection Projection (CBIP) algorithm, to generate geometry information required for the development of a BEP simulation model;
2. Development of a transformation process that converts the geometry information of IFC, along with the data obtained from the CBIP algorithm, to an EnergyPlus and/or TRNSYS input file — widely used BEP simulation engines — for detailed thermal auditing purposes;
3. Development of simulation speed-up approaches towards building thermal simulation models for Control-Design tasks;
4. Establishment of a co-simulation methodology that performs data-exchange whenever required; and finally,
5. Regarding the solar radiation data, the different established models and correlations that calculate hourly solar radiation components are selected, performed and tested to decide which model is recommended.

## 1.2 Outline

The multi-step methodology to extract IFC geometry inputs and transform to inputs appropriate for a BEP simulation model — for either energy auditing or control design

purposes — consists of different components, schematically shown in Figure 1.5 and explained in more detail in the following Chapters.

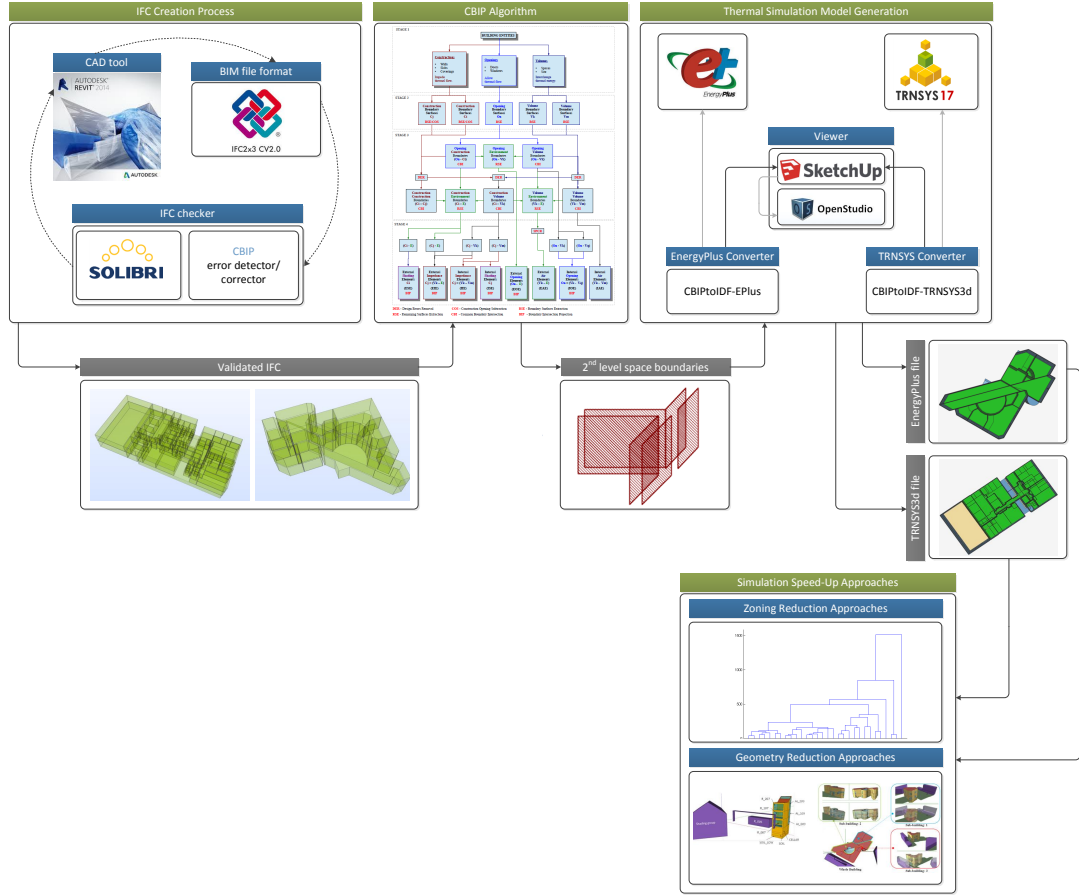


FIGURE 1.5: Static Data – Transformation of an IFC file's geometry to a BEP simulation model's geometry for energy auditing or control design purposes

In Chapter 2, we review some of the earlier work related to the topics of this thesis.

In Chapter 3, a four-step methodology for automated generation of 2<sup>nd</sup>-level space boundary information is presented (CBIP algorithm), including: the Identification (ID) stage, the Boundary Surface Extraction (BSE) stage, the Common Boundary Intersection (CBI) stage and the Boundary Intersection Projection (BIP) stage.

For a BEP model generation, a transformation process that converts the data obtained from the CBIP algorithm to a BEP simulation engine's input file is required; rules embedded in the transformation process are presented in Chapter 4.

In Chapter 5, BEP simulation speed-up approaches, required for Control-Design tasks, are presented and categorized to geometry and zoning reduction approaches. In zoning reduction approaches, various candidate zoning approximations are evaluated; hence,

multiple zonal-type thermal simulation models need to be generated. Their generation can be performed automatically, and as such an automatic process for generating speed-up models based on zoning reduction approaches is adopted.

The methodology to establish a link between the BEP simulation model and the building's sensed measurements, so that the dynamic schedules can be incorporated in the simulation, is depicted in Figure 1.6.

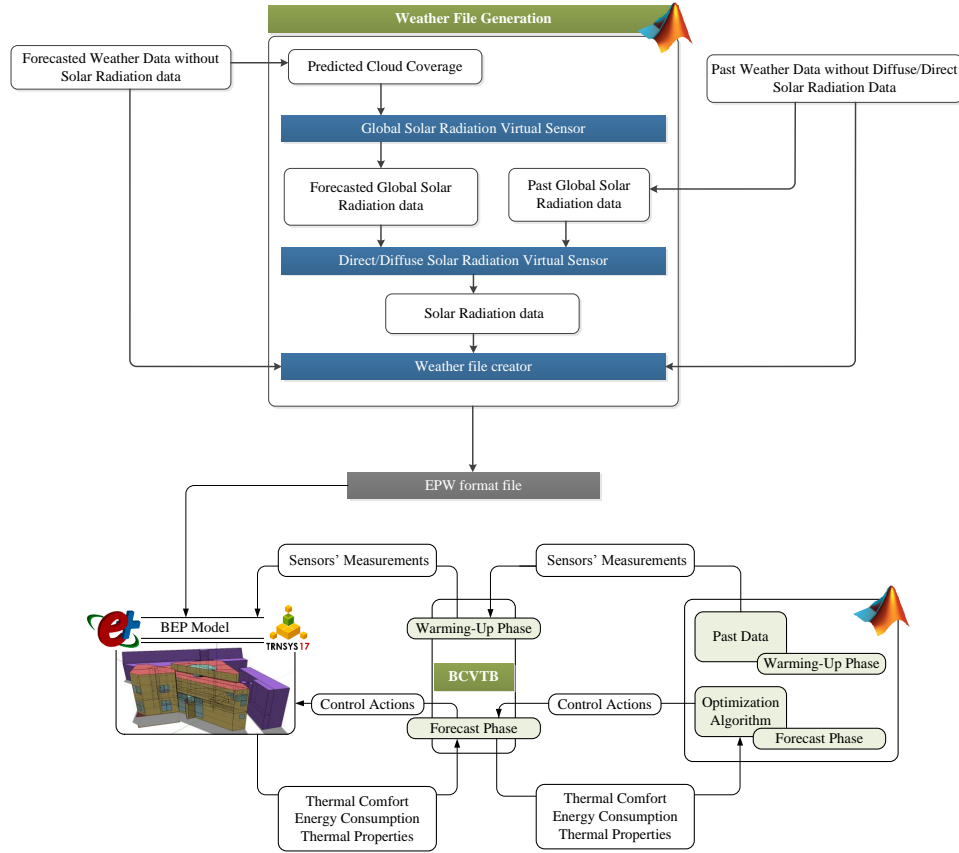


FIGURE 1.6: Dynamic Data – Improve the BEP simulation model utilizing in situ weather data (and/or weather predictions and a co-simulation set-up)

In Chapter 6, the co-simulation methodology for data-exchange is described. Moreover, models that are used to estimate the global solar radiation from existing percentage of cloud coverage data and diffuse/direct solar radiation components from existing global solar radiation data, are presented, while their impact to a BEP Simulation is investigated.

Finally, in Chapter 7, conclusions and future work are discussed.

### 1.2.1 Buildings used for Testing and Application of Proposed Methodologies

To investigate the applicability of the presented methodologies on existing buildings, three experimental buildings are selected and depicted in Figure 1.7.



FIGURE 1.7: Demonstration Buildings of the proposed methodologies

The CARTIF building has a rectangular shape with 3 floors (basement, ground floor and first floor). In the basement there are service spaces, like electrical controls and storage rooms. Boiler room and thermal-solar facilities are situated in two rooms close to the garage and exterior. All spaces are nor heated nor cooled. In ground floor there are two main zones, offices and conference rooms are situated on the north side (administrative area). Research laboratories and the warehouse are located in the southern side. On first floor there are research laboratories and head's offices. Energy Division laboratory is placed on South-East side.

The CARTIF building is chosen to present the applicability of CBIP algorithm, the accomplished “2<sup>nd</sup>-level space boundaries to BEP simulation model” transformation process, and to investigate the efficiency of zoning reduction approaches.

The TUC building is of triangular shape and comprises 10 office rooms, an open meeting space, two corridors (one in each floor), the main entrance, an equipment room, a toilet and a basement that is used as a storage area. In both floors (ground floor and first floor) there is a central corridor running the length of the building with offices on either side. In the middle of the corridor there is an open meeting space of semicircular form. The plan of the first floor is similar to the ground floor, with the only difference the presence in the first floor of a semicircular atrium to connect the ground floor meeting space with a large glazing on the roof.

The TUC building is used as a demonstration example of CBIP algorithm, while investigating the efficiency of geometry reduction approaches in Chapter 5, a validated thermal simulation model has been developed in EnergyPlus. The representation of the building geometry was created according to the floor plans using the Google SketchUp

plugin, Openstudio – see Figure 1.8. Moreover, TUC building is selected to analyze the co-simulation setup in Chapter 6.

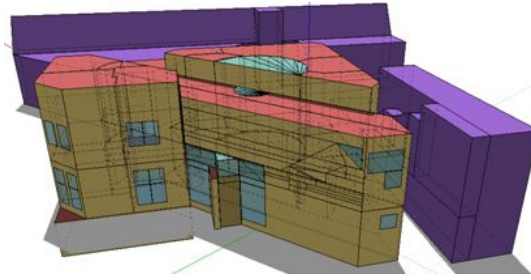


FIGURE 1.8: TUC building – full-scale simulation model

The FIBP building is the office building of the Centre for Sustainable Building in Kassel, situated at the University of Kassel, Germany. The building consists of 26 rooms, spanned in three floors and each physical room has at least one corresponding thermal zone in the full scale model.

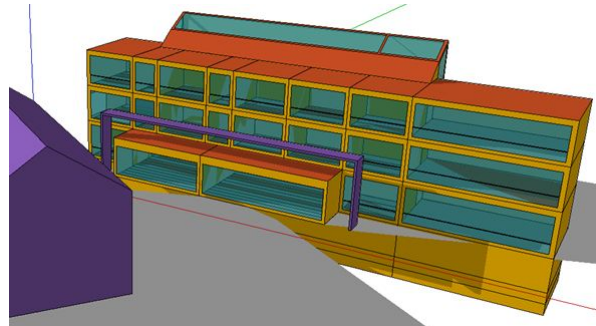


FIGURE 1.9: FIBP building – full-scale simulation model

The full-scale model of the FIBP building has been set up in Trnsys 17 using Google SketchUp with the purpose to prove the effectiveness of a geometry reduction approach.

For a detailed description of the FIBP and TUC buildings' full-scale thermal simulation models, refer to [23, 37], while the CARTIF building is described thoroughly in [37].

## Chapter 2

# Background

For a critical review of other attempts that have been done towards addressing the goals of this thesis, in this Chapter, initially the BEP simulation methodologies and the widely used BEP simulation engines are presented. Moreover, a categorization of BEP simulation data requirements to static and dynamic is discussed, and previous studies focusing on the BIM to BEP simulation data transformation are described. Finally, methodologies are introduced establishing a link between static and dynamic data.

### 2.1 BEP Simulation Methodologies

In general, model-based building thermal and energy simulation programs use mass and energy balances [69] as a basis for estimation of the evolution of the values of parameters referring to internal conditions (temperature, humidity, CO<sub>2</sub> concentration, luminance) and energy needs (total energy, maximum power demands) concerning building interiors. Energy conservation laws are used to investigate thermal energy transfers and exchanges among building elements, spaces, and systems, while mass conservation is used for evaluation of vapour-water transfers (humidity). Implicit in all methodologies is the discretization of the pertinent conservation equations over pre-determined time intervals. Based on time resolution criteria, calculation methodologies can be classified into two categories:

- Static or quasi-static calculation methodologies. These methods assume average parameter values for a long period of time (typically a month or a season), and account for dynamic effects using empirical correlations and averaging correction factors. These types of calculation methods are especially useful for estimation on energy performance on an annual basis.

- Transient calculation methodologies. Transient calculation methodologies take a more granular approach using a time resolution which is comparable to the time scale of time-varying effects that are being modeled. Consequently, these methods are capable of capturing transient phenomena such as weather changes, occupancy variations, thermal loading effects, or the effects of Building Energy Management (BEM) systems.

The monthly-based calculation methodology described in ISO 13790:2008 [38] is a prime example of a quasi-static calculation methodology. This fully-prescribed calculation methodology has been adapted – in the context of activities for the implementation of the EPBD [65] – by many EU member states to form at a national level an accepted calculation methodology for computing energy performance. Annex H of the standard discusses the accuracy of the calculation methodology and the sensitivity to errors in the input data. Under certain conditions, the calculation methodology can be validated against reality and relatively small deviations can be observed for annual predictions, but on a monthly scale these deviations can be significant. The sensitivity to input data is also discussed: uncertainties in the estimation of thermal properties or other input parameters can contaminate the results, and the propagation of these errors can yield sizable deviations in the end results. For this reason, in many cases, the calculation methodology is used to establish an ordering relation, that allows for meaningful comparisons of different retrofitting scenarios (and thus establishing the rating system used in many countries), but with lesser expectations with regards to prognosis of real performance.

### 2.1.1 Quasi-static Calculation Methodologies

A basic modeling assumption used here is the multi-zonal paradigm: dividing the volume of the building into disjoint regions (zones), each with the basic variables (say, temperature) assumed to be spatially constant. The evolution in time of the zonal parameters is evaluated from the solution of a system of algebraic and/or ordinary differential equations. Zones in that sense form the basic spatial component for performing the calculations. The selection of zoning is in essence the spatial discretization of the building. As it is typical in other domains (e.g. the numerical solution of partial differential equations), the resolution should be granular enough to be able to discern all basic effects that are being modeled. So, in effect, the calculation methodology, and in particular the level of modeling detail employed by that methodology, govern the spatial discretization to be used. Implicit in the choice of discretization is the balance between accuracy and complexity. An overly fine discretization can lead to many input requirements and increase disproportionately the effort required in setting up the calculation. Too coarse

of an approximation implies low complexity but also large approximation errors might be introduced invalidating the obtained results. Obviously selecting the zoning to strike a proper balance between accuracy and complexity is a critical consideration in setting up the simulation, and it is something that cannot be done automatically. In the calculations described in the standards, the desire for transparency and reproducibility push toward coarser zoning definitions. But for more detailed modeling and realistic results, zoning is the single most important assumption that can separate a properly conducted simulation that closely reflects reality from a nonsensical one. The multitude of ways that the spatial discretization can be defined, with the concomitant effects it has with respect to the quality of the simulation, and the ambiguity in its definition for complex building geometries, makes the entering threshold for whole-building simulation quite high. Once the spatial discretization has been established, at a second level the interactions between zones need to be prescribed: in most cases inter-zone exchanges should be enforced, but to simplify the calculation in certain cases (when there is presumed weak thermal coupling) adiabatic boundary conditions can be enforced thus simplifying the calculation. Once the spatial discretization has been established a connectivity graph can be created to represent the interactions between zones.

In Figure 2.1, a schematic of the calculation methodology is illustrated for a building split into three zones. Quoting the Standard [38], once the zoning has been established, the basic energy interactions that have to be accounted for in forming the energy (heat) balance at the building zone level include the following terms:

- transmission heat transfer between the conditioned space and the external environment, governed by the difference between the temperature of the conditioned zone and the ambient temperature;
- ventilation heat transfer (by natural ventilation or by a mechanical ventilation system), governed by the difference between the temperature of the conditioned zone and the supply air temperature;
- transmission and ventilation heat transfer between adjacent zones, governed by the difference between the temperature of the conditioned zone and the internal temperature in the adjacent space;
- internal heat gains (including negative gains from heat sinks), for instance from persons, appliances, lighting and heat dissipated in, or absorbed by, heating, cooling, hot water or ventilation systems;
- solar heat gains (which can be direct, e.g. through windows, or indirect, e.g. via absorption in opaque building elements);

- storage of heat in, or release of stored heat from, the mass of the building;
- energy need for heating: if the zone is heated, a heating system supplies heat in order to raise the internal temperature to the required minimum level (the set-point for heating);
- energy need for cooling: if the zone is cooled, a cooling system extracts heat in order to lower the internal temperature to the required maximum level (the set-point for cooling).

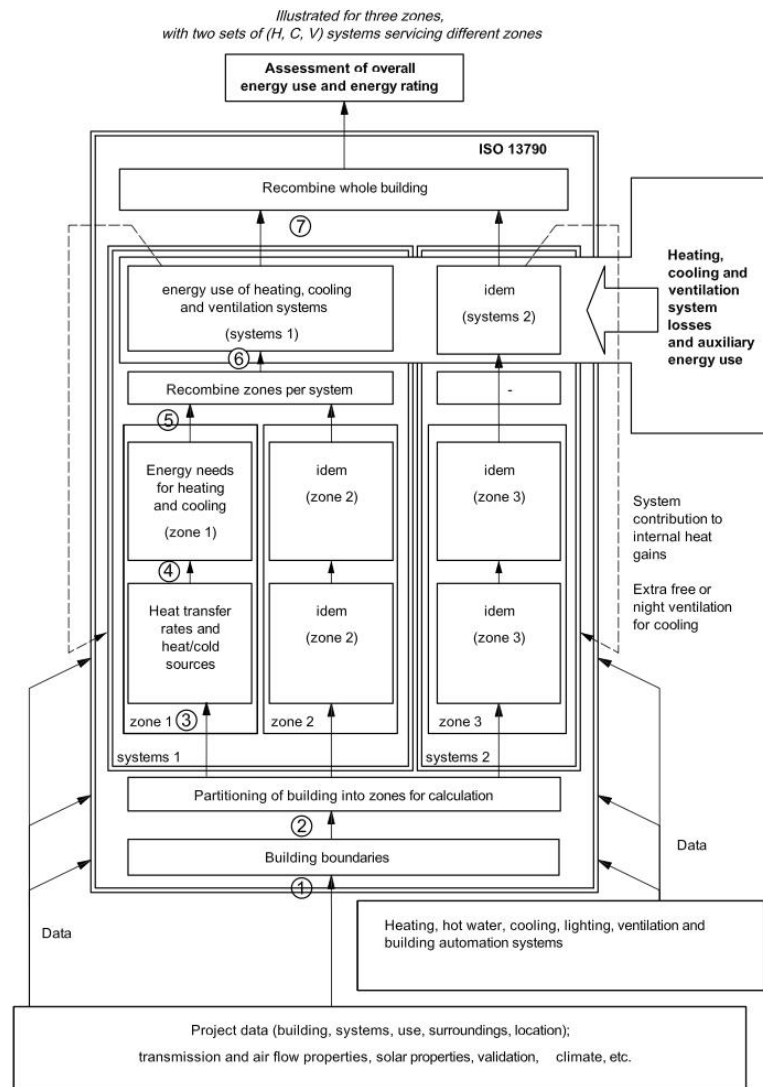


FIGURE 2.1: Schematic of the ISO 13790:2008 calculation methodology [38]

The basic energy calculations involving the terms described above are performed at each zone and then combined to estimate the energy use for heating, cooling and ventilation systems. The whole process is repeated in an iterative manner. Upon recombination from the zones to the whole building energy-use, indices are computed for the whole

building. Boundaries to the calculation are the presence of systems for heating, hot water, cooling, lighting, ventilation and building automation systems. Their presence and concomitant calculation methodologies modeling their effects are stipulated in other standards and should be used together with the building model.

### 2.1.2 Transient Calculation Methodologies

In the case where dynamic effects are important, the temporal resolution of a month is not sufficient to capture all relevant dynamics. In this case, smaller time steps are required and a different approach is essential. This has obvious benefits: certain physical effects, like transfer of heat from building thermal masses or the dynamic effects of the operation of active climate control HVAC components, happen on a time-scale which is comparable to the simulation time-step. It is then possible to use more detailed models that capture these dynamic interactions and there is no longer the need for averaging or the use of correlations and other correction factors. On the other hand, the need for defining boundary conditions, at each time step means that in many cases the problem definition has to be more detailed (at each time step) requiring, at this level of detail, information which may not be available.

An example of such model is the dynamic model described in [38]. Here a zone is represented as a thermal circuit with 5 thermal resistances and 1 thermal capacitance (5R1C).

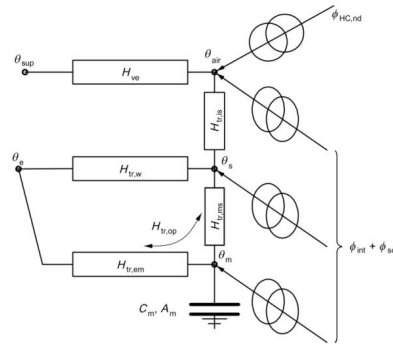


FIGURE 2.2: 5R1C Representation of a zone [38]

The thermal capacitance models thermal storage effects in the zone. The mathematical formulation of the problem in this case is an ordinary differential equation which models the evolution of the temperature as a function of time. Upon discretization of the equation using a finite-difference scheme, e.g. an implicit scheme like the Crank-Nicolson method, one gets the equations for the evolution in time of the relevant temperature

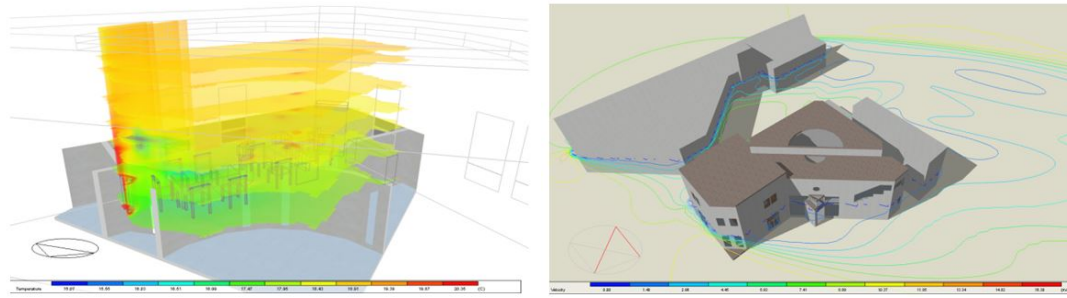
parameter. One obvious benefit in the model above is the ability to model the temperature in the walls and therefore it becomes possible to have estimates of thermal comfort (as the radiant temperature is an important parameter for thermal comfort). In a multi-zone configuration one needs to set individually for each of these nodes the thermal system and combine it to form the overall thermal network. The number of capacitances in this case is proportional to the number of zones, and a system of Ordinary Differential Equations has to be integrated in time. In [38], such a methodology is described and the boundary conditions are selected to ensure compatibility between the monthly and dynamic models.

A similar approach is followed in TRNSYS and EnergyPlus. TRNSYS has a modular and extensible structure where different models for the building and its systems (called Types in TRNSYS lingo) are combined to form the problem description. Type 56, implements the multi-zone building model. There the geometrical and connectivity information for the zonal splitting is provided along with parameters for describing opaque and transparent building materials. The models used for the multi-zone building are more detailed than the simple RC above, including a star-shaped topology for approximating radiant exchange between zone surfaces along with the transfer function method for modelling transient conductive exchanges through walls. The integration time step can vary from 1min to 1h. This higher level of simulation detail is especially useful when one considers the coupled interaction of the building and energy systems; for this reason it has been extensively used as a simulation-aid tool for energy systems development and testing.

In the previous calculation methodologies only sensible heat calculations were described. It is also possible to include latent heat into the calculation methodology: this is particularly desirable in the presence of humidification and dehumidification systems. In these cases, the air of building spaces is considered a mixture of dry air and water in the vapour state [4]. The amount of water vapour present building spaces' air affect the temporal thermal heat storing capability of the air and the heat transfer rate between the air and neighbour building elements [5]. Therefore, to specify the percentage of water vapour in building spaces' air, vapour transfers between the outside air or adjacent air volumes and the air volume of the space under consideration have to be accounted at every simulation time instant, by augmenting the system of energy conservation equations with mass conservation equations for the moisture content in the air.

In some cases, looking at finer-than-zone scales can be justified. Such cases include: thermal comfort studies where the temperature distribution and its variations within a room or a zone must be known; or in ventilation studies, where the age of air can be

an important parameter; or even for the determination of the placement of temperature sensors so that a good reading, representative of the average temperature in the zone, can be ensured. In all these cases, the granularity offered by considering zonally averaged parameters is just not enough. It is exactly for these cases, that the use of Computational Fluid Dynamics (CFD) calculation methodologies can come handy. Shown in Figure 2.3, are two examples, of the implementation of such methodologies for the TUC building [80]. Shown in the left (Figure 2.3(a)), is the temperature distribution on one of the rooms during winter heating mode, to estimate the homogeneity of the temperature fields due to heating from the radiators. A second example is shown in Figure 2.3(b) (right); here, an external CFD calculation was performed to find pressure and velocity fields, developed in the building due to the presence of winds and the interactions with nearby structures. The information shown, velocity field on a plane parallel to the ground, is especially useful for estimating the pressure coefficient on windows, so that better modeling of infiltration can be achieved.



(a) Temperature distribution in a zone during heating (b) External CFD calculation to determine the pressure and velocity fields

FIGURE 2.3: CFD calculations for the TUC building [80]

The topic of CFD is very mature and has been developed, applied, and extensively validated in many fields where the dynamic behaviour of fluids (e.g. air) needs to be computed. In these methodologies, first the solution space is defined and appropriate initial and boundary conditions are defined. Then a space-filling partition is introduced using pyramidal or hexahedral elements. The size of these elements should be smaller than a characteristic length scale related to the size of the flow structures that have to be resolved. This partition is often defined in a conforming manner, where the computational grid is defined in a conformant to the boundaries fashion. Then the conservation laws are stated: typically the Navier-Stokes equations for mass and momentum conservation and the energy conservation equation. A discretization methodology, like the finite-volume or finite-element method is then used to discretize the partial differential equations, on volumes or elements defined in the partitioned space. As a result of the discretization a large system of equations, is solved numerically to yield approximations to the temperature, pressure and velocity fields.

The biggest problem for CFD implementation as whole building calculation methodologies is the need for boundary conditions, to be prescribed on all boundaries of the computation domain. An approach which is often used is to first use a zonal-type approximation, which is seen as a cruder first step, to create the boundary conditions, and then use these boundary conditions to pose and solve the CFD problem. This can be problematic for complex geometries, as a data transformation process has to occur between different domains. As can be readily inferred the pollution due to uncertainties in the results of the zonal approximation, is propagated in the CFD calculations. Unless great care is taken in performing the transfer of information between the two approaches, the validity of the obtained results is always subject to scrutiny. In that sense, CFD for whole-building simulation can be helpful as a quasi-qualitative tool for understanding fundamental flow structures, but with little hope of matching real operation situations in great detail. It is for this reason, that the use of CFD calculation methodologies is mostly restricted in the design phases of a building life-cycle.

Hence, the 3D zonal approximation (followed by EnergyPlus and TRNSYS17 software) is acceptable for many envisaged and practical use scenarios, as it manages to strike a balance between accuracy and the errors introduced by uncertainties (values for thermal properties, occupancy and operation schedules, user actions, etc.). In this thesis, 3D zonal-type BEP simulation methodologies are investigated, and from this point BEP simulation will refer to this category of methodologies.

## 2.2 BEP Simulation Engines

Concerning the 3D zonal-type transient methodologies, currently numerous BEP simulation engines exist. The most popular between them are: BLAST (Building Loads Analysis and System Thermodynamics); BSim (Danish Building Research Institute); DeST (Designer's Simulation Toolkits); DOE-2.1E (Department of Energy); ECOTECH; Ener-Win; Energy Express; Energy-10; EnergyPlus; eQUEST; ESP-r; HAP (Hourly Analysis Program); HEED; IDA ICE (Indoor Climate and Energy); IES<VE> (<Vrtual Environment>); PowerDomus; SUNREL; Tas; TRACE (Trane Air Conditioning Economics); TRNSYS(Transient Systems Simulation); and Modelica.

The accuracy of the simulation results strongly depends on the calculation engine used. A relevant standard for calculation engine validation is the ANSI/ASHRAE Standard 140-2007 Standard Method of Test for the Evaluation of Building Energy Analysis Computer Programs [1]; here, a set of synthetic benchmarks is defined (Cases) ordered in order of modelling complexity. The goal of this standard is to provide a standardized methodology for testing and debugging building energy analysis methodologies. As part

of the standard, a well-defined testing procedure is established: if a calculation methodology fails a case, a number of diagnostic subcases are defined to help identify the root cause of the failure. Also, unlike other standards, there are no fixed accuracy limits to determine that a calculation methodology has “passed” a case, rather a comparative approach is recommended, in which the results of the calculation methodology are compared against other state-of-the-art tools. This comparative methodology serves two purposes: first, to help diagnose modeling and coding errors; and second, to compare against other state-of-the-art approaches so that output variability due to different modeling approaches can be better understood. In many cases empirical validation studies, have been conducted against calculation engines, strengthening the confidence in the capabilities to correctly mirror reality. It should be emphasized that once a calculation methodology has passed all tests of a validation procedure, and is deemed “validated”, this in no cases does it imply that the calculation methodology represents the truth. It does show that a set of algorithms have shown, through a repeatable procedure, to perform according to the state-of-the-art.

Different studies for different BEP simulation related tasks contrast the capabilities of existing BEP simulation engines. Crawley et al. [17], pioneers of such studies, have detailed the functionality and differences of twenty major building simulation tools. In [61] an energy performance comparison methodology to identify performance problems from a comparison of measured and simulated energy performance data is presented, and eight different simulation engines are evaluated for their capability to be used for that task. The simulation engines selection is based in their ability to contain more than the average number of HVAC components and system types. Eventually, EnergyPlus is reported as the most suitable simulation engine, since none of the other tools incorporates two of our requirements: the ability to create partial geometry models from IFC-based BIM geometry and/or the ability to directly link to control design tools. In [6] the review focuses on tools that can be used at multiple stages of the life-cycle and that provide functionalities to exchange data with other tools in open standard building information models, the IFC and gbXML. Concerning the optimization in BEP simulation, in [70], the intensity of utilization of twenty widely used building simulation programs [17] is investigated, concluding to the results presented in Figure 2.4. Optimization in BEP simulation could be performed in model-assisted control design tasks and/or to improve building performance through a method known as parametric analysis. The investigation is based on a search performed on Scopus (abstract and citation database) for the period 2000–2013, using the following keywords: name of a program; optimization; and building. There, EnergyPlus and TRNSYS seem to be the most frequently used tools, in all probability, due to their text-based format of inputs and output that facilitates the coupling with optimization algorithms and, their strong capabilities as well.

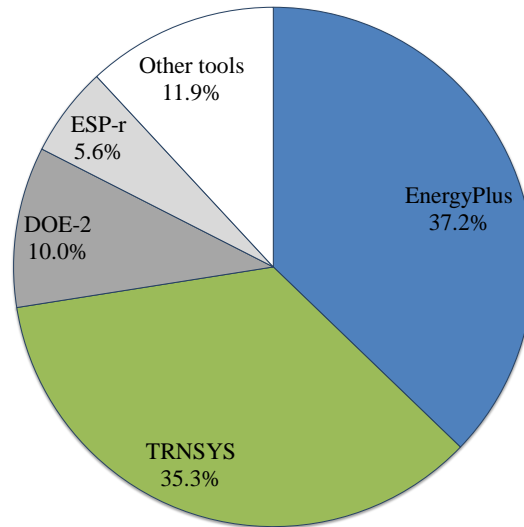


FIGURE 2.4: Utilization share of major simulation programs in building optimization [70]

Despite the many possibilities, given a similar level of modeling detail along with the possibility of a well defined validation procedure as advocated by the relevant ANSI/ASHRAE Standard 140-2007 Standard, it is expected that all simulation engines should yield similar results within the error tolerance which is introduced by virtue of the discretization procedure being used. Unfortunately, a simulation and its' predictive capabilities are as good as the multitude of assumptions (regarding occupancy, plug loads, occupant behaviour, BEMS actions, use of “typical year” weather data etc.) that have to be performed for the input data. In the design and retrofitting phases, where these tools are typically used, reasonable assumptions regarding all aspects of building operation and equipment are made and consequently used in the simulation process. It is very often the case, that such assumptions prove to be wrong and, for this reason, real-world (measured) energy performance can vary significantly from the one estimated upon invocation of the energy simulation models. Hence, a prerequisite for accurate BEP simulation is the capability of the simulation engine to import measured data. For example, does the tool functionality allow importing one-minute measured data as input for a space temperature set point, and does it provide automated routines to accomplish this import? An additional requirement is the tool's ability to integrate measured data into the simulation process. For example, does the tool allow overriding specific water temperatures of the main water loop? Three simulation engines seem to precisely fulfill the aforementioned requirement: EnergyPlus, TRNSYS and Modelica.

### 2.2.1 EnergyPlus

EnergyPlus [18] is a software released by the U.S. Department of Energy. EnergyPlus follows the zonal thermal models paradigm, where the building is divided into spaces (thermal zones), each with a constant temperature, humidity etc. The energy conservation differential equation and the mass conservation differential equation on each zone are used to evaluate the evolution in time of the zonal thermal parameters.

In EnergyPlus structure the whole building system is divided into three main parts: Zone, System and Plant. The entire system consists of many interacting modules which are integrated and controlled by the Integrated Solution Manager. The schematic subroutine calling tree shows the overall structure of the program.

- ProcessInput (InputProcessor)
- ManageSimulation (SimulationManager)
  - ManageWeather (WeatherManager)
  - ManageHeatBalance (HeatBalanceManager)
    - \* ManageSurfaceHeatBalance(HeatBalanceSurfaceManager)
    - \* ManageAirHeatBalance (HeatBalanceAirManager)
      - CalcHeatBalanceAir (HeatBalanceAirManager)

The main input file is the input data file (IDF), an ASCII file which contains information about the building and the HVAC system to be simulated. The EnergyPlus input data are structured into classes. For each class, fields are defined, which describe the characteristics of the class objects. Objects are the instances of a class. All the available classes are listed into the Input Data Dictionary file (IDD). The EnergyPlus Weather file (EPW) is an ASCII, csv format file containing the hourly or sub-hourly weather data needed by the simulation program.

EnergyPlus 8.2.0 is the first version written in C++, while earlier versions of EnergyPlus were all written in the FORTRAN programming language. EnergyPlus 8.2.0 is at least 20% faster than EnergyPlus 8.1.0 for a wide range of models.

A plethora of user interfaces for EnergyPlus exists, including DesignBuilder, EFEN, AECOsim Energy Simulator, N++ and Simergy, to name but a few.

### 2.2.2 TRNSYS

TRNSYS simulation software [49] is a transient systems simulation program with a modular structure where users design a determinate component order that represent real systems installed or to be installed. A wide type library covers most of the component possibilities able to be found in solar system facilities, HVAC systems.

The TRNSYS library includes many of the components commonly found in thermal and electrical energy systems, as well as component routines to handle input of weather data or other time-dependent forcing functions and output of simulation results. The modular nature of the software allows the possibility to create ad-hoc DLL models that represent particular problems not included in the software packages. The structure of a typical TRNSYS deck used in building simulation includes:

- Weather libraries in different formats as can be the Typical Meteorological Year (TMY) in its three different versions, EPW and German TRY's;
- Solar radiation models for fixed and tracked surfaces;
- Control components like PID's, differential or iterative feedback controllers;
- HVAC components like Fan coils, Air Handling Units, Thermal collectors, boilers, heat pumps, pumps, valves;
- Electrical generators as PV collectors, Fuel cells, engines;
- Hydronic systems that connect the HVAC components; and
- Building types. Especially, type 56 represents in an exhaustive way the behavior of zoned divided buildings.

Each one of the previously named TRNSYS types has a FORTRAN code behind with the equations that define their principles and solve the equations that define their working modes.

Some of the types need also of external files that define the efficiency machinery represented (i.e. Heat pumps need of a COP and Partial load files dependent of the two temperatures that define the heat sources) Building types included in TRNSYS usually need of interface software (TRNBUILD) that makes easier the zone definition of the building, its control, set points, occupancy loads, scheduling and output settling to be provided as input for other TRNSYS types or to the program user, allowing him/her to get an accurate overview of the ongoing processes.

The simulation of buildings, understood as the structural envelope and its installed component required of at least three files:

- Simulation control where the solution method, equation solver, simulation times relaxation factors, constants, and tolerance values are defined.
- A file that defines uni-vocally the building structure and the way it should be used.

Last TRNSYS version (TRNSYS 17) also includes:

- an extra file that defines an external shadow matrix that disturbs the solar radiation harvested by the building envelope
- 3D building graphical definition importing IDF Sketch up files.
- Files that define the machinery efficiency under different external conditions

### 2.2.3 Modelica

Modelica [26] is an object oriented, equation based language enabling the modelling of complex physical systems, containing subcomponents, such as electrical, thermal, control, etc., using differential, algebraic and discrete equations. The large number of widely-available libraries modelling such subcomponents situates Modelica among the most powerful and popular simulation tools.

To start, usually a Modelica simulation environment is used, such as OpenModelica or Dymola, providing a component-based GUI for developing Modelica models, to speed up the design process. Consequently, the graphical model is converted to a corresponding text, containing the Modelica language equations describing the model. Finally, the code is translated to C and simulations can be performed.

For building simulation, the Modelica Library for Building Energy and Control Systems is freely-available, containing models for:

- Air-based HVAC systems;
- Water-based heating systems;
- Controls;
- Heat transfer equations;
- Multizone airflow.

A significant advantage of Modelica buildings library over other thermal simulation modelling tools, such as EnergyPlus is the way the warming-up process is performed. Here, a simulation is performed for a predefined warming-up period, using available measurements and the resulting building thermal state at the end of the warming-up period is saved in a text file. Subsequently, a new simulation is initiated and the saved thermal state is loaded and used as the initial conditions of the thermal simulation, thus reducing the computational complexity of tasks requiring frequent calls to the simulation model, like model-based control design optimization modules.

## 2.3 BEP Simulation Data

The accuracy of the predicted outputs depends not only on the calculation engine used, but also on the quality of the input data. The BEP Simulation input Data consist of two components: the static data of the model and the dynamic schedules to assimilate the actual thermal state of the building under proper boundary conditions. Static data include the building geometry, construction materials, glazing information, systems used in the building, etc., while dynamic data consist of all time-dependent data such as user-actions (e.g. opening and closing the windows), occupancy schedules in each of the building zones, use of equipment, weather predictions, etc., commonly being in-building sensed measurements.

Dynamic Data can also be classified into three broad categories:

1. Past data include factual data obtained from in building sensors, in situ weather stations, or other historical data obtained during actual building operation, gathered from the present time instant and backwards.
2. Forecast data refer to predicted data obtained from external services or computed from forecasting modules; these data include weather predictions (obtained from a weather prediction service), occupancy forecasts (obtained from a room scheduling system), etc.
3. Reference data refer to synthetic data that is used in the absence of real measured data or forecasts. These can be reasonable default (reference) values to be used in the simulation when no other information is available. These might be obtained using statistical aggregations of past data (e.g. Meteonorm weather data) or can be inferred based on the building typology or other reasonable assumptions (e.g. occupancy schedules follow the working schedules). Obviously this synthetic data represents “average” schedules to be used in place of real data when these are

not available. An example could be when occupancy sensors are not installed in the building (or occupancy information cannot be indirectly inferred from other sources), reference occupancy data can then form reasonable substitutions.

As mentioned in Chapter 1, development methodologies for semi-automated BEP simulation input data generation make the BEP simulation modeling process much more expedient and as such lower to threshold for the use of such models. BIMs are information-rich repositories that could be used to streamline and expedite the collection of such data. In the following subsections, BEP input data are briefly described, while the ability of BIMs to encapsulate these data, that can be latter automatically transformed to specific BEP data, is presented.

### 2.3.1 Static Data – Building Geometry

The accuracy of thermal simulation depends significantly on the correct representation of the building geometry, commonly being initially generated by a Building Information Model (BIM) authoring tool according to an architectural perspective that must be altered for energy performance simulation tasks [8]. Geometrical data extracted from BIM have to be transformed and combined with material properties to be entered as inputs to energy simulation engines, a process which is time consuming and error prone. Relevant BIM data schemas include the Industry Foundation Classes (IFC) [39] and the green-building XML schema (gbXML). Many commercial authoring tools (e.g. Revit<sup>TM</sup>, Vasari<sup>TM</sup> and ArchiCad<sup>TM</sup>) support export in one of these two BIM schemas. One of the problems is that this export is often not perfect: unlike what would be expected the resulting exports are of poor quality and therefore not directly usable.

The topic of automated data translation between BIMs and thermal simulation tools has received considerable interest as of late. Figures 2.5 and 2.6 present numerous attempts towards developing such an automated data translation process, utilizing the gbXML and the IFC BIM schemas, respectively.

Concerning the preponderant of these attempts, the IDF Generator [7] works in conjunction with the Geometry Simplification Tool (GST) and transforms IFC-format building geometry into EnergyPlus input-data file (IDF); GST simplifies the original building geometry defined in IFC-format and converts it into gbXML-format, while the IDF Generator converts the gbXML-format file into EnergyPlus input-data file. The resulting IDF file contains all information related to building geometry and constructions needed to run an EnergyPlus simulation. For complex geometries, the IDF Generator requires revisiting the generated files to include corrections to windows in curtain walls, missing

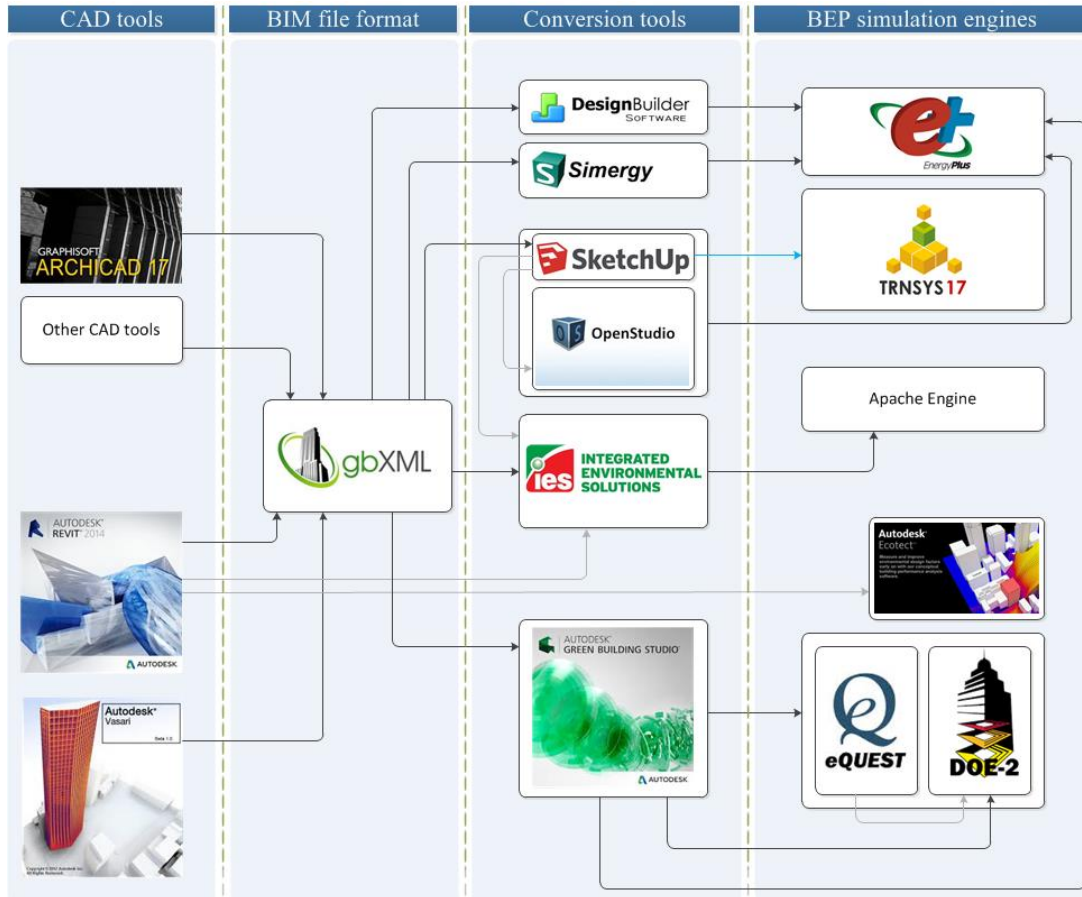


FIGURE 2.5: From gbXML to BEP simulation models — attempts for an automated data translation

floors and ceilings [73]. The RIUSKA [42] uses the DOE-2.1 thermal simulation engine and imports the IFC-defined building geometry, utilizing the BSPro server middleware [44]. Limitations of its IFC import exist, since RIUSKA ignores slabs in the IFC file and generates them internally, according to the size of the space defined by the bounding walls. Moreover, high quality of import results are achieved only when RIUSKA is used in conjunction with SMOG, while problems occur if other BIM tools are used to develop and export the IFC file.

The Green Building Studio (GBS)<sup>TM</sup> web service requires a gbXML-based description of the building which is converted into a DOE-2.2 or an EnergyPlus input file. Studies indicating problems that occur during the conversion process [62] are abundant: most notable errors include the incorrect shading surface definitions and omission of some walls. Google SketchUp along with its Openstudio and IFC2SKP plugins, is able to upload any gbXML or IFC well-formatted geometry and convert it into the EnergyPlus or TRNSYS17-format file. However, due to imperfections of the import tool, information related to floors and ceilings is neglected. Virtual Environment (VE)<sup>TM</sup>, developed by

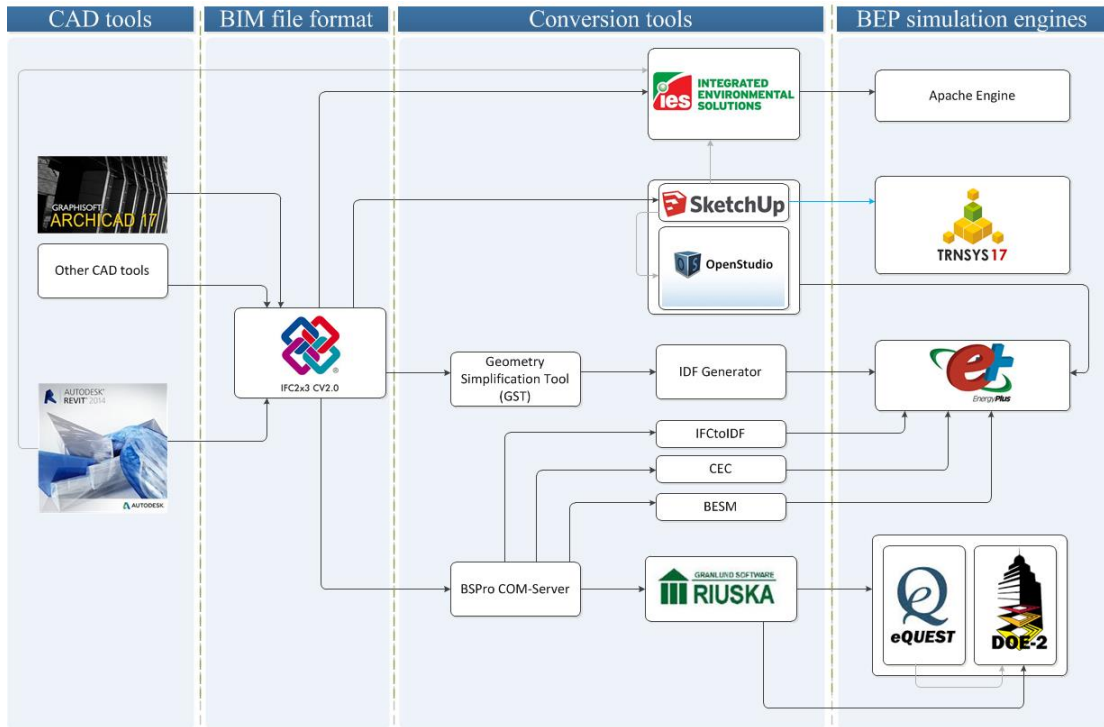


FIGURE 2.6: From IFC to BEP simulation models — attempts for an automated data translation

Integrated Environmental Solutions (IES), is an integrated system that uses the Apache simulation engine. IES VE supports import of gbXML and IFC file formats. Nevertheless, the results depend on the correctness of 2<sup>nd</sup>-level space boundary definitions contained in the IFC file, which currently are not exported properly by any BIM tool.

Among the two most popular BIM schemas, gbXML and IFC, IFC appears to be a suitable choice as its rich content enables interoperability among different software environments and can be easily updated following building's life cycle [32]. Concerning the building geometry, IFC can provide static building information that include geometric configuration and material properties. In many cases however, the necessary for energy simulations 2<sup>nd</sup>-level space boundary data [73], contained in IFC, are missing or are incorrect due to design errors or exporting software imperfections. Hence, a consistent way of transforming the building geometry information, contained in the IFC, to the 2<sup>nd</sup>-level space boundary information is required.

In view of this, a 2<sup>nd</sup>-level space boundary generation algorithm has recently been proposed [90], based on graph theory to convert a three-dimensional architectural building model, without defined thermal space boundaries (2<sup>nd</sup>-level space boundaries), to one suitable for import into a whole-building energy performance model.

### 2.3.1.1 First and Second Level Space Boundaries

From the BEP simulation viewpoint, the amount of information provided from an IFC file is too detailed for thermal simulation purposes. While the building elements such as interior/exterior walls, floors, roofs and openings assume to be monolithic in the architectural view, their subdivision into thermal boundary surfaces is a prerequisite for energy simulation purposes. Hence, reduction, simplification and transformation of the data provided by the architectural view are the thermal simulation geometrical requirements [9]. With respect to the transformation, the process of subdividing architecturally defined surfaces into thermal boundary surfaces is a difficult geometric operation and it is more reliable to be performed by the BIM tool being used to during the building's design phase [8]. This way, according to [104], a space boundary data object, part of the IFC data model and named `IfcRelSpaceBoundary`, differentiates the way that space boundaries-surfaces are defined for architectural and thermal simulation purposes as follows:

- 1<sup>st</sup>-level space boundaries - the building's spaces are defined by space boundaries according to the architectural view;
- 2<sup>nd</sup>-level space boundaries - the building's spaces are defined by space boundaries according to the thermal simulation view.

1<sup>st</sup>-level space boundaries are the boundaries of a space defined by the surfaces of building elements bounding this space. These space boundaries do not take into account any change of material in the bounding building elements, or different spaces behind a wall or slab. 1<sup>st</sup>-level space boundaries form a closed shell around the space and include overlapping boundaries representing openings in the building elements. In order to describe voids, there are 1<sup>st</sup>-level space boundaries with holes and separate 1<sup>st</sup>-level space boundaries representing openings which overlap and are coplanar with the space boundaries representing the host space boundary.

On the other hand, 2<sup>nd</sup>-level space boundaries still represent building elements that bound the space, but these elements are subdivided in cases where:

1. openings are attached to a space boundary,
2. differences in materials of a space boundary occur, and
3. the other side of a building element is subdivided due to the existence of a wall separating different spaces.

2<sup>nd</sup>-level space boundaries represent both sides of a heat transfer surface separated by the thickness of the building element. The building geometry description by 2<sup>nd</sup>-level space boundaries is a prerequisite for the thermal simulation purposes. They can be used by thermal simulation software, but the combination of the two adjacent surfaces to form a single heat transfer surface is required.

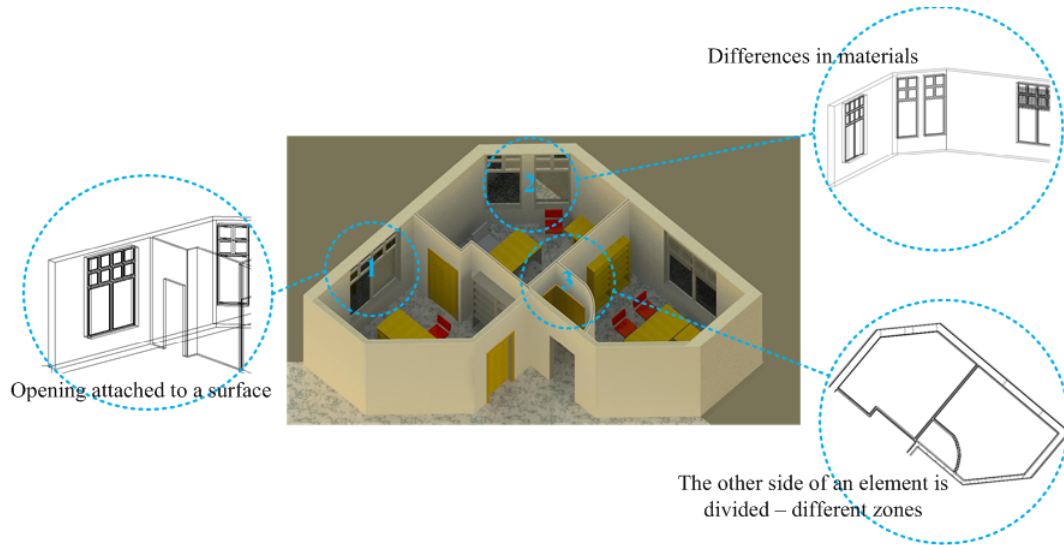
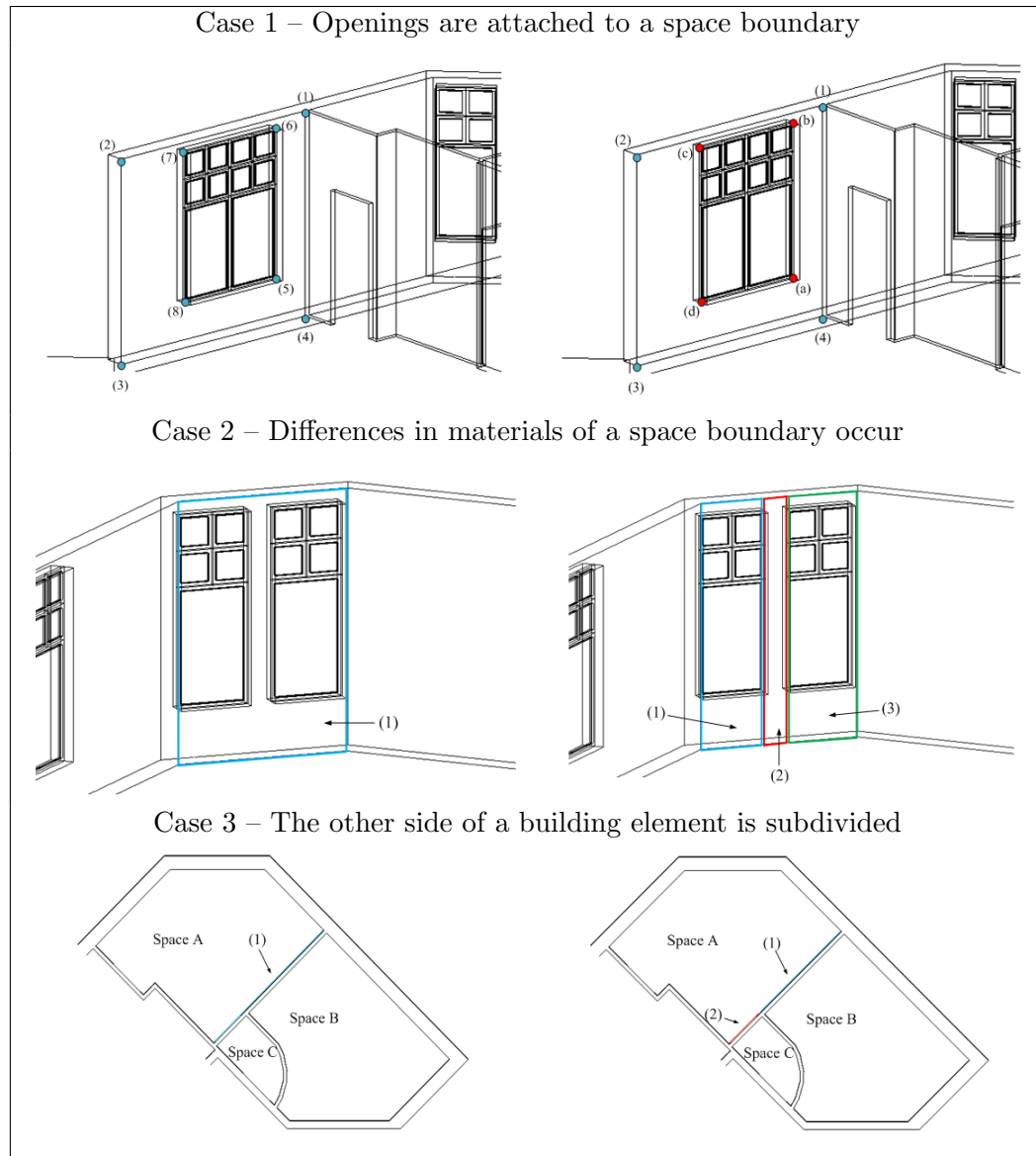


FIGURE 2.7: Cases where building elements are subdivided

The connection geometry of 2<sup>nd</sup>-level space boundaries is restricted to planar surfaces only. Curved surfaces must always be segmented since, 2<sup>nd</sup>-level space boundaries are described by planar surfaces only. Overlapping between 2<sup>nd</sup>-level space boundaries of building elements and their hosted openings is not allowed. Hence, in order to describe a void in a wall, two different common orthogonal surfaces representing the host element and the opening are defined.

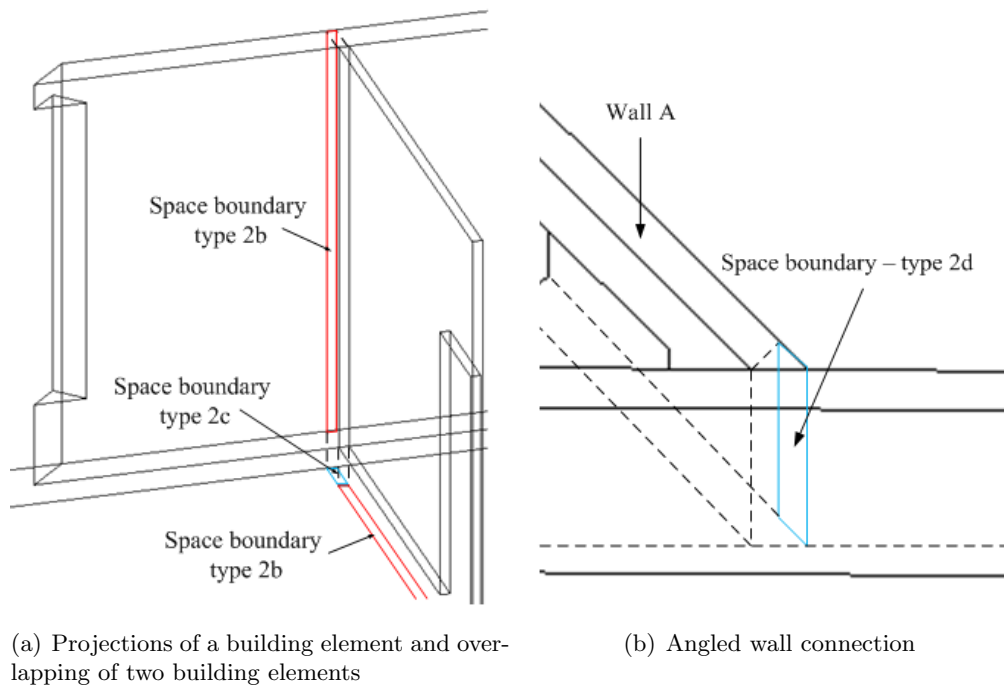
Figure 2.7 is a part of the three-dimensional geometry of a real building, developed in Revit 2014. In this Figure, three examples are identified, where subdivision of building elements is required (the numbers 1, 2 and 3 correspond to the order in which the cases are enumerated above). Table 2.1 presents the differences between 1<sup>st</sup>- and 2<sup>nd</sup>-level space boundaries for these three examples.

As Figure 2.7 depicts, in case 1 an opening is hosted to a wall. In 1<sup>st</sup>-level space boundaries description, a wall with a hole, described by 8 points, is defined (see the 1<sup>st</sup> row of Table 2.1); 4 points describe the vertices of the wall and 4 the vertices of the hole while at the same time, 4 points are required for the description of the opening. In 2<sup>nd</sup>-level space boundaries description, only 4 points are required for the wall definition. The opening constitutes a different space boundary and is defined separately from the wall.

TABLE 2.1: Describing the differences between 1<sup>st</sup> (left) and 2<sup>nd</sup> (right) level space boundaries

Assuming that the construction of the gap between the two groups of windows differs from the construction of the rest of the wall in case 2, differences in materials of a wall occur (see the 2<sup>nd</sup> row of Table 2.1). For the definition of this wall at 1<sup>st</sup>-level, the description of one surface is sufficient; while at 2<sup>nd</sup>-level three different surfaces need to be described.

In case 3, the other side of a Space A boundary is subdivided due to the existence of a wall separating space B and Space C (see the 3<sup>rd</sup> row of Table 2.1). Here, at 1<sup>st</sup>-level space boundary description, this Space A boundary is described by one surface while at 2<sup>nd</sup>-level, its division into two surfaces is required.

FIGURE 2.8: Special types of 2<sup>nd</sup>-level space boundaries

Regarding the data reduction and simplification, there are some special types of 2<sup>nd</sup>-level space boundaries and are observed when the element behind the boundary is a building element. Figure 2.8 presents three common cases of special types of 2<sup>nd</sup>-level boundaries.

Figure 2.8(a) shows two surfaces created by the projection of a wall element on its attached surfaces (red orthogonal) and a surface created by overlapping of two building elements (cyan orthogonal). The definition of these two special types of 2<sup>nd</sup>-level space boundaries is required only in cases where these projections constitutes surface of another space and simulation requires complete enclosure of zone volumes. These, very thin, boundaries increase the complexity of the model, without having significant impact in a BEP simulation. Hence, although different special types of 2<sup>nd</sup>-level space boundaries exist (2b, 2c and 2d), they can be grouped to one type (lets say 2b), which can be neglected during the BIM to BEP translation process.

### 2.3.2 Static Data – Building Materials

Every building element (2<sup>nd</sup>-level spaces boundaries excluding its special types) is associated with a construction. A construction describes a layer bedding, where each layer corresponds to a one or more materials as the example of Figure 2.9 displays.

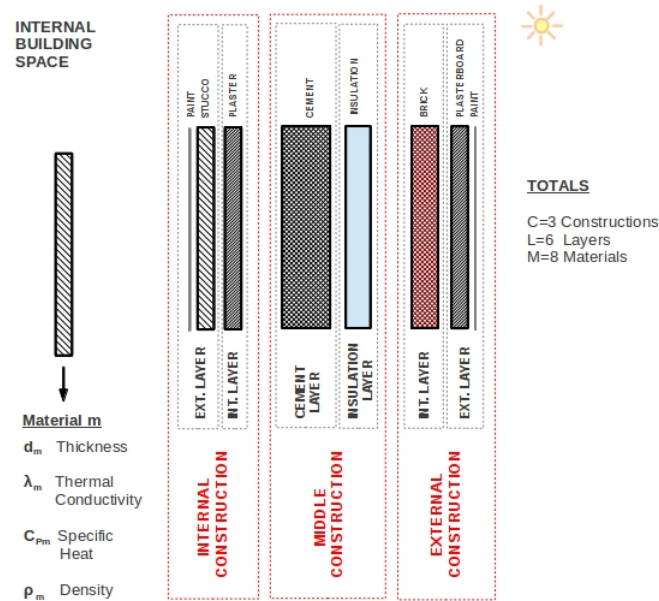


FIGURE 2.9: Example of a building's element construction

Every material present in the building's construction must be listed from the outer to the inner by convention. For example in an external wall the material attached to the inner space air should be listed last and the material attached to the outside air should be listed first. Different materials and their properties, play different roles during the calculations performed by simulation calls.

In IFC each Construction is defined as an `IfcMaterialLayerSet` for which the layer bedding is a list of `IfcMaterialLayers`.

Opaque materials of a construction absorb shortwave (solar) radiation and emit and absorb long-wave (infra-red) radiation. Consequently, knowledge of the values of their properties, referring to these phenomena is required only for the external and internal materials. These properties are characterized as Opaque properties, as their values are required only for the external and internal materials of the constructions.

Furthermore, some materials associated with glazing surfaces (windows, glass doors, etc.) permit a fraction of sunlight to pass through to building interiors. This fraction of sunlit is represented by the solar heat gain and solar transmittance coefficients, which should be provided for every glazing construction and for different values of incident solar beam angles. Material properties associated with only glazing constructions are defined as glazing properties.

Finally, all the construction materials impede and store thermal energy and permit by a fraction or completely block the sunlight. Therefore properties associated with these

functions are universal properties as the knowledge of their values is required for all the materials.

Most IFC to energy simulation transformation methods currently default construction and material thermal and optical properties of building elements. Although IFC could incorporate information about the thermal and optical properties of each building entity construction's material (e.g. an object for thermal conductivity `IfcThermalConductivityMeasure`), current IFC-export tools are not able to export such information.

### 2.3.3 Static Data – Systems

Building systems refer to a variety of devices ranging from active (energy consuming) micro-climate control devices (fans, HVACs, heaters) to passive devices (consuming zero or considerably smaller amounts of energy than active) such as mechanical blinds. Generally, systems include any device which might be uninstalled or installed during the building's operation and affects the conditions of building spaces. Knowledge of the way these devices affect simulation parameters is of paramount importance. Building systems can be classified into the following categories:

- Daylight Control Systems affect the luminance of building spaces. They control the artificial lighting equipment of building spaces depending on whether the luminance levels measured by light sensors are acceptable.
- Shading Control Systems control the luminance (directly) as well as the thermal (indirectly) parameters of building spaces by completely or partially blocking the sunlight passing through building openings.
- Airflow Control Systems control the air quantity entering to or leaving from building spaces. They include natural ventilation devices such as mechanical openings and also forced ventilation devices such as fans.
- Thermal Control Systems These systems affect mainly the temperature of building spaces. They include active devices such as heaters and cooling devices.
- Humidity Control Systems affect the percentage of water vapour present in the air of building spaces. They include devices which either increase this percentage (humidifiers) or decrease it (dehumidifiers).
- Energy Generation Systems are capable of producing energy making use of natural resources such as sunlight (photovoltaic panels) or wind (wind turbines).

Apart from the above classification there are building devices which belong to more than one of the above categories. The definition of systems belongs to the static category, although data related to their operation, and interaction with other building elements is defined as dynamic

Concerning the HVAC data embedded to the BIM and their translation to a BEP simulation model, IFC is the most widely used BIM schema, though suffering from limitations in the description of HVAC systems [89]. Commonly, HVAC modeling in BEP simulation engines requires further information than what is included in an IFC file. A relatively simple, but limited solution is to use available HVAC Templates (like in EnergyPlus, limiting the HVAC input data requirements). Currently, only the IFC HVAC Interface to EnergyPlus effectively exchanges HVAC information between IFC and EnergyPlus simulation [10]; however, only for a range of simplified HVAC systems.

#### 2.3.4 Dynamic Data – Schedules

Simulation engines require timing signals referring to the operation of devices including energy consuming, climate control equipment (such as HVACs, heaters, coolers) and passive devices (such as openings and blinds). These timing signals are in a broad sense, time dependent continuous functions which determine the operation state (on/off) as well as the operation mode characteristics of these devices. To define the values of the schedules used by the simulation programs, the time functions they refer to, have to be sampled at the simulation time instances.

Building spaces often remain unoccupied during specific time periods. By turning off energy consuming devices, during these unoccupied periods, substantial energy can be saved without violating comfort conditions. These time periods are defined by an occupancy parameter which is a binary-valued Schedule. The value of one (1) is used to indicate that, the respective space, is occupied and the value of zero (0), otherwise. Generally, occupancy is a parameter which is difficult to estimate. In order to predict future occupancy values statistical data can be used referring to past occupancy schedules. Consequently, either past occupancy data or forecasts based on these data could be used.

Operating building equipment (computers and electrical equipment) as well as the presence of people act as internal thermal sources (air and surrounding internal surfaces). Since this total amount of thermal energy, is a general non-negative number varying with time, it can be represented by a schedule (a time varying non-negative continuous

function). This schedule is called internal gains schedule and is assigned to every building space. Usually the internal gains are estimated based on the number of people being inside a building space and the operational schedules for equipment in the same space.

The operation schedule of controllable devices play important role in the calculations performed during simulation. The schedules of controllable devices are determined by either model-based or rule-based control decisions.

The IFC schema includes classes for representing operation schedules (*IfcTimeSeries*). Objects of these classes can be attached to individual space instances. However, concerning the existing BIM-authoring tools, the operating schedules can be selected from predefined variants. Moreover, objects of these classes are usually related to another external information layer. Data exchange with external programs is not supported and as such sensed building data can not be used. To overcome such a drawback, sensed data can be forwarded to BEP simulation model through a co-simulation setup (see [Section 2.5](#)).

### 2.3.5 Other Simulation Parameters

Every building simulation process requires timing information in order to be performed as well as knowledge of the required output parameters. This type of information and can be grouped to a new category called other simulation parameters. More precisely, Other Simulation Parameters include:

- Simulation start time (defined by: month, day, hour and minute).
- Simulation end time (defined by: month, day, hour and minute).
- Simulation inter-sample time interval (in minutes).
- Preferred outputs (for example: temperature, humidity, energy demands).

Surface convection and heat balance algorithm options, equipment and system sizing options, daylighting options, dynamic fenestration controls airflow analysis models, etc. also fall into this category. Values of these features can be initially set to default values. However, the modeler must have in mind that these features require domain expertise for input specification and output assessment that cannot be addressed in automated transformation processes.

### 2.3.6 Dynamic Data – Weather Data

The gbXML and IFC schemas are able to store the geographical coordinates of the building location and its orientation. The orientation is normally transmitted together with the geometry data of building components. However, they do not contain any actual information on weather conditions, such as temperature, solar radiation, wind direction, etc. This climate data and weather conditions are derived from geographic coordinates read from external data sources.

Weather data refer to parameters characterizing the building prevailing conditions, consisting of the air surrounding the building and the ground space it is founded on information. They significantly affect the behavior of the building and are one of the major disturbances to BEP simulation, thus accurate weather measurements and/or forecasts are essential for enhanced model accuracy.

A plethora of weather data for BEP simulation exist, ranging from locally recorded weather data to preselected ‘Actual’ or ‘Typical’ years.

Concerning the reference data, several organizations have developed typical weather data sets. Examples of these typical data include TMY2 in the United States and TRY in Europe, enriched with more solar radiation data than older formats such as TMY and the older version of TRY.

However, the traditional simulated energy performance, using reference data, represents the building performance for a typical year but not necessarily the average or typical long-term performance as buildings with different energy systems and designs respond differently to weather changes. Hence, even for energy auditing purposes the use of reference data poorly estimate the actual thermal behavior of a demonstration building.

Furthermore, it is obvious that the use of reference data in control-design processes is prohibitive. For instance, a BEP model-assisted control design process consists of the warming-up and the forecast phase [28]. During the warming-up phase, the BEP simulation model is used — combined with past weather data — to estimate the actual thermal state of the building at the beginning of the forecast phase. Subsequently, when the warming-up phase finishes, a model-assisted stochastic optimization algorithm, trying to solve a constraint optimization problem that requires minimization of energy consumption, while preserving user thermal comfort levels, evaluates a series of candidate controllers, using the “warmed-up simulation model” along with weather forecasts to design a new controller during the forecast phase. Thus, weather data for a BEP model-assisted control design process must consist of past and forecast data.

In BEP simulation engines, weather data are provided as input through a properly formatted weather file. The weather file creation process can be defined as a dedicated service; every time a new weather-file is required, the service is invoked, using as arguments the starting date, the end date and the time-stepping for the weather file. Subsequently, the service gathers past weather data and weather predictions (if necessary) from the appropriate databases. A major drawback during the weather file creation process is that commonly, weather predictions do not include information concerning solar radiation, but only a percentage of cloud coverage.

Although location data can be exported and stored, the inability of the simulation tools to read such information from the BIM model file is disadvantageous. For example, when importing both exchange formats into the Ecotect Analysis simulation tool, the location is reset to the default city of New York and there is no data exchange.

### 2.3.6.1 Solar Radiation Models

Towards estimating the global solar radiation from existing percentage of cloud coverage data, the forecasting model described in [46] can be adopted.

With past and estimated-forecasted global radiation data at hand, a crucial input in the simulation of buildings' energy performance is the availability of both diffuse and direct radiation data. Nevertheless, another drawback is that in most cases, measured data of diffuse and direct radiation are not available. On the other hand, measurements of global radiation are available for numerous different locations. Towards estimating hourly diffuse fraction of global solar radiation  $f_d$  a plethora of models is available and there are many studies comparing them [19, 20, 41, 71, 99]. Most of these models are polynomial correlations of different orders [19, 25, 34, 35, 41, 76, 77, 86]. These models correlate the diffuse fraction with the clearness index  $k_t$ , the ratio of a particular hour's global radiation  $I$  to the extraterrestrial radiation for that hour  $I_o$ . Boland et al. [14] proposed an exponential function to obtain the hourly diffuse fraction using the clearness index as an input. An exponential function was also used by Ridley et al. [88] where a persistence index is introduced in order to take into account dynamics of the process. Skartveit et al. [93] developed a procedure for determining the hourly diffuse fraction and include a variability index that accounts for the influence on the diffuse fraction of the changes in the type and location of the clouds and the surface albedo. In a quasi-physical model, proposed by Maxwell [66], the hourly normal direct irradiance is determined based on physical expressions including the clearness index and the relative air mass. Finally, the Dirint model [78] is an improvement of the Maxwell model where, a coefficient from a 6 x 6 x 5 x 7 look-up table, that depends on the following parameters,

is introduced: the clearness index, the sun elevation, the dew point temperature and a variability index for considering the dynamics of the process. However, a 3-D version of the model exists, which does not require the dew point temperature data.

The question of the best method for estimating the diffuse (or direct) fraction of global solar radiation are not fully settled. In many studies [19, 20, 41, 99] the different established models and correlations that calculate hourly solar radiation components have been selected, performed and tested to decide which model is recommended.

## 2.4 Simulation Speed-Up

The high complexity of building simulation stemming from the requisite information, including actual climate data, geometries, building physics, HVAC-systems, energy generation systems, natural ventilation, user behavior (occupancy, internal gains, manual shading) to name but a few, makes the development of accurate simulation models a challenging task. It is becoming quite common, especially during the design (or subsequent retrofitting) phases of a building lifecycle, that simulation models are employed to prognosticate energy performance and help identify underlying problems with respect to energy design. As mentioned earlier, the calculation methodologies used can range from “simple” quasi-steady-methods [38] to dynamic, implemented in energy-performance simulation zonal-type software like EnergyPlus [18] and TRNSYS [49]. Each calculation method supports different use cases and, as such, the modeling assumptions and the associated inputs can vary greatly in the levels of detail and information that has to be provided. The zonal approximation is acceptable for many envisaged and practical use scenarios, as it manages to strike a balance between accuracy and computational complexity.

Beyond the zonal-paradigm approaches, models developed specifically for a particular purpose (e.g. control design) exist. For model-based control design purposes, typically state-space models adhering to certain mathematical constraints are required. The accuracy of such models may not be of essence but rather their ability to correctly capture dynamics and sensitivities of the system that is being modeled, as this is the critical quality needed for control design. For computational efficiency, the number of states should be “small”, as repeated evaluations might be required within the control design context where they will be applied.

In the building application domain, the development of such models remains an open problem [85]. Towards defining such models many approaches are possible. The simulation models can be developed using first principle approaches [55, 75, 94], but for

larger buildings their construction is impractical, due to the increased complexity [82–84]. Data-driven models, produced by system identification methods [50, 100], can be viable approaches, but still it is very often that the identification process fails when applied on real, occupied buildings, due to under-excitation of system dynamics [100].

In more recent approaches [84] a full scale, zonal-type, thermal simulation model of the building is used for the identification phase, which remains computationally expensive. What is particularly attractive in this approach is that excitation necessary for the identification happens at the simulation level, so it is possible to excite the system in many ways that would be impractical, or even unrealistic, if they were to be applied in the real building. As can be inferred from the discussion above, the use of simplified models that are highly computationally efficient can be a viable option. When accuracy and computational effort constraints are present, simulation speed-up techniques like the ones discussed in this paper, can yield a good trade-off between accuracy and computational complexity. Even though for small buildings, full scale models have been used successfully for model assisted control design [29, 52], field studies [36] have shown that, for large buildings a very detailed model is impractical to simulate, since it is too complex and takes too long to run. Simulation is predominantly slowed down by the increasing number of zones, windows and surfaces, and as such simulation speedup techniques should focus on geometry reduction, zoning reduction and/or order reduction approaches.

## 2.5 Co-Simulation

A review of the previous sections reveals two fundamental simulation functional requirements: a) the availability of a mathematical model for describing all the components of the building, along with a proper solver for executing/simulating the model and providing outputs in each requested timestep, and b) the ability to simulate the model, using external inputs as values for certain model parameters (e.g. control schedules, occupancy patterns, etc.). Since for both the above requirements significant research effort is underway, a variety of software tools is available.

Establishing a link between the calculation methodology and the external data sources, so that the dynamic schedules can be incorporated in the simulation, is of paramount importance for achieving BEP simulation tasks. This functionality is especially desirable also for the testing and design of control strategies. Creating therefore an interface to enable two-way communication at each time step of the simulation is essential, and this functionality we refer to as co-simulation.

In research studies two promising approaches are used for co-simulation; the Buildings Controls Virtual Test Bed (BCVTB) [105] and the Functional Mockup Interface (FMI) [68]. In the former, an additional transaction layer is defined for the data-exchange, increasing the complexity of the co-simulation. In the latter, the independent tools communicate through a standardized interface (FMI).

The FMI standard [68] defines a set of C-functions that are needed to perform co-simulation with other simulation programs. The FMI also defines an XML-file (model description file) which contains all information required by the importing tool to inquire information about the model and its interface variables. Tools that support FMI can import and/or export a simulator for co-simulation. The exported simulator or model is called an Functional Mockup Unit (FMU) [72]. An FMU is distributed in form of a zip file. This file may contain: the FMI model description file; the C sources of the FMU, including the needed run-time libraries used in the model, and/or binaries for one or several target machines; additional FMU data (such as tables, diagram) in FMU specific file formats.

Although FMI seems to be a promising attempt, currently BCVTB appears to be a more mature co-simulation approach. BCVTB, an enhanced flavor of Ptolemy II heterogeneous modeling and design framework, is a software environment, developed by Lawrence Berkeley National Laboratory which allows a dynamic data exchange between the two or more programs, at each time step of the co-simulation. The latest version of the software supports coupling of (building or system) models developed in Modelica, EnergyPlus, TRNSYS detailed thermal simulation engines, as well as models developed in a variety of programming languages, such as Fortran, C or Matlab.

## Chapter 3

# CBIP Algorithm

Following a different approach to address the 2<sup>nd</sup>-level space boundary generation requirement, in this Chapter the Common Boundary Intersection Projection (CBIP) algorithm is presented. A recent study has shown that BEP models obtained based on CBIP algorithm results, are comparable to models that developed from scratch in other popular softwares [56, 58].

Algorithmically, CBIP is divided into four operational stages: the Identification (ID) stage, the Boundary Surface Extraction (BSE) stage, the Common Boundary Intersection (CBI) stage and the Boundary Intersection Projection (BIP) stage which are analyzed in Section 3.4. CBIP's stages involve geometric operations based on well-known methods for representing shapes, therefore an initial description of such methods, adopted by the algorithm, are presented in Sections 3.2 and 3.3. Design requirements and design recommendations to ensure the correct execution of the algorithm are discussed in Sections 3.5 and 3.6, respectively. Finally, CBIP algorithm is evaluated on two demonstration buildings and its results are presented in Section 3.7.

### 3.1 IFC Data Quality

The IFC file schema manages to represent building's geometry with a very compact and precise way, accommodating multiple geometry representations in a single file. Although IFC has the necessary classes to support the description of building geometry, these classes sometimes contain incorrect data, mainly induced due to two reasons: either the designer has erroneously defined certain building entities, or the IFC-exporter software has flaws and exports incorrectly the geometrical data in the IFC file.

For the 2<sup>nd</sup>-level boundary generation process, the input geometry should be of good quality; such errors can slow or make that process to fail. In [73], errors that affect the creation of properly defined 2<sup>nd</sup>-level space boundaries are presented. Consequently, as a post-processing step, after the IFC file export, such errors should be detected and corrected either automatically or manually by communicating them back to the designer.

Concerning their detection, existing software packages such as Solibri Model Checker<sup>TM</sup>, perform detection (checking) of geometric inconsistencies, which are communicated back to the BIM software and corrected manually. In cases of exporter induced errors, it is hard to rectify them unless we work directly with the IFC file — something that requires quite often expert knowledge. Hence, automatic error detection and correction mechanisms, should be employed, however within limits; unless there are unambiguous conditions (“small” errors) which can be automatically corrected, these errors have to be manually corrected by the designer. Towards automatic detection and correction of such inaccuracies, detection and correction algorithms have been developed and could be performed, though their description is not the topic of this work.

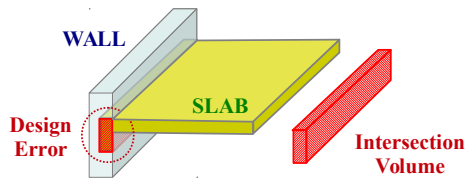
### 3.1.1 Geometric Error Classification

The geometry errors that affect the creation of BEP simulation models can be classified into three categories namely: design errors, space definition errors and surface orientation errors.

Design errors occur when two or more geometric representations of solid building entities intersect each other; for instance, when a building wall intersects a neighbor slab, or when a building space is not defined correctly and includes parts of neighbor building walls. Such cases are displayed in Figure 3.1.

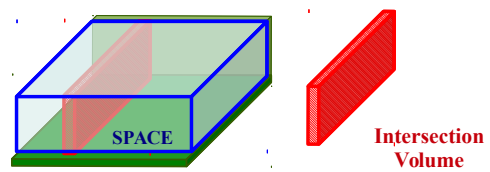
#### DESIGN ERROR EXAMPLES

A. SLAB-WALL design error



A building slab intersects with a building wall.  
The volume of the intersection wall-slab is nonzero.

B. SPACE-WALL design error



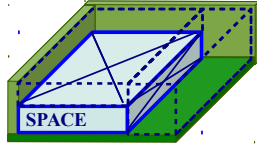
A building space contains an internal wall.  
The volume of the intersection space-wall is nonzero.

FIGURE 3.1: Design error examples – A. Wall-Slab and B: Space-Wall

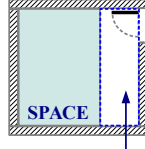
Building spaces must be completely enclosed by other building entities. In other words, each space boundary surface must touch another building's entity boundary surface. However, there are cases where, spaces are incorrectly defined leaving undefined volume gaps. An example of an incorrect and respective correct internal space definition is presented in Figure 3.2.

#### INCORRECT SPACE DEFINITION EXAMPLE

Incorrect space definition

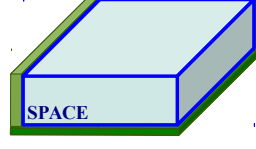


Space is incorrectly defined leaving undefined volume gaps indicated by the dashed lines



Undefined space volume

Correct space definition



Space is correctly defined with no undefined volume gaps.

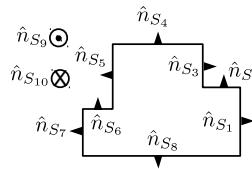
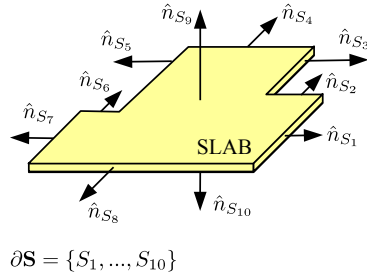


No undefined space volume

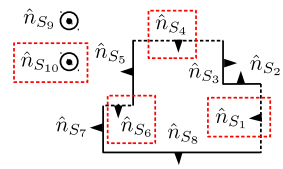
FIGURE 3.2: Space incorrect and correct definitions example

As a result of the IFC exporting process, performed by routines, encapsulated in BIM tools, there are cases where geometrical data referring to B-reps of building solids contain surfaces which are not oriented according to the right-hand rule (see Section 3.2.1) and as such surface orientation errors exist. Examples of correct and incorrect orientations in the surfaces of a B-rep are demonstrated in Figure 3.3. Although in some 3D-rendering applications, enforcing correct orientation of the building B-reps' surfaces is not of interest, there are other applications, which use computational geometry and correct orientation of surfaces is a prerequisite.

#### SURFACE ORIENTATION ERROR EXAMPLE



Correctly oriented B-rep surfaces



Incorrectly oriented B-rep Surfaces (Errors : dashed boxes)

FIGURE 3.3: B-reps of a slab with correctly and incorrectly oriented surfaces

## 3.2 Geometric Defintions

CBIP takes as input the geometric representations of various building entities, which are assumed to be polyhedrons, performs certain operations on them and outputs polygonal surfaces which are the 2nd-level space boundaries. Consequently, CBIP algorithm's mathematical foundation consists of geometric operations, applied on geometric representations of the involved building entities.

Two methods of geometric representation are used in CBIP, related to polyhedral objects. CBIP operates on the boundary surfaces of the building entities' polyhedrons to extract the space boundary surfaces, which are common surfaces among space and other construction polyhedrons. Consequently, the first required geometric representation method of the building polyhedrons is the Boundary representation (B-rep) [40], described in Section 3.2.1. B-rep theory is adopted in order to describe each polyhedron by its corresponding boundary polygons. Additionally, to determine the space boundaries which are essentially common surfaces shared by two polyhedrons, another polyhedral representation called Boundary Space Partitioning tree (BSP-tree) representation [97] is adopted and described in Section 3.2.2.

### 3.2.1 Boundary representation

The B-rep of a polyhedron  $\mathbf{A}$  associated with a building entity, is denoted by  $\partial\mathbf{A}$ . Essentially,  $\partial\mathbf{A}$  is a set of boundary polygonal surfaces  $\partial\mathbf{A} = \{A_1, \dots, A_i, \dots, A_N\}$  (see Figure 3.4). In order to facilitate the geometric operations of CBIP, the boundary polygons of these representations conform to the right hand outward normal convention: the normal vector  $\hat{n}_{A_i}$  of every boundary polygon  $A_i$  points outside the polyhedron  $\mathbf{A}$ , as displayed in Figure 3.4.

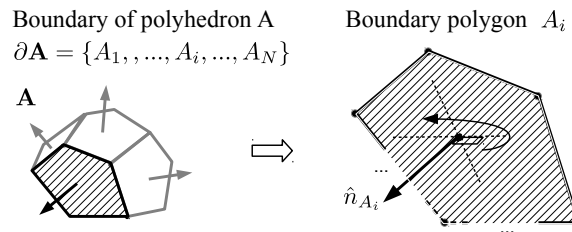


FIGURE 3.4: B-rep of a building entity by a set of boundary surfaces

### 3.2.2 Boundary Space Partitioning tree representation

The BSP-tree representation of a polyhedron  $\mathbf{A}$  is denoted by  $T_{\mathbf{A}}$ . An example of such representation is displayed in Figure 3.5.  $T_{\mathbf{A}}$  contains the boundary polygons of polyhedron  $\mathbf{A}$  and defines a partition of the 3D space into a finite set of sub-spaces, depending on the orientation of the polyhedron's boundary surfaces. In a broad sense, a BSP-tree of a polyhedron is a binary tree that partitions the 3D space into finite number of sub-spaces according to the outward normal vectors of its boundary surfaces, as described below.

$T_{\mathbf{A}}$  is a structure with three fields. The root value of a BSP-tree  $T_{\mathbf{A}}$  contains a single-root polygon, or multiple-root coplanar polygons with identical normal vectors, and is denoted by the *val* field ( $T_{\mathbf{A}}.val$ ). The plane of the root divides the 3D space into two sub-spaces, the outside and the inside sub-space. The outside sub-space is indicated by the common normal vector of the root polygon(s). The outside sub-space contains polygons which are placed in the right sub-tree of  $T_{\mathbf{A}}$  and is denoted by the field  $T_{\mathbf{A}}.out$ . The inside sub-space is indicated by the opposite of the normal vector of the root polygon(s). The inside sub-space contains polygons which are placed in the left sub-tree of  $T_{\mathbf{A}}$ , and is denoted by the field  $T_{\mathbf{A}}.ins$ .

Consequently, moving from the root to the leaves of the tree and following the left/right branches lead to inside/outside sub-spaces (opposite/towards the direction of the outward normal vectors), respectively. The final partitions (sub-regions) of the 3D space are indicated by the leaves of the tree which contain binary values. By convention, a leaf has value 1, if the respective sub-region is inside the polygons of the node above the leaf, and the value 0, if the respective subregion lies outside these polygons.

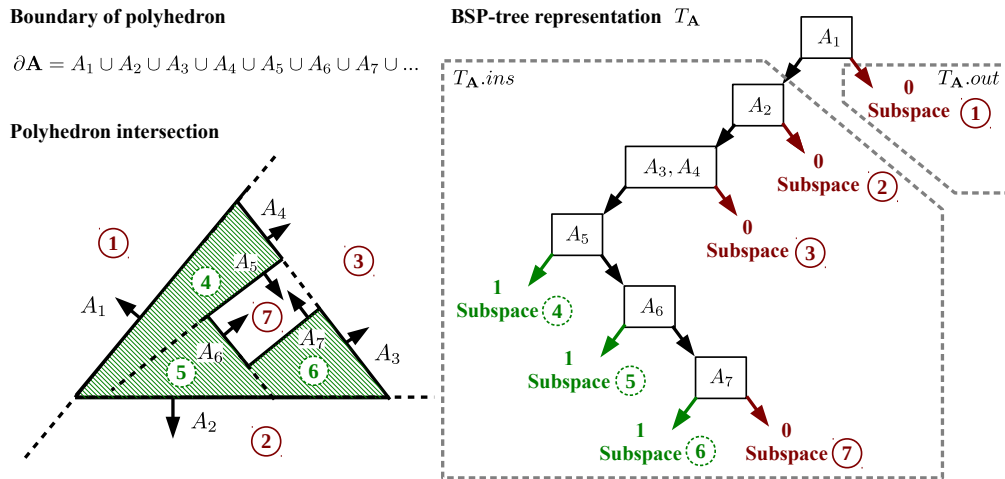


FIGURE 3.5: Cross section and BSP-tree representation for a set of polygon boundary surfaces forming a non-convex region

If two boundary polygons are coplanar but their outward normal vectors have opposite directions, one polygon is considered to lie outside the other and therefore are placed in separate root nodes. The sequence of the boundary polygons of  $\partial\mathbf{A}$ , used to populate the BSP-tree  $T_{\mathbf{A}}$ , does not matter. The tree representation  $T_{\mathbf{A}}$  of a polyhedron  $\mathbf{A}$  is obtained from its B-rep representation  $\partial\mathbf{A}$  using the recursive algorithm described in [97].

### 3.3 Geometric Operations

In order to obtain the 2<sup>nd</sup>-level space boundary surfaces, which are parts of common boundary surfaces between the polyhedral representations of building spaces and the polyhedral representations of building constructions, CBIP performs geometric operations defined by three geometric clipping functions. These clipping functions are applied on polyhedral pairs  $\mathbf{A}$ ,  $\mathbf{B}$ , and use: (1) their B-reps  $\partial\mathbf{A} = \{A_1, \dots, A_{N_{\partial\mathbf{A}}}\}$ ,  $\partial\mathbf{B} = \{B_1, \dots, B_{N_{\partial\mathbf{B}}}\}$ , with  $N_{\partial\mathbf{A}}$ ,  $N_{\partial\mathbf{B}}$  the cardinalities of the sets  $\partial\mathbf{A}$ ,  $\partial\mathbf{B}$ ; (2) the respective BSP-tree representations  $T_{\mathbf{A}}$ ,  $T_{\mathbf{B}}$ ; (3) two polygon clipping operators  $c_1$  and  $c_2$ ; and (4) a polygon set partition function.

#### 3.3.1 Polygon clipping operators

Polygon clipping operators  $c_1$  and  $c_2$  involve two polygons  $A_i$  and  $B_j$ . Essentially,  $c_1$  and  $c_2$  modify their second operand (polygon  $A_i$ ), depending on the the relative position of their first operand (polygon  $B_j$ ) and the direction of its normal vector  $\hat{n}_{B_j}$  (see Figure 3.6).

Mathematically, these operations are defined by:

$$A_{1i} = B_j(c_1)A_i \text{ and } A_{2i} = B_j(c_2)A_i \quad (3.1)$$

Generally, three clipping cases can be distinguished:

- A. The plane of  $B_j$  *dissects*  $A_i$  into two parts:  $A_{1i}$  towards the normal vector  $\hat{n}_{B_j}$  and  $A_{2i}$  towards the opposite direction  $-\hat{n}_{B_j}$ . This dissection is performed by  $c_1$  or  $c_2$ , and return  $A_{1i}$  or  $A_{2i}$ , respectively, with  $A_i = A_{1i} \cup A_{2i}$ .
- B1. The plane of  $B_j$  *dissects the plane of*  $A_i$  into two half-planes and  $A_i$  is in the half-space pointed by  $\hat{n}_{B_j}$ . In this case  $c_1$  returns  $A_i$  and  $c_2$  returns an empty set.

B2. The plane of  $B_j$  dissects the plane of  $A_i$  into two half-planes and  $A_i$  is in the half-space pointed by  $-\hat{n}_{B_j}$ . In this case  $c_1$  returns an empty set and  $c_2$  returns  $A_i$ .

The previous clipping operations are illustrated by 2D set operations on polygons: intersection, union and subtraction. These 2D set operations, are implemented using the algorithm proposed in [101].

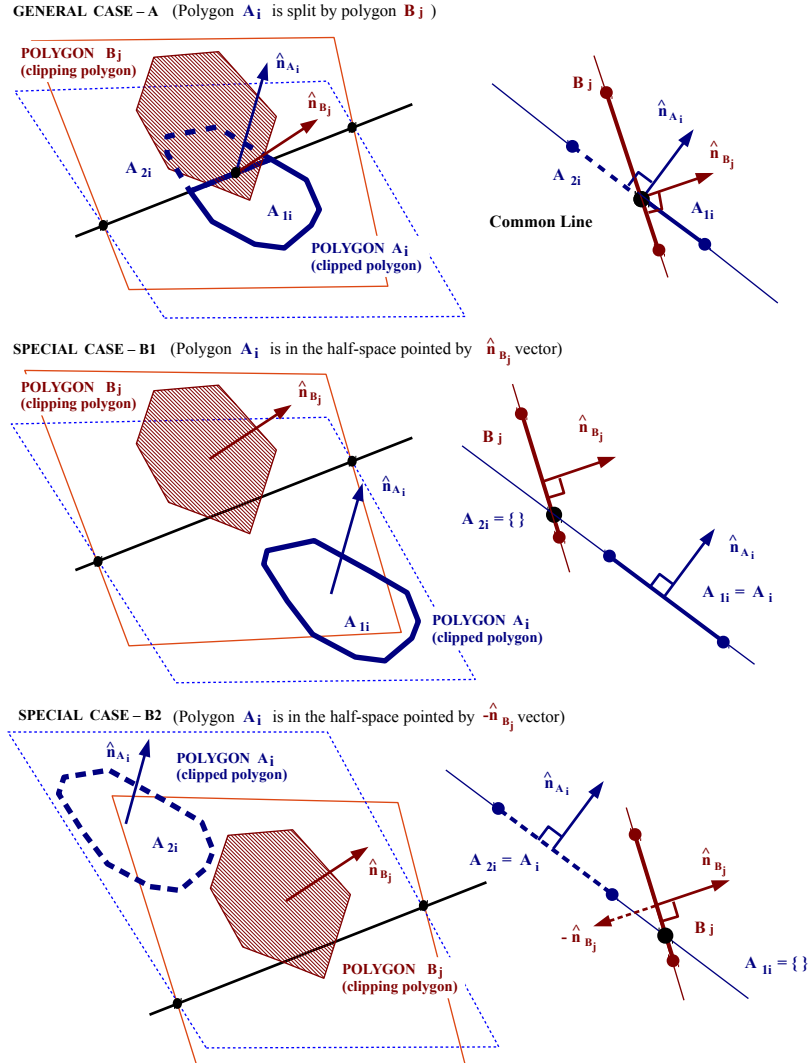


FIGURE 3.6: Illustration of polygon clipping operators  $c_1$  and  $c_2$  on polygon  $A_i$  by polygon  $B_j$ .

### 3.3.2 Polygon set partition function

The polygon set partition function  $P$ , used by the geometric clipping functions, can be defined as a partition of a polygon set  $\mathcal{A}$ , intersected by a set of coplanar polygons

(partition set  $\mathcal{B}$ ) with the same outward normal vector  $\hat{n}_B$  (see Figure 3.7).

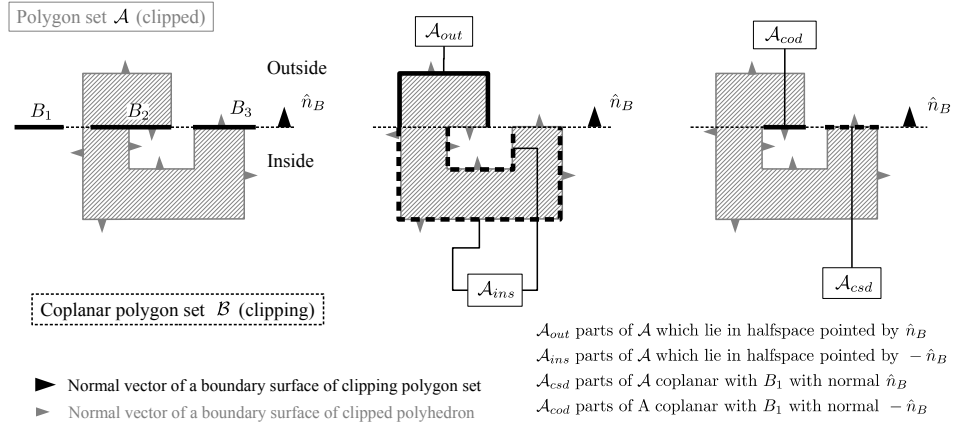


FIGURE 3.7: Illustration of the polygon set partition function

---

**Algorithm 1** Partition function  $P(\mathcal{B}, \mathcal{A})$

---

```

1:  $\mathcal{A} = \{A_1, \dots, A_N\}$  // Polygon set (to be partitioned) //
2:  $\mathcal{B} = \{B_1, \dots, B_M\}$  // Polygon set (partitioning set) //
3:  $\mathcal{A}_{ins} = \emptyset, \mathcal{A}_{csd} = \emptyset, \mathcal{A}_{cod} = \emptyset, \mathcal{A}_{out} = \emptyset$  // Initialize output sets //
4: for  $i = 1 : N$  do
5:   if  $A_i \in \mathcal{A}, B_1 \in \mathcal{B}$  are coplanar then
6:     for  $j = 1 : M$  do
7:        $ASB_{ij} = A_i - B_j$  // Subtract  $B_j$  from  $A_i$  and form polygon  $ASB_{ij}$  //
8:        $\mathcal{A}_{out} \leftarrow \mathcal{A}_{out} \cup ASB_{ij}$  // Include polygon  $ASB_{ij}$  in the  $\mathcal{A}_{out}$  set //
9:        $AIB_{ij} = A_i \cap B_j$  // Intersect  $B_j$  with  $A_i$  and form  $AIB_{ij}$  polygon //
10:      if  $\hat{n}_{A_i} \uparrow \hat{n}_{B_j}$  then
11:         $\mathcal{A}_{csd} \leftarrow \mathcal{A}_{csd} \cup AIB_{ij}$  // Include polygon  $AIB_{ij}$  in the  $\mathcal{A}_{csd}$  set //
12:      else
13:         $\mathcal{A}_{cod} \leftarrow \mathcal{A}_{cod} \cup AIB_{ij}$  // Include polygon  $AIB_{ij}$  in the  $\mathcal{A}_{cod}$  set //
14:      end if
15:    end for
16:  else
17:     $\mathcal{A}_{ins} \leftarrow \mathcal{A}_{ins} \cup [A(c_2)B_1]$  // Include clipped polygon  $[A(c_2)B_1]$  in the  $\mathcal{A}_{ins}$  set //
18:     $\mathcal{A}_{out} \leftarrow \mathcal{A}_{out} \cup [A(c_1)B_1]$  // Include clipped polygon  $[A(c_1)B_1]$  in the  $\mathcal{A}_{out}$  set //
19:  end if
20: end for

```

---

Mathematically, the polygon set partition function  $P$  is defined by the following expression:

$$[\mathcal{A}_{ins}, \mathcal{A}_{csd}, \mathcal{A}_{cod}, \mathcal{A}_{out}] = P(\mathcal{B}, \mathcal{A}) \quad (3.2)$$

The returning arguments  $\mathcal{A}_{ins}$  /  $\mathcal{A}_{out}$  are subsets of the set  $\mathcal{A}$  containing polygons lying in the half-space pointed by  $-\hat{n}_B$  /  $\hat{n}_B$ , respectively.  $\mathcal{A}_{cop}$  /  $\mathcal{A}_{csd}$  contain polygons coplanar with the polygons in  $\mathcal{B}$ , which have opposite (cod) / same direction (csd) normals with  $\hat{n}_B$ , respectively (see Figure 3.7). The above sets are populated using Algorithm 1 and the polygon clipping operator  $c_1$  and  $c_2$ .

### 3.3.3 Polygon set clipping functions

Using the operators  $c_1$ ,  $c_2$ , and the polygon partition function  $P$ , three recursive clipping functions  $F_{ins}$ ,  $F_{out}$  and  $F_{cod}$  are defined. These functions are applied on a polygon set  $\mathcal{A}$  (clipped polyhedron), using the BSP-tree representation  $T_B$  of a polyhedron  $B$  (clipping polyhedron). These clipping functions return:

- $\mathcal{A}_{ins} = F_{ins}(T_B, \mathcal{A})$ : The parts of  $\mathcal{A}$ , which are inside polyhedron  $B$ ;
- $\mathcal{A}_{out} = F_{out}(T_B, \mathcal{A})$ : The parts of  $\mathcal{A}$ , which are outside polyhedron  $B$ ;
- $\mathcal{A}_{cod} = F_{cod}(T_B, \mathcal{A})$ : The parts of  $\mathcal{A}$ , which are coplanar with the surfaces of  $\partial B$  and have opposite outward normal vectors.

Function  $F_{ins}$  is described by Algorithm 2, function  $F_{out}$  by Algorithm 3 and function  $F_{cod}$  by Algorithm 4. In these Algorithms, the clipping BSP-tree  $T_B$  has three fields:  $T_B.pol$  refers to the polygons contained in the root of  $T_B$ ;  $T_B.ins$  contains the left (inside) sub-tree of  $T_B$ ; and  $T_B.out$  contains the right (outside) sub-tree of  $T_B$ .

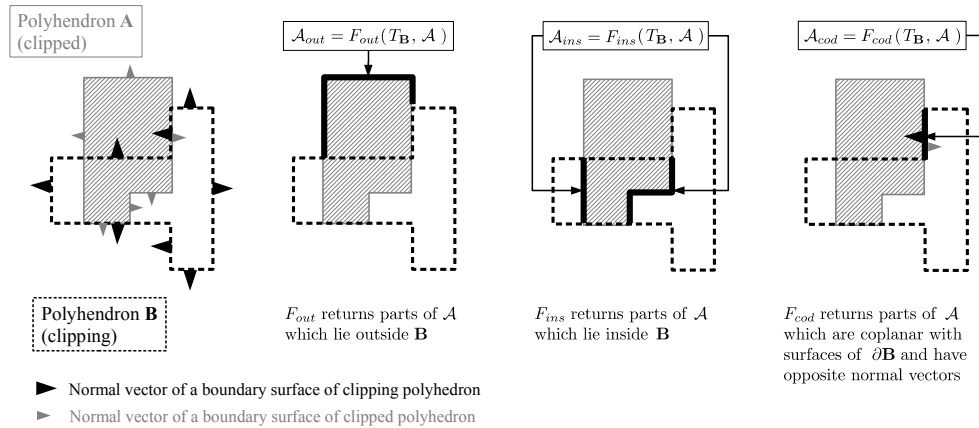


FIGURE 3.8: Results of clipping functions –  $\mathcal{A}$  is the polygon set of a clipped polyhedron and  $B$  is the clipping polyhedron

The  $F_{cod}$  function is used by CBIP to identify the Common Boundary Intersection surfaces, which are coplanar surface pairs belonging to two different polyhedrons and

have opposite normal vectors. Examples of the clipping functions  $F_{out}$ ,  $F_{ins}$  and  $F_{cod}$ , applied on  $\mathcal{A}$ , using a clipping polyhedron  $\mathbf{B}$  are displayed in Figure 3.8.

---

**Algorithm 2** *Inside clipping function  $F_{ins}$ :*  $\mathcal{A}_{ins} = F_{ins}(T_{\mathbf{B}}, \mathcal{A})$

---

```

1: if  $T_{\mathbf{B}}$  is binary then
2:   if  $T_{\mathbf{B}} = 0$  then
3:      $\mathcal{A}_{ins} = \emptyset$  // Initialize output set  $\mathcal{A}_{ins}$  //
4:   end if
5:   if  $T_{\mathbf{B}} = 1$  then
6:      $\mathcal{A}_{ins} = \mathcal{A}$  // Initialize output set  $\mathcal{A}_{ins}$  with  $\mathcal{A}$  //
7:   end if
8: else
9:    $[\mathcal{A}_{ins}, \mathcal{A}_{csd}, \mathcal{A}_{cod}, \mathcal{A}_{out}] = P(T_{\mathbf{B}}.val, \mathcal{A})$  // Partition  $\mathcal{A}$  with  $T_{\mathbf{B}}.val$  //
10:  if  $\mathcal{A}_{ins} \neq \emptyset$  then
11:     $\mathcal{A}_{i,ins} = F_{ins}(T_{\mathbf{B}}.ins, \mathcal{A}_{ins})$  // Apply  $F_{ins}$  recursively on  $\mathcal{A}_{ins}$  with BSP tree  $T_{\mathbf{B}}.ins$  //
12:     $\mathcal{A}_{ins} \leftarrow \mathcal{A}_{ins} \cup \mathcal{A}_{i,ins}$  // Include  $\mathcal{A}_{i,ins}$  in  $\mathcal{A}_{ins}$  //
13:  end if
14:  if  $\mathcal{A}_{out} \neq \emptyset$  then
15:     $\mathcal{A}_{i,out} = F_{ins}(T_{\mathbf{B}}.out, \mathcal{A}_{out})$  // Apply  $F_{ins}$  recursively on  $\mathcal{A}_{out}$  with BSP tree  $T_{\mathbf{B}}.out$  //
16:     $\mathcal{A}_{ins} \leftarrow \mathcal{A}_{ins} \cup \mathcal{A}_{i,out}$  // Include  $\mathcal{A}_{i,out}$  in  $\mathcal{A}_{ins}$  //
17:  end if
18: end if

```

---



---

**Algorithm 3** *Outside clipping function  $F_{out}$ :*  $\mathcal{A}_{out} = F_{out}(T_{\mathbf{B}}, \mathcal{A})$

---

```

1: if  $T_{\mathbf{B}}$  is binary then
2:   if  $T_{\mathbf{B}} = 0$  then
3:      $\mathcal{A}_{out} = \mathcal{A}$  // Initialize output set  $\mathcal{A}_{out}$  with  $\mathcal{A}$  //
4:   end if
5:   if  $T_{\mathbf{B}} = 1$  then
6:      $\mathcal{A}_{out} = \emptyset$  // Initialize output set  $\mathcal{A}_{out}$  //
7:   end if
8: else
9:    $[\mathcal{A}_{ins}, \mathcal{A}_{csd}, \mathcal{A}_{cod}, \mathcal{A}_{out}] = P(T_{\mathbf{B}}.val, \mathcal{A})$  // Partition set  $\mathcal{A}$  with  $T_{\mathbf{B}}.val$  //
10:  if  $\mathcal{A}_{ins} \neq \emptyset$  then
11:     $\mathcal{A}_{o,ins} = F_{out}(T_{\mathbf{B}}.ins, \mathcal{A}_{ins})$  // Apply  $F_{out}$  recursively on set  $\mathcal{A}_{ins}$  with BSP tree  $T_{\mathbf{B}}.ins$  //
12:     $\mathcal{A}_{out} \leftarrow \mathcal{A}_{out} \cup \mathcal{A}_{o,ins}$  // Include  $\mathcal{A}_{o,ins}$  in  $\mathcal{A}_{out}$  //
13:  end if
14:  if  $\mathcal{A}_{out} \neq \emptyset$  then
15:     $\mathcal{A}_{o,out} = F_{out}(T_{\mathbf{B}}.out, \mathcal{A}_{out})$  // Apply  $F_{out}$  recursively on set  $\mathcal{A}_{out}$  with BSP tree  $T_{\mathbf{B}}.out$  //
16:     $\mathcal{A}_{out} \leftarrow \mathcal{A}_{out} \cup \mathcal{A}_{o,out}$  // Include  $\mathcal{A}_{o,out}$  in  $\mathcal{A}_{out}$  //
17:  end if
18: end if

```

---

---

**Algorithm 4** *Coplanar-opposite direction clipping function  $F_{cod}$ :  $\mathcal{A}_{cod} = F_{cod}(T_B, \mathcal{A})$*

---

```

1: if  $T_B$  is a tree (not binary value) then
2:    $[\mathcal{A}_{ins}, \mathcal{A}_{csd}, \mathcal{A}_{cod}, \mathcal{A}_{out}] = P(T_B.val, \mathcal{A})$     // Partition  $\mathcal{A}$  with  $T_B.val$  //
3:   if  $\mathcal{A}_{cod} \neq \emptyset$  then
4:      $\mathcal{A}_{cod} \leftarrow \mathcal{A}_{cod}$     // Initialize output set  $\mathcal{A}_{cod}$  //
5:   end if
6:   if  $\mathcal{A}_{ins} \neq \emptyset$  then
7:      $\mathcal{A}_{c,ins} = F_{cod}(T_B.ins, \mathcal{A}_{ins})$     // Apply  $F_{cod}$  recursively on  $\mathcal{A}_{ins}$  with BSP tree  $T_B.ins$  //
8:      $\mathcal{A}_{cod} \leftarrow \mathcal{A}_{cod} \cup \mathcal{A}_{c,ins}$     // Include  $\mathcal{A}_{c,ins}$  in  $\mathcal{A}_{cod}$  //
9:   end if
10:  if  $\mathcal{A}_{out} \neq \emptyset$  then
11:     $\mathcal{A}_{c,out} = F_{cod}(T_B.out, \mathcal{A}_{out})$     // Apply  $F_{cod}$  recursively on  $\mathcal{A}_{out}$  with BSP tree  $T_B.out$  //
12:     $\mathcal{A}_{cod} \leftarrow \mathcal{A}_{cod} \cup \mathcal{A}_{c,out}$     // Include  $\mathcal{A}_{c,out}$  in  $\mathcal{A}_{cod}$  //
13:  end if
14: end if

```

---

### 3.4 CBIP algorithm stages

As mentioned earlier, CBIP consists of four operational stages. The input of CBIP are the geometric data, contained in IFC and related to three types of building entities: Constructions; Openings; and Volumes. The final output of the CBIP process is the generation of the 2nd-level space boundaries, which are essentially surface pairs, associated with four types of thermal simulation elements.

The input data of CBIP are gathered in the first stage. Their classification to Constructions, Openings and Volumes is performed according to their roles in a thermal simulation process. The first stage is described in Section 3.4.1.

The scope of the second stage, is to generate the B-reps of the building entities, isolated from the first stage. This is accomplished using a process called *Boundary Surface Extraction (BSE)*, described in Section 3.4.2. In some cases, building entities of the Construction type may contain entities of the Opening type, for instance building walls (Constructions) which contain doors or windows. In such cases, the B-reps of these constructions have to be updated by subtracting the B-reps of the opening volumes, they contain. This is performed by the *Opening Construction Subtraction (OCS)* process, described in Section 3.4.2.1.

The boundary surfaces' B-reps, deduced from the second stage, are processed further in the third stage, where the *Common Boundary Intersection (CBI)* process (described in Section 3.4.3) is applied to obtain the Common Boundary (CB) surfaces shared by B-rep pairs. CB surfaces' types are denoted as *Primary types* and are described in Section

**3.4.3.1.** The remaining B-rep surfaces, which are not CB surfaces, are also gathered using the *Remaining Surface Extraction (RSE)* process (described in Section 3.4.3.2), and are marked as *Derived types* of surfaces, which are attached to the environment.

Finally, the  $2^{\text{nd}}$ -level space boundary surfaces, the associated four types of thermal simulation model elements (thermal, shades, openings and air boundaries) and their connectivity information are obtained in the forth stage. This is accomplished by projection of a CB surface (first surface), obtained from the third stage, to the plane of another CB surface (second surface) and vice versa. This process, called *Boundary Intersection Projection (BIP)*, is described in Section 3.4.4.

### 3.4.1 Identification

Although IFC files contain information referring to multiple building geometry entities, only some of them are required for building thermal simulations. These building geometry entities can be classified into three categories depending on their role in thermal building simulations: Constructions; Openings; and Volumes.

Constructions are single- or multi-layer entities, which are involved in thermal simulations in two different ways: (1) directly, by impeding thermal energy flow between building volumes, where the construction layers and their specific thermal properties are taken into account; and (2) indirectly, by blocking sunlight, thus impeding solar heat gains (shading), where their thermal behavior is not considered. Certain IFC classes, which refer to building constructions, belong to the abstract *IfcBuildingElement* class and are indicated by the “CONSTRUCTIONS” dashed rectangle in Figure 3.10.

Openings are building entities described by the *IfcOpeningElement* class. These entities contain doors, windows and skylights, which are generally holes on building Constructions. These entities play important role in thermal simulations, since depending on their state either impede or allow thermal flow. The *IfcOpeningElement* class contains information associated with building openings and belongs to the abstract *IfcElement* class. This class and its relations are indicated by the “OPENINGS” dashed rectangle in Figure 3.10.

Building volumes are entities that exchange thermal energy, which are categorized as follows:

- *Building spaces* refer to the air volumes of rooms or room partitions (separated by air boundaries) with a finite thermal storage capacity. Building spaces interchange thermal energy with other spaces, with the surrounding environment or with the site encompassing the building. Building spaces are defined by the *IfcSpace* class.

- *Building site* refers to the surrounding ground volume, encompassing the building under consideration. It can be assumed that the building site has constant temperature over long periods of time, as it has infinite thermal storage capacity. The building site is defined by the *IfcSite* class.

IFC classes related to Volume entities, belong to the abstract *IfcBuildingSpatialStructureElement* class and are indicated by the “VOLUMES” dashed rectangle in Figure 3.10.

The aforementioned building entities are extracted and their polyhedral boundary surface representations are obtained from their boundary surfaces, as described in Section 3.4.2.

### 3.4.2 Boundary Surface Extraction

In IFC, all the relative building entities, required for the execution of CBIP, are considered products which are related to the abstract *IfcProduct* class. All of the associated products have a 3D shape representation, condition which is met by the Coordination View 2.0 definition, as Figure 3.10 indicates. However, an essential input requirement of CBIP algorithm, is that all of the involved products must have an outward oriented boundary surface geometric representation (B-rep), as described in Section 3.2.1, condition which is not always satisfied. Hence, further processing on some products’ shape representations is required to obtain the desired B-reps. The required data for the generation of the B-reps are contained in the *IfcGeometricRepresentationItem* class, related to the *IfcProductDefinitionShape* subclass of the *IfcProduct* class (see Figure 3.10).

There are five main solid geometrical representations and respective sub-classes of the *IfcGeometricRepresentationItem* class, according to Figure 3.10. The involved geometric representations and the respective IFC classes, contained in Coordination View 2.0 [15], are:

- (1) *Face based surface model representation*, described by *IfcFacedBasedModel* class – According to this representation, the solid of the building entity is described by a set of boundary surfaces “faces” in a 3D space. Such representation need no further processing and can be used directly by the CBIP algorithm, given that the surfaces are correctly oriented.
- (2) *Solid model representation*, described by *IfcSolidModel* class – This class consists of five subclasses referring to the way the solid model is being represented:

- *Manifold solid representation*, described by *IfcManifoldSolidBrep* class – A manifold solid B-rep is a finite, arc-wise connected volume bounded by one or more surfaces, each of which is a connected, oriented, finite, closed 2D-manifold. In this case no further processing is required, since all the points of the boundary surfaces are given.
- *Swept area solid representation*, described by *IfcSweptAreaSolid* class – This class contains solids, either described by a 2D profile being extruded towards a given direction and length (*IfcExtrudedareaSolid*), or revolved around a fixed axes (*IfcRevolvedareaSolid*), or translated along a curve trajectory (*IfcSurfaceCurvedSweptAreaSolid*).

In this case, based on the base profile points, the extrusion direction and the extrusion length, the remaining points of the boundary surfaces are calculated and the respective boundary polygons are obtained. Essentially, the obtained base points are being translated or rotated (depending on the case) following a certain direction, generating the rest boundary surface points.

(3) *Half Space Solid representation*, described by *IfcHalfSpaceSolid* class – Two cases of half-space solid representation can be distinguished:

- *Polygonal bounded half-space* representation – the half space solid is bounded by a base polygon that is extruded at a specific depth and is intersected by a 3D surface (plane or curved surface in general). As in the case of the extruded area solid, the points of the boundary surfaces are obtained from the base points, the extrusion direction and length, and the intersecting plane.
- *Boxed half-space* representation – similarly to the polygonal bounded half-space solid, it is bounded by a bounding box. In this case the points of the bounding box determine the points of the boundary surfaces.

(4) *Boolean result representation*, described by *IfcBooleanResult* class – This class refers to solid geometric representations, which are obtained by performing boolean operations (union, intersection, difference) on solids, represented by the previous classes. Consequently the B-reps of all the involved solids are extracted and the final results are obtained by the clipping functions, applied on the extracted B-reps.

Representations that do not contain the desired B-reps for CBIP, require geometric calculations. These calculations are preformed in the second stage of the algorithm called *Boundary Surface extraction (BSE)* process. The sub-classes, data of which require geometric calculations to obtain the respective B-reps, are indicated by dashed blocks in the Express-G diagram of Figure 3.10.

### 3.4.2.1 Opening Construction Subtraction process

Constructions containing openings are represented in IFC files as solid objects, without considering the openings as holes. Therefore, to obtain a more accurate B-rep of these constructions and to determine the common boundary surfaces among these constructions and their opening volumes (frames of doors, windows, etc.), the polyhedral geometrical representations of the opening volumes must be subtracted from the polyhedral representations of the constructions. Such subtraction is performed by the *Opening Construction Subtraction* (OCS) process, which uses the  $F_{ins}$  and  $F_{out}$  clipping function, given as inputs, the B-rep  $\partial\mathbf{A}$ , the BSP-tree representation  $T_A$  of the construction  $\mathbf{A}$ , the B-rep  $\partial\mathbf{A}_{op}$  and BSP-tree representation  $T_{\mathbf{A}_{op}}$  of the union of its openings  $\partial\mathbf{A}_{op}$ :  $\partial\mathbf{A}_{op} = \partial\mathbf{O}_1 \cup \dots \cup \partial\mathbf{O}_N$  ( $\partial\mathbf{O}_i$  is the B-rep of opening  $i$ ).

OCS process returns a set of boundary polygons (B-rep) of the construction with its openings subtracted. OCS is illustrated in Figure 3.9, for the case of a wall containing a door and a window.

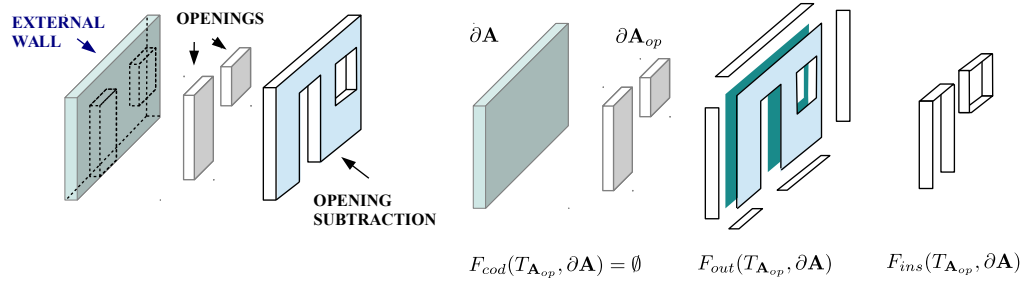


FIGURE 3.9: Illustration of OCS process applied on a rectangular wall containing door and window openings

Mathematically, OCS process is described as follows:

$$\{F_{out}(T_{\mathbf{A}_{op}}, \partial\mathbf{A}), F_{cod}(T_{\mathbf{A}_{op}}, \partial\mathbf{A}), F_{ins}(T_{\mathbf{A}}, \partial\mathbf{A}_{op})^{-1}\} = OCS(T_{\mathbf{A}}, \partial\mathbf{A}, T_{\mathbf{A}_{op}}, \partial\mathbf{A}_{op}) \quad (3.3)$$

The exponent  $-1$ , applied to  $F_{ins}$  function, inverts the ordering of the points to the resulted polygons.

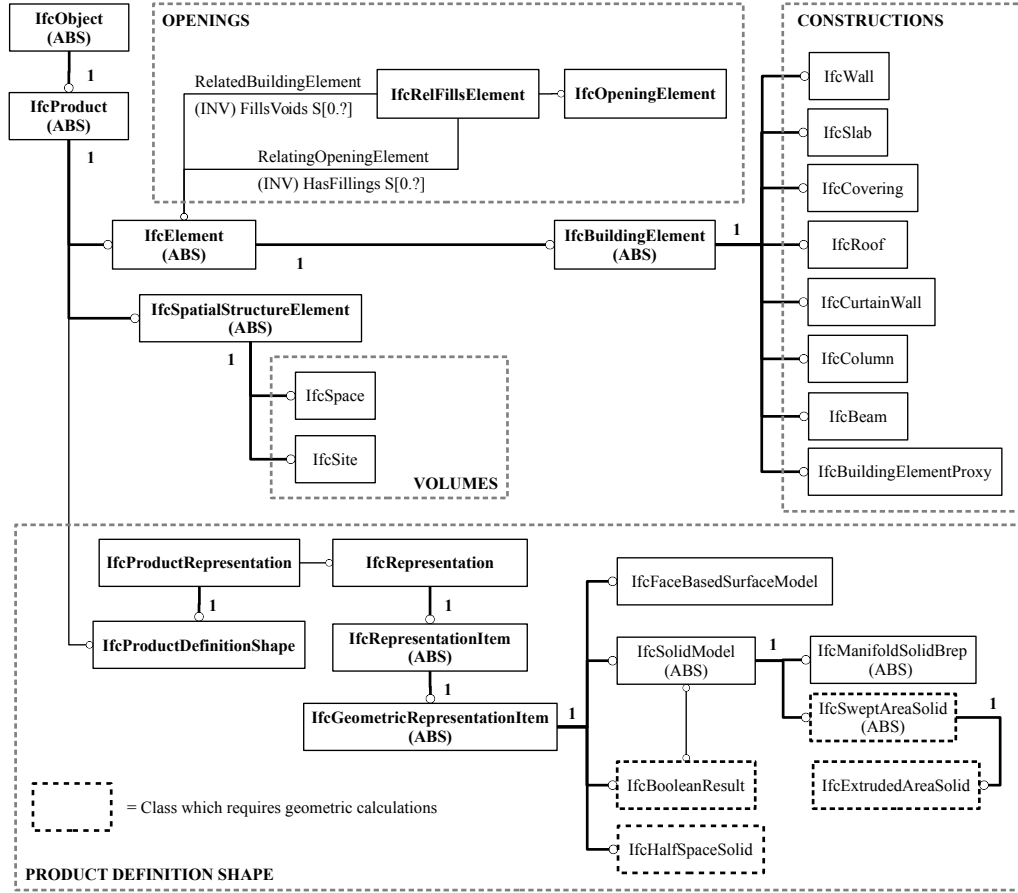


FIGURE 3.10: Part of IFC Coordination View 2.0 EXPRESS-G schema [15] containing the required classes for CBIP algorithm. Sub-classes which require calculations (performed by BSE) are indicated by dashed blocks.

### 3.4.3 Common Boundary Intersection

The Common Boundary Intersection (CBI) process determines the CB surfaces shared by two polyhedrons **A** and **B** representing two building entities. There are two types of CB surfaces: the *Primary type*, described in Section 3.4.3.1, and the *Derived type*, described in Section 3.4.3.2. In a nutshell, CBI is applied on the pairs  $\partial\mathbf{A}$  and  $\partial\mathbf{B}$  of polyhedrons **A** and **B**, and outputs the set of CB surfaces  $\mathcal{CB}_{AB}$ , shared by the two polyhedrons. After the opening volumes subtraction from their constructions, the CB surface set  $\mathcal{CB}_{AB}$  is obtained by applying the  $F_{cod}$  clipping function on  $\partial\mathbf{A}$  using the BSP-tree  $T_B$ . CBI process is expressed mathematically by Equation 3.4.

$$\mathcal{CB}_{AB} = F_{cod}(T_B, \partial\mathbf{A}) \quad (3.4)$$

### 3.4.3.1 Primary types of common boundary surfaces

After the OCS process, B-reps of the resulting building constructions (obtained from the OCS process) are forwarded to the CBI process, from where the following five primary types of CB surfaces, depicted in Figure 3.11, are derived:

- (1) *Construction - Construction (C-C)* CB surfaces. C-C CB surfaces are surfaces where constructions (walls, slabs, roofs, etc.) touch other constructions. Although C-C surfaces are not used directly as elements of thermal models, they contribute towards specifying the construction - environment boundaries.
- (2) *Construction - Volume (C-V)* CB surfaces. Examples of C-V CB surfaces include surfaces shared by walls and spaces, slabs and spaces, or slab and sites.
- (3) *Volume - Volume (V-V)* CB surfaces. Examples of V-V CB surfaces include boundaries between building spaces and boundaries between building spaces and building site. Such boundaries do not impede the thermal energy flow among the building volumes.
- (4) *Opening - Construction (O-C)* CB surfaces. Examples of O-C CB boundaries include the door and window frames and thresholds. Although such boundaries do not participate directly in the calculation of the thermal model elements, they contribute towards deriving the Opening-Environment (O-E) surfaces.
- (5) *Opening - Volume (O-V)* CB surfaces. O-V CB boundaries include surfaces shared by openings and spaces, or openings and site. These surfaces contribute to derive the opening thermal simulation elements.

### 3.4.3.2 Derived types of surfaces (Environment surfaces)

After subtracting the Primary types of CB surfaces from the B-reps of the building entities, the remaining surfaces define surfaces attached to the environment. These surfaces are obtained by the *Remaining Surfaces Extraction (RSE)* process (see Algorithm 5).

Depending on the building entity's type (Construction, Opening or Volume), three sets of Derived (or environment) surfaces are defined (examples displayed in Figure 3.11):

- (1) *Construction - Environment (C-E)* CB surfaces. Examples of such surfaces include the external surfaces of a wall or a slab (balcony), attached to the outside air.
- (2) *Opening - Environment (O-E)* CB surfaces. Examples of O-E surfaces include the external surfaces of doors and windows, attached to the outside air.

- (3) *Volume - Environment (V-E)* CB surfaces. Examples of such surfaces include the external surfaces of spaces (flats), attached to the outside air.

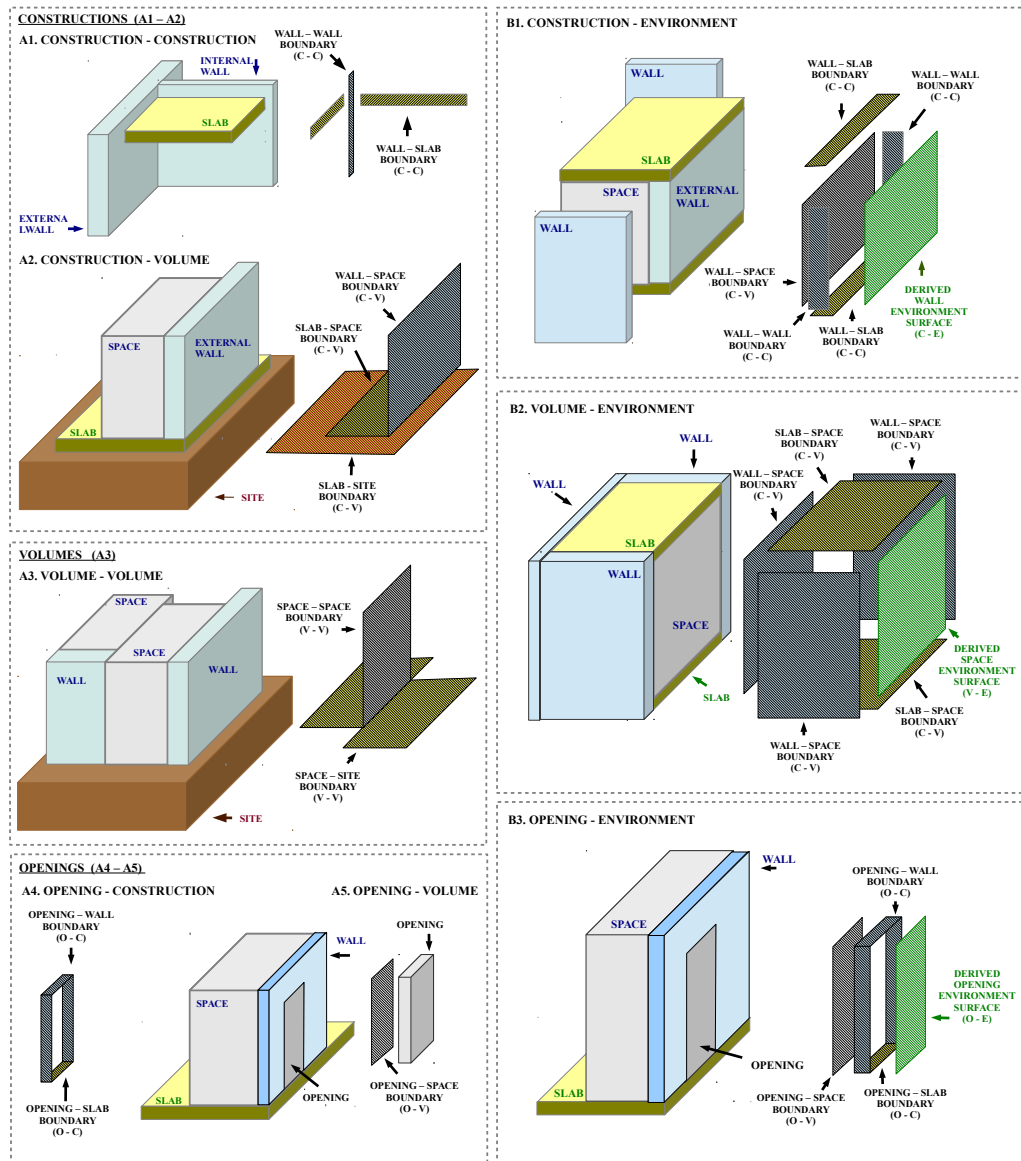


FIGURE 3.11: Primary types of common boundary surfaces referring to building constructions (A cases) and Derived environment surfaces (B cases)

**Algorithm 5** *Remaining Surface Extraction (RSE) process*


---

```

1: A                                     // Polyhedron under consideration //
2:  $\partial\mathbf{A} = \{A_1, A_2, \dots, A_N\}$     // Polyhedral boundary of the polyhedron under consideration //
3:  $\mathcal{CB} = \{CB_1, \dots, CB_M\}$       // Set of common boundaries between A and other polyhedrons //
4:  $\mathcal{R} = \emptyset$                      // Initialization of remaining surface set //

5: for  $i = 1 : N$  do
6:   for  $j = 1 : M$  do
7:     if  $A_i$  and  $CB_j$  are coplanar. then
8:        $A_i \leftarrow A_i - CB_j$         // The boundary surface  $CB_j$  is subtracted from  $A_i$  //
9:     end if
10:  end for
11:  if  $A_i \neq \emptyset$  then
12:     $\mathcal{R} \leftarrow \mathcal{R} \cup A_i$       //  $A_i$  is added to the remaining surfaces set  $\mathcal{R}$  //
13:  end if
14: end for

```

---

**3.4.4 Boundary Intersection Projection**

The CB surfaces are forwarded to the *Boundary Intersection Projection (BIP)* process to generate the required geometry elements of a BEP model. These elements, are essentially *surface pairs*, which include thermal material properties and other relative information.

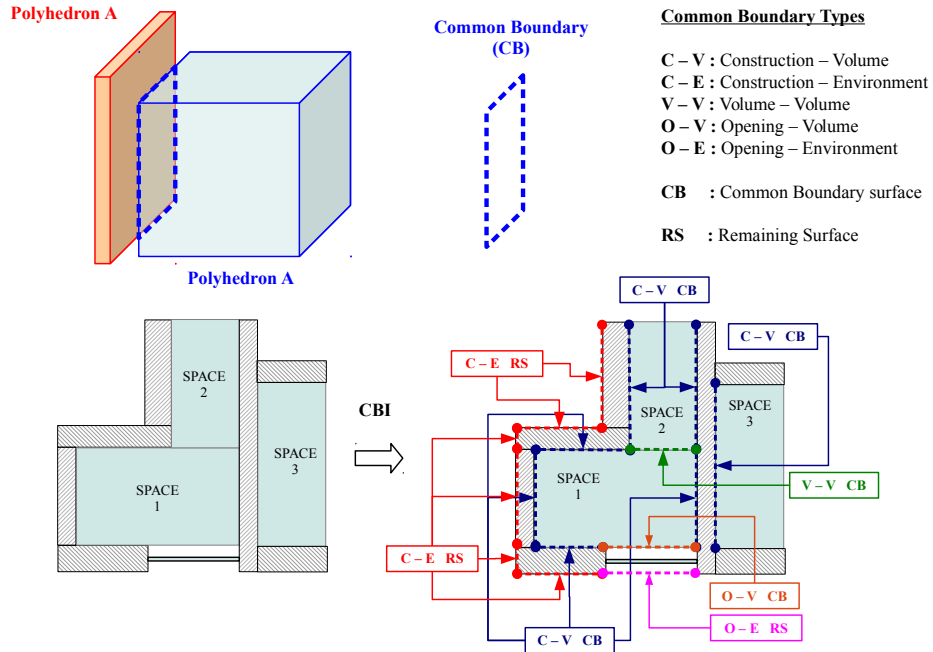


FIGURE 3.12: Geometrical illustration of CBI process on two polyhedrons (Top) – Plan view of the resulted Common Boundaries (CB) for a three space building (Bottom)

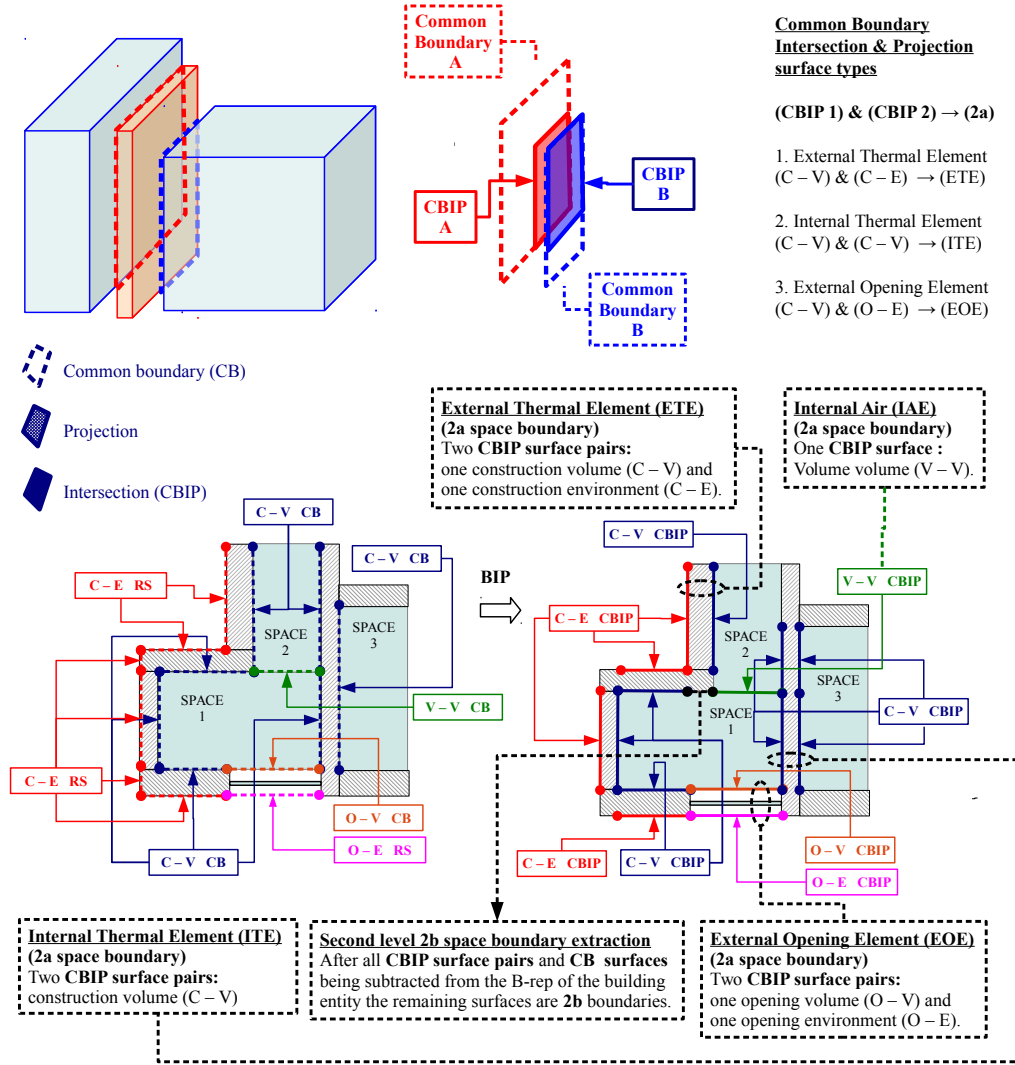


FIGURE 3.13: Geometrical illustration of BIP process on two polyhedrons (Top) – Illustration of: C-V and O-V CBIP surface pairs, V-V Common Boundaries (CB) and extracted 2b space boundary surface types, for a building space (Bottom)

The BIP process can be described by two geometrical operations: (1) the *projection* of one of the common boundaries on the plane of another; and (2) the *intersection* of the projection with the other element.

The results of CBI and BIP operations for a naive example of three spaces' floor plan, are depicted in Figures 3.12 and 3.13, respectively.

The projections of BIP process are applied to four types of CB, derived from the CBI stage: C-V (Construction - Volume), C-E (Construction - Environment), O-V (Opening - Volume), O-E (Opening - Environment). In case a C-V, C-E, O-V or O-E common boundary is not projected into a C-V, C-E, O-V or O-E common boundary, it remains as a CB and is associated with a specific building entity (wall, slab, etc.). Such a case is

depicted in Figure 3.13 for a V-V CB surface of the examined space 1, which is also common boundary surface of space 2. Other possible CB surfaces include common boundary surfaces of the C-C type which appear in boundary surface of building constructions.

#### 3.4.4.1 Second level boundaries of type 2a

After the completion of BIP process on all building elements of interest, CBIP surface pairs are extracted. These CBIP surface are related to the 2nd-level space boundaries of type 2a (as defined in [8]), and are associated to the following eight types of simulation model elements:

(1) *External Thermal Elements (ETE).*

External thermal elements are obtained by applying BIP on a Construction - Volume (C-V) / Construction - Environment (C-E) surface pair referring to the same building entity. A common example of such element is an external wall illustrated in case A of Figure 4.2.

(2) *Internal Thermal Elements (ITE).*

Internal thermal elements are extracted using BIP on two Construction - Volume (C-V) CB surfaces, which refer to the same building entity. Examples include internal wall's space boundary surfaces, as displayed in case B of Figure 4.2 and slab - space boundary pairs.

(3) *External Shading Elements (ESH).* These elements are obtained by applying BIP on Construction - Environment (C-E) CB surfaces, which refer to the same building entity (see Figure 4.2 case C).

(4) *Internal Shading Elements (ISH).*

Internal Shading Elements refer to construction building entities which cause shading effects inside building spaces. They are obtained by applying the BIP process on Construction - Volume (C-V) CB surfaces. Examples include recesses of building spaces caused by internal walls or slabs (see case D of Figure 4.2).

(5) *External opening elements (EOE).*

External Opening Elements refer to surface pairs of building entities, that allow airflow between the environment and the building spaces. EOE are obtained by applying the BIP process on an Opening - Volume (O-V) CB surface and its Opening - Environment (O-E) CB surface counter part (see case E of Figure 4.2).

(6) *Internal Opening Element (IOE).*

Similarly to the external opening elements, internal opening elements are represented by surface pairs of building entities, that allow airflow among building spaces. IOE are obtained using BIP on an Opening - Volume (O-E) CBIP surface pair referring to the same element (see case F Figure 4.2).

(7) *External Air Element (EAE).*

External Air Elements are obtained directly from the external air boundaries without requiring BIP processing.

(8) *Internal air element (IAE).*

Internal Air Elements are obtained directly from the internal air boundaries without requiring BIP processing.

#### 3.4.4.2 Second level boundaries of type 2b and 2c

Apart from the second level boundaries of type 2a, those of type 2b, 2c and 2d [8], are also extracted. These special cases of 2nd-level space boundaries can be ignored or be entered as adiabatic surfaces in a thermal simulation model. The extraction process of 2b, 2c and 2d space boundary types is similar to the RSE process, performed by Algorithm 5. Here, the polyhedron under consideration  $\mathbf{A}$ , is assigned to the B-rep of a building space, and not to a construction,

Additionally, the set  $\mathcal{CB}$ , contains all the associated obtained CBIP surface pairs or CB surfaces (C-V, O-V CBIPs or O-E, V-V CBs), related to this space (volume) which is represented by the polyhedron  $\mathbf{A}$ . After executing a process similar to 5 the resulting set  $\mathcal{R}$ , if its is not empty, will contain the second level space boundaries of type 2b and 2c, associated with the space volume under consideration.

An example of a 2b space boundary extraction, after all the CBIP and CB surfaces referring to a single space, are collected, is illustrated in the bottom part of Figure 3.13.

#### 3.4.4.3 Connectivity information

All of the previous types of elements are associated with certain connectivity information which is also required in thermal simulations. CBIP provides this information in the form of matrices of different number of entries according to Table 3.1.

TABLE 3.1: Connectivity information of thermal elements.  
 Ci. = Construction index, Isi = Internal space index, Ei = Environment index

<i>Element</i>	<i>Connectivity information</i>
ETE	(Ci) / (Isi) / (Ei <sup>1</sup> )
ITE	(Ci) / (1 <sup>st</sup> Isi) / (2 <sup>nd</sup> Isi)
ESE	n/a
ISE	(Isi)
EOE	(Isi) / (Ei <sup>2</sup> )
IOE	(1 <sup>st</sup> Isi) / (2 <sup>nd</sup> Isi)
EAE	(Isi)
IAE	(1 <sup>st</sup> Isi) / (2 <sup>nd</sup> Isi)

### 3.5 Design requirements

In order to ensure the correct execution of CBIP algorithm certain design requirements must be satisfied.

#### 3.5.1 Building site and spaces

The building entities which are associated with the operation of CBIP must contain at least one element referring to the building site and at least one element referring to a building space which is going to be thermally studied. Such requirements are not met by the Coordination View 2.0 [15] model view definition.

The building site acts as a reference level in thermal simulations attaining the ground temperature, which is considerably different than the air or building interior temperatures. Consequently, its presence and relative location to other building elements is of paramount importance. On the other hand building spaces are associated with simulation output values (temperature, humidity, etc.) and therefore the presence and the geometrical definition of at least one building space is prerequisite.

#### 3.5.2 Curved solids

The geometric information of any curved building entity must be exported in the IFC file considering a polyhedral approximation of the entity. This approximation must

have its boundary surfaces oriented according to the right hand outward normal rule, as explained in Section 3.2.1. Such a requirement can be set by the exporting program.

## 3.6 Design recommendations

Apart from the previous mandatory site and building space requirements, there are some additional design recommendations, compliance of which guarantees accuracy of CBIP results. These recommendations are related to certain scenarios.

### 3.6.1 Nonzero volume intersections

A nonzero volume intersection occurs when two or more building entities (wall, slab, space, etc) share a common nonzero volume, meaning that their solid geometric representations are intersected. Such cases can be identified using a model checking software such as Solibri [48]. These cases do not impede the execution of CBIP, but affect the accuracy of its results. They can be corrected manually or automatically by using the algorithms of [57]. An example of a nonzero volume intersection between a wall and a slab is displayed in the images of Case A1 (inaccurate) and Case A2 (accurate) in Figure 3.14.

### 3.6.2 Space-environment surfaces

The accuracy of CBIP results is also affected by the presence of space-environment surfaces associated with internal surfaces. Space environment surfaces are derived surfaces of V-E type (see section 3.4.3.2), which define areas where a building space is not attached to any other building entity. These surfaces occur when an internal building space is not defined correctly, leaving small undefined space gaps between the space and surrounding building entities. In such cases the building spaces should be redefined correctly by redesign, or corrected using a space correction algorithm [57]. Examples of space-environment surfaces related to an incorrect space definition is displayed in the images of Case B1 (inaccurate) and Case B2 (accurate) of Figure 3.14.

### 3.6.3 Curtain walls

If a curtain wall is present, it should always be contained inside an opening volume — a volume with surface area equal to the surface area of the curtain wall and thickness equal

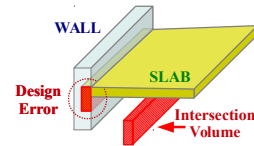
to the thickness of the wall it is attached to. This requirement is displayed graphically in the images of Case C1 (inaccurate) and Case C2 (accurate) of Figure 3.14.

### 3.6.4 Suspended ceilings

If a suspended ceiling is present, the space volume beneath it should extend to the floor (or the roof if the space is in the last level) above it. This requirement is displayed graphically by the images of Case D1 (inaccurate) and Case D2 (accurate) of Figure 3.14. Otherwise, an additional space volume should be defined between the suspended ceiling and the floor, or roof above it.

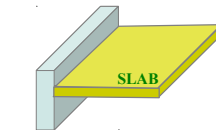
#### DESIGN RECOMMENDATIONS

Case A1 (inaccurate)



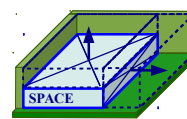
A building slab intersects with a building wall. The volume of the intersection is nonzero.

Case A2 (accurate)



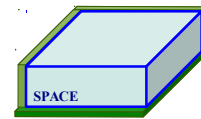
The building slab does not intersect with the adjacent wall. There is a common surface between the wall and slab.

Case B1 (inaccurate)



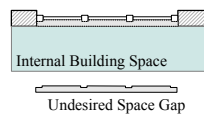
Space is incorrectly defined leaving undefined volume gaps indicated by the normal vectors

Case B2 (accurate)



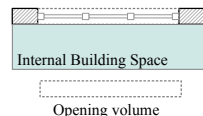
Space is correctly defined With no space-environment surfaces or undefined volume gaps.

Case C1 (inaccurate)



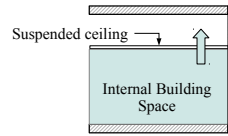
If no opening volume is defined in a curtain wall, an undesired space gap between the space volume and the plates/beams of the curtain wall exists.

Case C2 (accurate)



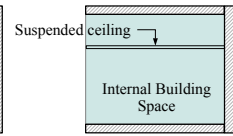
If the opening volume of a curtain wall is defined no undesired space gap exists.

Case D1 (inaccurate)



Space volume reaches the bottom surface of the suspended ceiling

Case D2 (accurate)



After the space volume expansion the volume reaches the floor/roof of the level above.

FIGURE 3.14: Design requirements for curtain walls (Left) and suspended ceilings (Right).

### 3.6.5 Orientation of boundary surfaces

To ensure accuracy in the results of CBIP, all the boundary surface polygons of the geometric representations of the involved building entities should have a right-hand outward normal orientation, as described in Section 3.2.1. However, not all IFC exporting software satisfy such requirement, as this requirement is not necessary for building rendering purposes. Therefore, an outward surface normal vector check of the involved polyhedrons' boundary surfaces and corrections of them, where necessary, are required.

### 3.7 Demonstration Examples

CBIP is implemented on the geometrical three dimensional data of CARTIF and TUC buildings, which are queried from the respective IFC files. The results obtained from CBIP are surface pairs which refer to: Internal air elements (part 2 in Figures 3.15 and 3.16), External thermal elements (part 3 in Figures 3.15 and 3.16), Internal thermal elements (part 4 in Figures 3.15 and 3.16), Shading elements (part 5 in Figures 3.15 and 3.16) and Opening elements (part 6 in Figures 3.15 and 3.16).

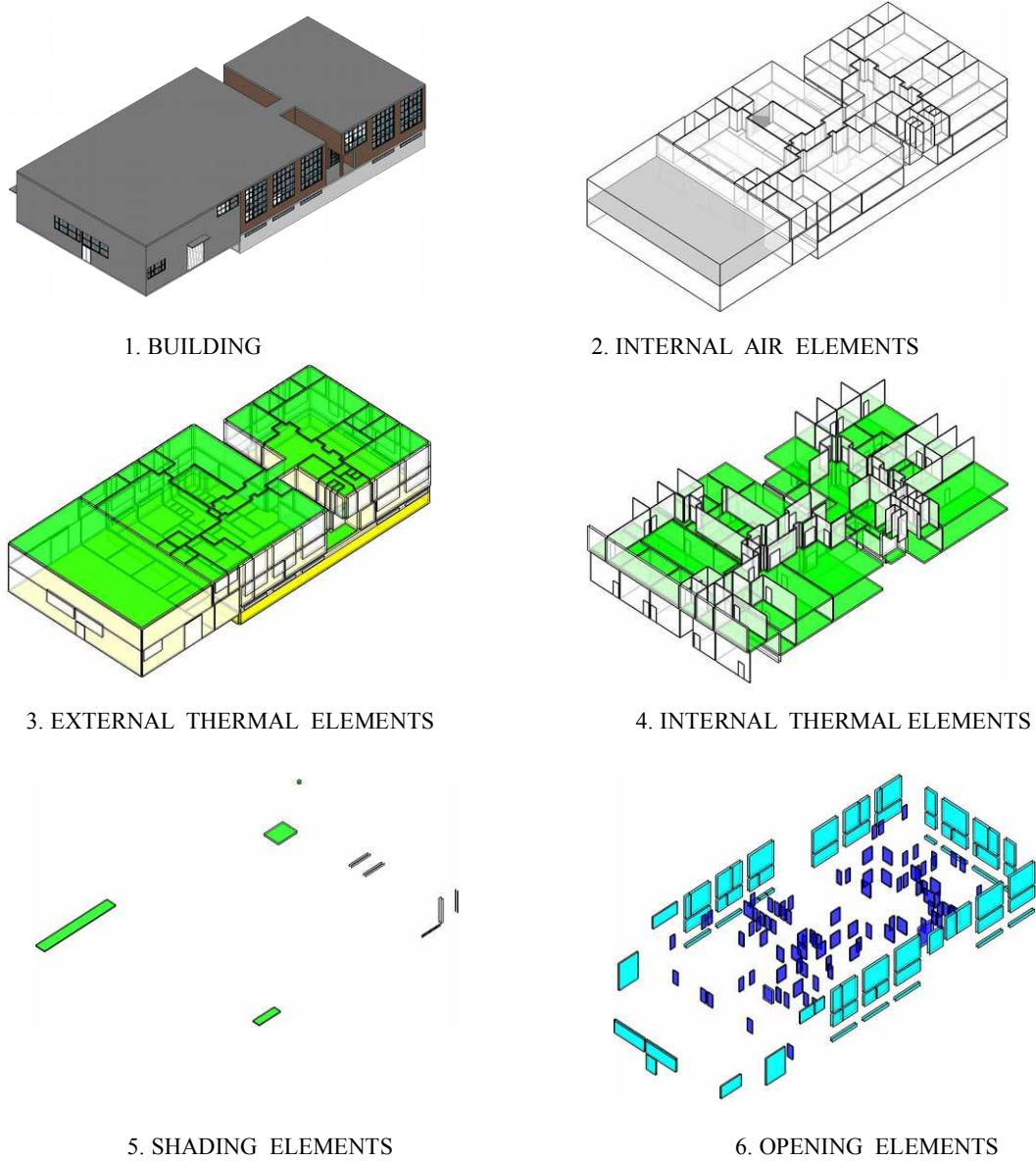


FIGURE 3.15: Categories of surface pairs of CARTIF building, obtained from CBIP.

In both building cases the geometrical data of the obtained surface pairs are either collected in single xml file <SB.xml> (see Appendix A) or used to update the IFC data (see Section 4.2), which can be used in order to create the BEP model for any simulation software (as presented in Chapter 3 for EnergyPlus and TRNSYS).

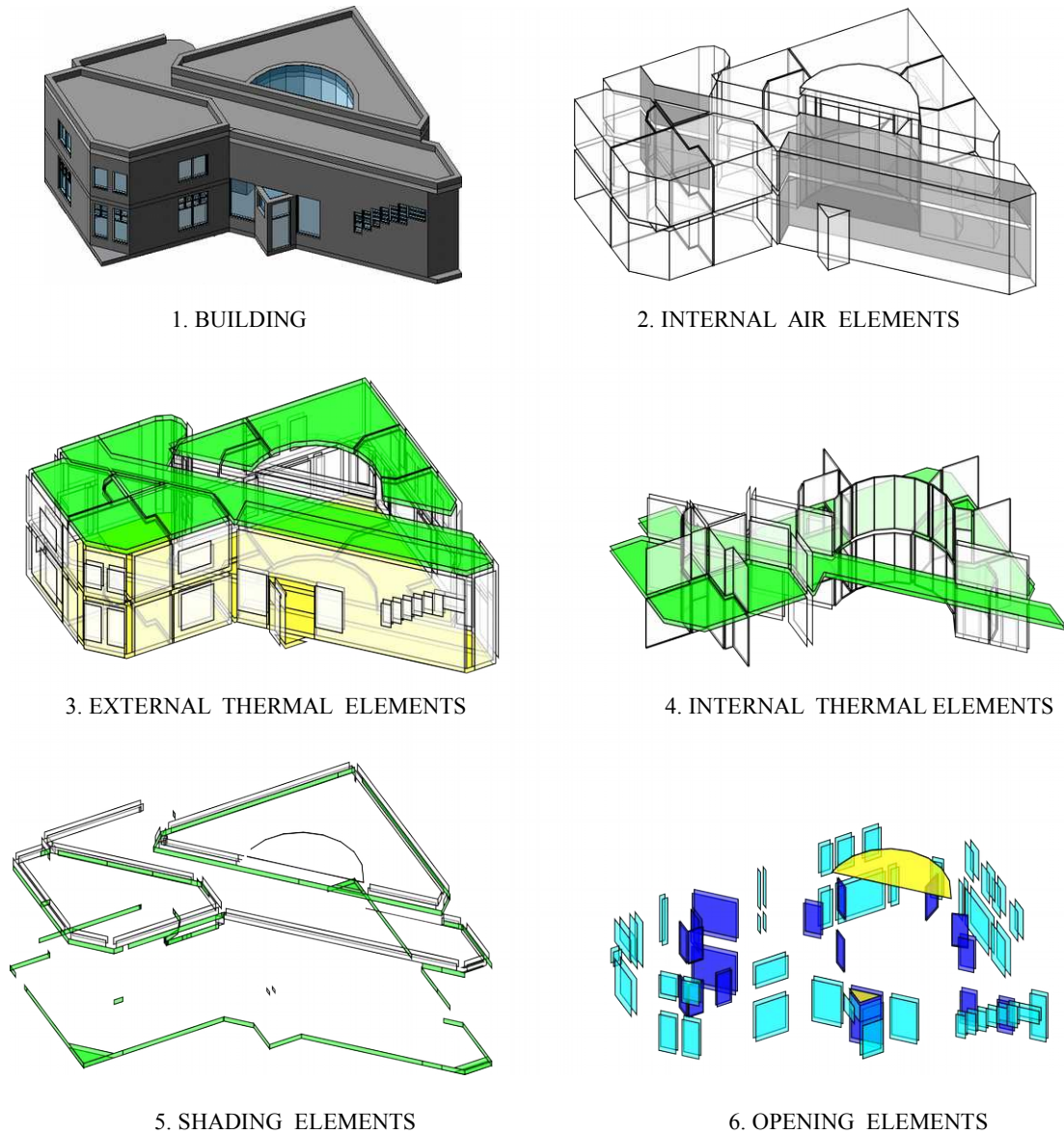


FIGURE 3.16: Categories of surface pairs of TUC building, obtained from CBIP.

## Chapter 4

# BEP Model Generation

In this Chapter, a transformation process that converts the data obtained from the CBIP algorithm, to an EnergyPlus and/or TRNSYS input file, is presented. After a short overview of the data requirements to properly define an BEP simulation input file, concerning the buildings envelope, data transformation rules embedded in the transformation process are presented along with an illustrative example.

### 4.1 Data Requirements

Towards numbering data requirements to properly define an BEP simulation input file, concerning the building's envelope, a brief introduction to the energy transfer mechanisms inside and outside of the building will be presented here, focusing on EnergyPlus simulation engine.

In order to estimate the zone air temperature, the energy balance for each zone  $z$  is formulated:

$$\begin{aligned} C_{a_z} \frac{dT_{a_z}}{dt} = & \sum_{i=1}^{N_L} q_i + \sum_{i=1}^{N_{surf}} h_{c_i} A_i (T_{s_i}^{\text{in}} - T_{a_z}) \\ & + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p (T_{a_i} - T_{a_z}) + \dot{m}_{\infty} C_p (T_{\infty} - T_{a_z}) + q_{sys}; \end{aligned} \quad (4.1)$$

here  $\sum_{i=1}^{N_{si}} q_i$  is the sum of the convective internal loads;  $\sum_{i=1}^{N_{surfaces}} h_{c_i} A_i (T_{s_i}^{\text{in}} - T_{a_z})$  is the convective heat transfer from the zone surfaces;  $\sum_{i=1}^{N_{zones}} \dot{m}_i C_p (T_{a_i} - T_{a_z})$  is the heat transfer due to inter-zone air mixing;  $\dot{m}_{\infty} C_p (T_{\infty} - T_{a_z})$  is the heat transfer due to infiltration of outside air;  $q_{sys} = \dot{m}_{sys} C_p T_{sup}$  are the air systems outputs; and  $C_{a_z} \frac{dT_{a_z}}{dt}$  is the energy stored in zone  $z$  air.

To obtain the discretized equations we assume that the design process is to start at time  $t_0$  and the prediction horizon is  $T$ . We select a set of  $N - 1$  points  $\{t_1, \dots, t_{N-1}\}$  with  $t_i \in I \equiv (t_0, t_0 + T)$ ,  $i \in \{1, \dots, N-1\}$  such that  $t_0 < t_1 < t_2 < \dots < t_{N-1} < t_N \equiv t_0 + T$ . Therefore a partition  $\mathcal{I}$  of  $I$  in  $N$  intervals  $I_k = (t_{k-1}, t_k)$ ,  $k \in 1, 2, \dots, N$  is introduced. A first-order discretization scheme can be utilized to obtain the discrete equations:

$$\frac{dT_{a_z}}{dt} = (\delta t)^{-1}(T_{a_z}^t - T_{a_z}^{t-\delta t}) + O(\delta t). \quad (4.2)$$

By combining (4.2) and (4.1) the formula for updating the zone mean air temperature is expressed:

$$\begin{aligned} T_{a_z,t} = & \left( \sum_{i=1}^{N_{si}} q_i + \dot{m}_{sys} C_p T_{sup} + (C_{a_z} \frac{T_{a_z}}{\delta t} + \sum_{i=1}^{N_{surfaces}} h_{c_i} A_i (T_{s_i}^{in} - T_{a_z}) \right. \\ & \left. + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p (T_{a_i} - T_{a_z}) + \dot{m}_{\infty} C_p (T_{\infty} - T_{a_z}) \right)^{t-\delta t} \\ & / \left( C_{a_z} + \sum_{i=1}^{N_{surfaces}} h_{c_i} A_i + \sum_{i=1}^{N_{zones}} \dot{m}_i C_p + \dot{m}_{\infty} \right). \end{aligned} \quad (4.3)$$

To integrate the effects of the HVAC systems a Predictor/ Corrector approach is used: first the air system load needed to reach the desired temperature is computed (predictor); then the air system to determine its actual capability is simulated (corrector); and finally, the zone air heat balance is computed to obtain the actual zone mean air temperature. To predict the air system load needed to reach the desired temperature (predictor) at each timestep  $k$ , the Heat Balance method [3] is used, modeling four distinct processes: a) inside-face heat balance; b) outside-face heat balance; c) quasi-steady air heat balance; and d) wall conduction process.

More analytically, the heat balance on the inside face can be written as follows:

$$q_{lw}^{in} + q_{sw}^{in} + q_{sol}^{in} + q_f^{in} + q_c^{in} = 0, \quad (4.4)$$

where  $q_{lw}^{in}$  is the sum of the longwave radiant exchange transfer between zone surfaces and the longwave radiation transfer from equipment in zone and  $q_{sw}^{in}$ ,  $q_{sol}^{in}$ ,  $q_f^{in}$  and  $q_c^{in}$  refer to the shortwave radiation transfer to surface from lights, the transmitted solar radiation flux absorbed at surface, the conduction transfer through the wall and the convective heat transfer to the zone air, respectively.

Similarly, the heat balance on the outside face is expressed as follows:

$$q_{lw}^{out} + q_{sol}^{out} - q_f^{out} + q_c^{out} = 0, \quad (4.5)$$

where  $q_{lw}^{out}$  is the longwave radiant exchange with the air and surroundings and  $q_{sol}^{out}$ ,  $q_f^{out}$ ,  $q_c^{out}$  are the absorbed solar shortwave radiation flux, the conduction transfer into the wall and the convective transfer to the outside air, respectively. Here, the conduction transfer term is negative since the direction from outside to inside of the wall is considered to be positive. Note that every thermal simulation engine incorporates the appropriate mathematical models to quantify each term of (4.4) and (4.5).

If the air capacitance in the zone is neglected and the air heat balance is formulated as a quasi-steady balance in each time step, four factors contribute to the air heat balance as follows:

$$q_c + q_{CE} + q_{IV} + q_{sys} = 0, \quad (4.6)$$

where  $q_c$  is the convective heat transfer from surfaces and  $q_{CE}$ ,  $q_{IV}$ ,  $q_{sys}$  refer to the convective parts of internal loads, the sensible load caused by infiltration and ventilation air and the heat transfer from Heating Ventilating and Air Conditioning (HVAC) system, respectively.

Building thermal simulation engines provide solution methods to formulate the conduction process, such as the finite element, transform, finite-difference and time series methods, of which the last two are widely used. The less computationally expensive time series method uses Conduction Transfer Functions (CTFs) which relate conductive heat fluxes to current, past surface temperatures and past heat fluxes. CTFs are constants that only need to be determined once for each construction type, with methods to pre-calculate CTFs of building constructions including the Laplace transform method [33], the frequency-domain regression method [103] and the state-space method [91] which is used in EnergyPlus.

When CTFs are used, the inside heat flux of surface  $i$  at  $k$  is:

$$\begin{aligned} q_{f,i,k}^{in} = & -Z_{i,0}T_{s,i,k}^{in} - \sum_{j=1}^{N_z} Z_{i,j}T_{s,i,k-j}^{in} + Y_{i,0}T_{s,i,k}^{out} \\ & + \sum_{j=1}^{N_y} Y_{i,j}T_{s,i,k-j}^{out} + \sum_{j=1}^{N_q} \Phi_{i,j}q_{f,i,k-j}^{in}, \end{aligned} \quad (4.7)$$

while the outside heat flux is:

$$\begin{aligned} q_{f,i,k}^{out} = & -Y_{i,0}T_{s,i,k}^{in} - \sum_{j=1}^{N_y} Y_{i,j}T_{s,i,k-j}^{in} + X_{i,0}T_{s,i,k}^{out} \\ & + \sum_{j=1}^{N_x} X_{i,j}T_{s,i,k-j}^{out} + \sum_{j=1}^{N_q} \Phi_{i,j}q_{f,i,k-j}^{out}. \end{aligned} \quad (4.8)$$

Here,  $X_{i,j}$ ,  $Y_{i,j}$ ,  $Z_{i,j}$ ,  $\Phi_{i,j}$  refer to the surface-to-surface exterior, surface-to-surface cross, surface-to-surface interior and flux CTF coefficients respectively. Moreover,  $N_x$ ,  $N_y$ ,  $N_z$ ,  $N_q$  indicate the number of exterior, cross, interior and flux history terms respectively. All of the previous coefficients depend on the thermal properties of the material layers of the construction element, are evaluated in EnergyPlus using Seem's state space method described in [91].

Subsequently, (4.4) and (4.7) are combined and solved for  $T_s^{\text{in}}$  at each inside wall's face  $i$  for each  $k$ , while (4.5) and (4.8) are combined and solved for  $T_s^{\text{out}}$  at each outside wall's face  $i$  for each  $k$ :

$$T_{s_{i,k}}^{\text{in}} = \left( T_{s_{i,k}}^{\text{out}} Y_{i,0} + \sum_{j=1}^{N_z} Y_{i,j} T_{s_{i,k-j}}^{\text{out}} - \sum_{j=1}^{N_z} Z_{i,j} T_{s_{i,k-j}}^{\text{in}} + \sum_{j=1}^{N_q} \Phi_{i,j} q_{f_{i,k-j}}^{\text{in}} + T_{a_k} h_{c_{i,k}}^{\text{in}} + q_{lw}^{\text{in}} + q_{sw}^{\text{in}} + q_{sol}^{\text{in}} \right) / (Z_{i,0} + h_{c_{i,k}}^{\text{in}}); \quad (4.9)$$

$$T_{s_{i,k}}^{\text{out}} = \left( T_{s_{i,k}}^{\text{in}} Y_{i,0} + \sum_{j=1}^{N_z} Y_{i,j} T_{s_{i,k-j}}^{\text{in}} - \sum_{j=1}^{N_z} Z_{i,j} T_{s_{i,k-j}}^{\text{out}} + \sum_{j=1}^{N_q} \Phi_{i,j} q_{f_{i,k-j}}^{\text{out}} + T_{\infty} h_{c_{i,k}}^{\text{out}} + q_{lw}^{\text{out}} + q_{sol}^{\text{out}} \right) / (Z_{i,0} + h_{c_{i,k}}^{\text{out}}). \quad (4.10)$$

Given that the zone air temperature is equal to the set-point temperature, (4.9) and (4.10) are solved simultaneously to calculate the surface temperatures at each  $k$ . Then, the heating/cooling demand is estimated using the air heat balance equation:

$$q_{sys_k} = \sum_{i=1}^{N_s} A_i h_{c_{i,k}}^{\text{in}} (T_{s_{i,k}}^{\text{in}} - T_{a_k}) + q_{CE} + q_{IV}. \quad (4.11)$$

Having this quantity as a demand, the air system is simulated and its actual supply capability at the time of the simulation is determined. According to the actual supply air system capacity estimated by the HVAC system's simulation, (4.11) is solved for  $T_{a_k}$  and used to calculate the resulting zone temperature.

Hence, the following steps summarize the Predictor/ Corrector process:

1. The air system energy required to balance the equation (4.11) is estimated, given that the zone air temperature is equal to the set-point temperature  $T_{a_k}$  (defined by a zone thermostat).

2. Having this quantity as a demand, the air system is simulated and its actual supply capability at the time of the simulation is determined.
3. Equation 4.3 is used to calculate the resulting zone temperature, where  $\dot{m}_{sys}C_pT_{sup}$  is equal to the actual air system capability estimated by step 2.

From a building's envelope thermal simulation perspective, the effect of the HVAC system is neglected and as such the thermal simulation model should incorporate all the data required for individual calculations of each part of Equations 4.4 and 4.5. According to [63] and the Heat balance method shortly presented above, we infer that the following data are required:

1. opaque surfaces' thermal properties;
2. reflectance properties of surrounding surfaces;
3. transparent surfaces' glazing and thermal properties;
4. surface type (internal or external);
5. surface area;
6. convection coefficients (inside and outside);
7. 3D position of surfaces (to calculate correctly the solar radiation effects);
8. normal vector of each surface (to determine the inner most and the outer most layer of surface' construction);
9. relationship between surfaces and spaces (to define the flow paths of each zone);
10. relationship between materials and the surfaces (to define the construction of a building element);
11. relationship between two opposite surfaces (for internal heat transfer);
12. relationship between parent and child surfaces (for openings hosted to a wall).

## 4.2 IFC Data Refinement

After the CBIP's geometric operations, the IFC database can be updated using the results of the algorithm. More precisely, the surface pairs obtained by CBIP can populate the *IfcRelSpaceBoundary2ndLevel* classes, as displayed in Figure 4.1.

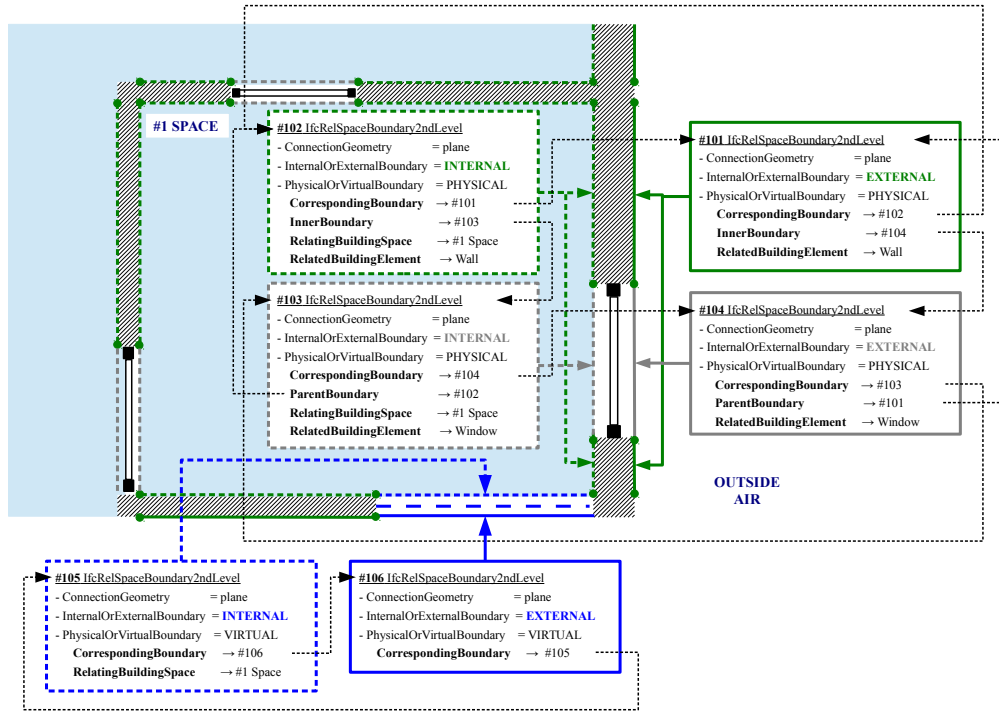


FIGURE 4.1: Example of IFC4 classes referring to second level boundaries and their relating building entities

Each boundary surface's geometry (surface polygon) is pointed by the *ConnectionGeometry* item to a respective geometric representation. The boundary's location, with respect to other building entities, is indicated by the *InternalOrExternal* item, which can potentially receive the following values:

- *INTERNAL*, if the boundary surface is attached to an internal building space (Boundaries #102, #103 and #105 in Figure 4.1);
- *EXTERNAL*, if the boundary surface is attached to the outside air environment (Boundaries #101, #104 and #106 in Figure 4.1);
- *EXTERNAL\_EARTH*, if the boundary surface is attached to the ground;
- *EXTERNAL\_WATER*, if the boundary surface is attached to water;
- *EXTERNAL\_FIRE*, if the boundary surface is attached to another building; and
- *NOTDEFINED*, if none of the previous cases holds.

If the boundary surface refers to a building construction (wall, slab, etc.), the item *PhysicalOrVirtual* receives the *PHYSICAL* value (boundaries #101, #102, #103, #104

in Figure 4.1); if it refers to a surface separating building spaces, the *PhysicalOrVirtual* receives the *VIRTUAL* value (boundaries #105 and #106 in Figure 4.1); otherwise, *PhysicalOrVirtual* becomes *NOTDEFINED*.

CBIP's surface pairs are defined by the *CorrespondingBoundary* attribute. For example, the external wall of Figure 4.1 contains a thermal element, defined by two boundary surfaces (#101, #102), which forms a pair indicated by the *CorrespondingBoundary* attribute (the *Corresponding boundary* of #101 is #102 and vice versa).

If a boundary surface contains openings (doors, windows, etc.), these openings are indicated by the *InnerBoundary* attribute of the boundary surface, which contains the boundary surface pairs of these openings. In Figure 4.1 for example, the boundary surface #102 contains an opening indicated by the *InnerBoundary* #103. In the same manner, for the inner space boundaries, the space boundaries they belong to are indicated by the attribute *ParentBoundary* (as indicated in Figure 4.1, inner boundaries #104 and #103 have boundaries #102 and #101 as parent boundaries, respectively).

If the boundary surface refers to an internal boundary attached to a specific building space, this space is indicated by the *RelatingBuildingSpace* attribute which points to the respective *IfcSpace* class. For instance, in Figure 4.1, boundaries #102, #103 and #105 indicate space #1 as their internal space in Figure 4.1.

Finally, the building element in which the boundary surface corresponds to, is indicated by the *RelatedBuildingElement* attribute. If the boundary surface is a virtual boundary, such an attribute does not exist. In Figure 4.1, the boundaries #101 #102 refer to an external wall and the boundaries #103 and #104 refer to an external window.

TABLE 4.1: Simulation model elements and respective space boundary pairs information of thermal elements – PHY = PHYSICAL, VIR. = VIRTUAL, INT = INTERNAL, EXT = EXTERNAL

<i>Sim. Model Element</i>	<i>1<sup>st</sup> boundary surface</i>	<i>2<sup>nd</sup> boundary surface</i>
ETE	PHY / INT	PHY / EXT
ITE	PHY / INT	PHY / INT
ESE	PHY / EXT	PHY / EXT
ISE	PHY / INT	PHY / INT
EOE	PHY / INT	PHY / EXT
IOE	PHY / INT	PHY / INT
EAE	PHY / INT	VIR / EXT
IAE	VIR / INT	VIR / INT

Table 4.1 summarizes the relation between the thermal element surface pairs, obtained by CBIP, and their Boundary surface representation. For example, an external thermal element consists of two PHYSICAL boundary surfaces; the first is INTERNAL facing an internal building space and the second is EXTERNAL facing the outside air environment.

Contrasting Figure 4.1 and the enumerated list of the BEP simulation data requirements discussed above, it is obvious that, while most data requirements can be fulfilled by data available through the CBIP algorithm output, missing data for the material thermal properties is a major drawback.

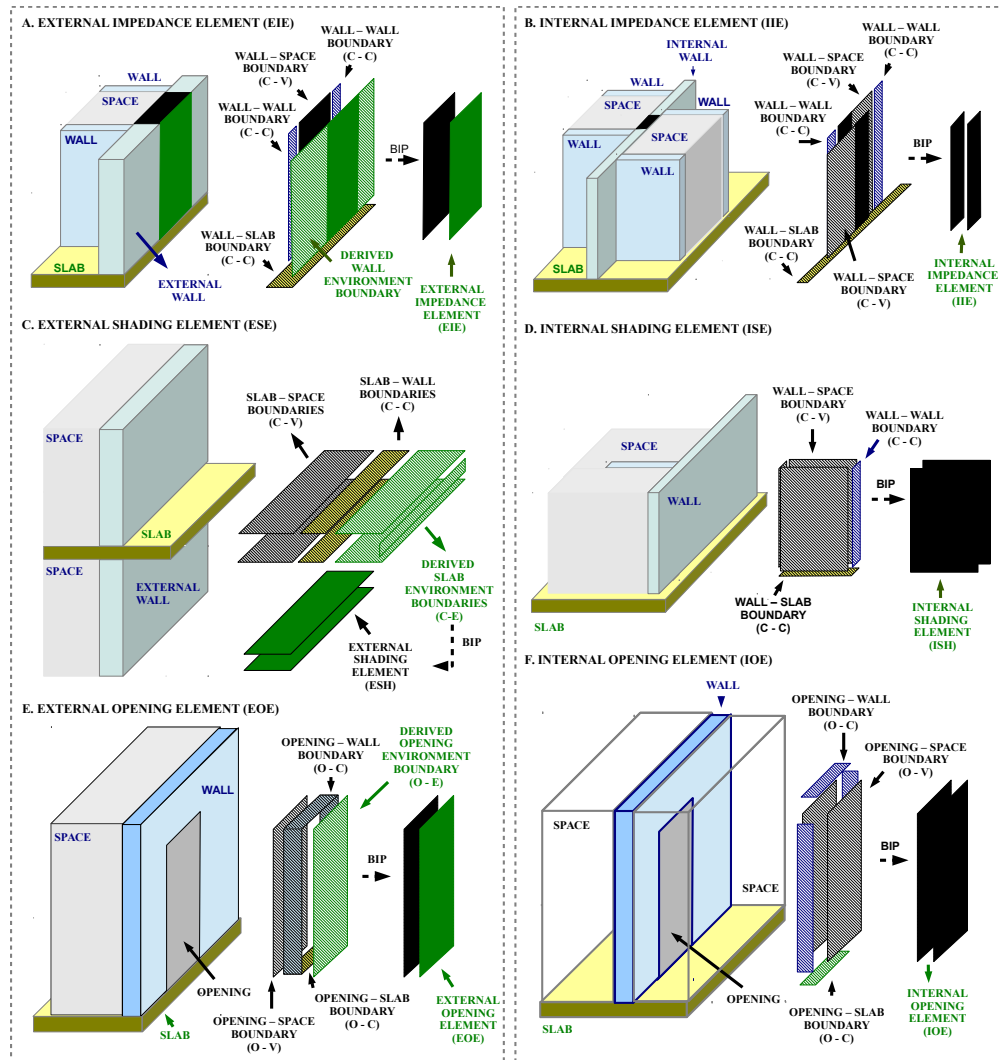


FIGURE 4.2: Simulation model element examples

### 4.3 EnergyPlus IDD-classes

In EnergyPlus, model input data are supplied by two ASCII (text) files: the Input Data Dictionary (IDD) and the Input Data File (IDF).

All possible EnergyPlus classes, and a specification of the properties each class has, are defined in the IDD file. IDD file consists of a comma separated text by a semi column, where an entry defines a class of input objects and specifies all the data needed to model it. The IDD structure defines an input object as a keyword. Each version of Energyplus has a different IDD file. Refer to [21] for a full description of the IDD. IDF file [22] consists of all the necessary IDD-classes' objects to properly define a thermal simulation model of a certain building. Each thermal simulation model has a different IDF file.

In order to utilize all this information, a Matlab script has been developed that identifies the version of the IDF file, parses the appropriate IDD file, and creates a library (MatlabIDD<sub>xx</sub>, where xx is the EnergyPlus version) of Matlab classes, corresponding to EnergyPlus classes.

For a building's envelope thermal simulation, assuming all zones to be free floating, the following conditions hold: shading devices are completely open; internal gains are equal to zero, since lights and electric equipment are switched off; there is no HVAC system available to control the temperature of the zone. In order to develop a model satisfying the aforementioned conditions, objects of the following IDD-classes must be determined:

1. Version – to define the proper version that the IDF is created for.
2. SimulationControl – to specify what kind of calculations a given EnergyPlus simulation will perform; for instance the user may want to perform one or more of the sizing calculations but not proceed to an annual weather file simulation.
3. Building – to describe parameters that are used during the simulation of the building (temperature and loads convergence tolerance values, solar distribution algorithm, etc.).
4. SurfaceConvectionAlgorithm:Inside – to determine the convection algorithm used for surface convection at the inside face of all the heat transfer surfaces in the model.
5. SurfaceConvectionAlgorithm:Outside – to determine the convection algorithm used for surface convection at the outside face of all the heat transfer surfaces in the model.

6. HeatBalanceAlgorithm – to determine the conduction transfer model.
7. Timestep – to specify the simulation timestep.
8. RunPeriod – to describe the elements necessary to create a weather file simulation.
9. Material – to define the thermal properties of opaque materials
10. WindowMaterial:SimpleGlazingSystem – to define thermal and glazing properties of a glazing material only by the U-factor and the SHGC (Solar Heat Gain Coefficient).
11. Construction – to specify the material layer bedding of building elements; each layer of each Construction class object is a material name listed in order from outside to inside.
12. GlobalGeometryRules – a description of geometric parameters before the surface objects are explained in detail (Coordinate system, vertex entry direction, etc.).
13. Zone – to set-up the thermal zones of the building.
14. BuildingSurface:Detailed – to describe each of the opaque surfaces.
15. FenestrationSurface:Detailed – to describe each of the transparent surfaces.

## 4.4 CBIPtoIDF – EnergyPlus

Simulation input parameters, described by objects of classes 1 - 8 and 12, can be set to default values or be entered by users. However, a mapping (transformation) of IFC and CBIP algorithm output data to 9 - 11/ 13 - 15 classes' properties must be performed, respectively. The major resulting transformation rules are numbered throughout this section for easier reference.

### 4.4.1 Opaque and Glazing Materials

Material class has 9 properties and its corresponding MatlabIDD<sub>8.2</sub> class properties are: (1) Name – a unique name associated with each opaque material; (2) Roughness – one of the following choices can be used: VeryRough, Rough, MediumRough, MediumSmooth, Smooth, and VerySmooth; (3) Thickness – the thickness of the material; (4) Conductivity – the thermal conductivity of the material; (5) Density – the density of the material; (6) SpecificHeat – the specific heat of the material; (7) ThermalAbsorptance

– the fraction of incident long wavelength radiation that is absorbed by the material; (8) SolarAbsorptance – the fraction of incident solar radiation that is absorbed by the material; (9) VisibleAbsorptance – the fraction of incident visible wavelength radiation that is absorbed by the material.

WindowMaterial:SimpleGlazingSystem class properties are: (1) Name – a unique name associated with each window material; (2) UFactor – the value for window system U-Factor; (3) SolarHeatGainCoefficient – the value for window system SHGC; (4) VisibleTransmittance – the value for window visible transmittance; optional.

**Rule 1:** Although IFC could incorporate information about the thermal properties of each building entity construction’s material, current IFC-export tools are not able to export such information. Towards eliminating the number of Material and WindowMaterial:SimpleGlazingSystem classes properties that are set manually by the user, only names and thicknesses can be retrieved from the IFC file (each material is an IfcMaterialLayer).

#### 4.4.2 Constructions’ Material Layer Bedding

Construction class has 11 properties and its corresponding MatlabIDD<sub>8.2</sub> class properties are: (1) Name – a unique name associated with each building construction; (2-11) OutsideLayer and Layer2 - Layer10 – each layer of the construction is a material name listed in order from outside to inside.

**Rule 2:** In IFC each Construction is defined as an IfcMaterialLayerSet and is mapped to a Construction class object, for which the layer bedding is a list of IfcMaterialLayers.

#### 4.4.3 Thermal Zones

Zone MatlabIDD<sub>8.2</sub> class properties are: (1) Name – a unique name associated with each building thermal zone; (2) DirectionofRelativeNorth – the Zone North Axis specified relative to the Building North Axis; (3-5) XOrigin, YOrigin and ZOrigin – the X, Y and Z coordinate of a zone origin, respectively; (6) Type – currently unused and always left empty; (7) Multiplier – always set to 1; (8) CeilingHeight – zone’s ceiling height; (9) Volume – zone’s volume; (10) FloorArea – zone’s floor area; (11)-(12) ZoneInsideConvectionAlgorithm and ZoneOutsideConvectionAlgorithm – name corresponding to a zone’s inside and outside convection algorithm which are used to calculate the inside and outside face convection coefficients, respectively; and (13) PartofTotalFloorArea – defaults to Yes if not specified, with (8)-(12) being optional or auto-calculated properties.

**Rule 3:** Each building's space, defined as an IfcSpace class object, is mapped to a new Zone MatlabIDD<sub>8.2</sub> class object with the IfcSpace GID used as the Name of the Zone.

From the CBIP algorithm results, we are interested only for space boundaries surfaces which are attached to a space (from now on zone), pointing to surfaces which have a relating space and that space is not the external earth. Supposing that  $N$  is the number of such surfaces, each surface  $i \in \{1, \dots, N\}$  must be transformed to a BuildingSurface:Detailed or FenestrationSurface:Detailed class object.

#### 4.4.4 Opaque Building Surfaces

The BuildingSurface:Detailed class properties are: (1) Name – a unique name associated with each building surface; (2) SurfaceType – one of the following choices can be used: Wall, Floor, Ceiling, Roof and Surface; (3) ConstructionName – name of the construction used in the surface; (4) ZoneName – zone name to which the surface belongs; (5) OutsideBoundaryCondition – many choices for this property can be used; here, surface, adiabatic, outdoors and ground choices will be investigated; (6) OutsideBoundaryConditionObject – if the Outside Boundary Condition is Surface, then this property's value must be the surface name whose inside face temperature will be forced on the outside face of the base surface; (7) SunExposure – one of the following choices can be used: SunExposed and NoSun; (8) WindExposure – one of the following choices can be used: WindExposed and NoWind; (9) ViewFactortoGround – the fraction of the ground plane (assumed horizontal) that is visible from a heat transfer surface; (10) NumberofVertices – the number of vertices describing a surface; and  $\forall v \in \{1, \dots, \text{NumberofVertices}\}$ , Vertex- $v$ -X, -Y, -Z coordinate – the X, Y, Z coordinate of vertex  $v$ , respectively.

**Rule 4:** If surface  $i$  does not have a parent boundary (which means that it is not an opening), it must be transformed to a BuildingSurface:Detailed class object, where the Name and ZoneName properties values will be equal to the space boundary GID and the space boundary relating space GID, respectively.

Then, the normal vector  $n_i$  of a space boundary is calculated based on the space boundary coordinates, derived by the CBIP algorithm.

**Rule 5:** For each surface  $i$  adhering to rule 4, if  $n_i(3) = 0$ , its SurfaceType is Wall and its ViewFactortoGround is 0.5; else if  $n_i(3) = 1$ , its SurfaceType is Floor and its ViewFactortoGround is 1; else its SurfaceType is Roof/Ceiling and its ViewFactortoGround to ground is 0.

Moreover, if that surface has 2<sup>nd</sup>-level space boundary of type 2a and its corresponding space boundary does not have a relating space, then for that surface the following hold:

OutsideBoundaryCondition, SunExposure, WindExposure properties values are equal to Outdoors, SunExposed and WindExposed, respectively.

In case that surface's corresponding space boundary has as relating space the earth, then for that surface the following hold: OutsideBoundaryCondition, SunExposure, WindExposure properties values are set to Ground, NoSun and NoWind, respectively; however, if corresponding space boundary has a relating space except the earth, then for that surface, the OutsideBoundaryCondition value is set to Surface, instead of Ground, while the OutsideBoundaryConditionObject is the the corresponding space boundary GID.

Else, surface  $i$  has 2<sup>nd</sup>-level space boundary of type 2b and the corresponding BuildingSurface:Detailed object can be neglected or be defined with an Adiabatic OutsideBoundaryCondition.

Completing the transformation of the surface  $i$  to a BuildSurface:Detailed class object, NumberofVertices and coordinates of each vertex are fulfilled according the space boundary coordinates.

#### 4.4.5 Fenestration Surfaces

The FenestrationSurface: Detailed class properties are: (1) Name – a unique name associated with each fenestration surface; (2) SurfaceType – one of the following choices can be used: Window, Door; (3) ConstructionName – name of the construction used in the surface; (4) BuildingSurfaceName – building surface name to which the fenestration surface belongs; (5) OutsideBoundaryConditionObject – if the fenestration surface is interior, then this property's value must be the fenestration surface name whose inside face temperature will be forced on the outside face of the base surface; (6) ViewFactor-toGround – the fraction of the ground plane that is visible from a heat transfer surface; (7) ShadingControlName – the name of the window shading control for this surface; (8) FrameandDividerName – used to specify window frame, divider; (9) Multiplier – the number of identical items on the base surface (always set to 1); (10) NumberofVertices – the number of vertices describing a surface; and  $\forall v \in \{1, \dots, \text{NumberofVertices}\}$ , Vertex- $v$ -X, -Y, -Zcoordinate – the X, Y, Z coordinate of vertex  $v$ , respectively.

**Rule 6:** For each surface  $i$ , if it has a parent boundary, surface  $i$  must be transformed to a FenestrationSurface:Detailed class object.

**Rule 7:** For each surface  $i$  adhering to rule 6, the SurfaceType property value is determined as follows: if its construction consists of opaque Materials (defined as Material class objects) then the SurfaceType is Door; else the SurfaceType is Window.

For that surface, the BuildingSurfaceName is the parent boundary surface GID, while the ViewFactorToGround is autocalculate.

Moreover, if its corresponding space boundary has a relating space, the OutsideBoundaryConditionObject is the corresponding boundary GID.

The FenestrationSurface:Detailed NumberofVertices and coordinates of each vertex are fulfilled according to the space boundary coordinates.

## 4.5 CBIPtoIDF – TRNSYS3d

In TRNSYS 17, to easily input the geometry information into the building model, a plugin called TRNSYS3d for Google SketchUp exists. The TRNSYS3d plugin exports an IDF consisting of Zone,

BuildingSurface:Detailed and FenestrationSurface:Detailed IDD-classes objects. The TRNSYS17 Simulation Studio offers the opportunity to automatically set up a simulation by importing the IDF file with the 3D-Building Wizard.

Since TRNSYS3d utilize particular IDD-classes to export the IDF file, transformation rules that were described in the CBIPtoIDF – EnergyPlus transformation process and correspond to Zone, BuildingSurface:Detailed and FenestrationSurface:Detailed IDD-classes can be performed.

Hence, in the CBIPtoIDF – TRNSYS3d process, rules 3-7 are adopted, considering the following alteration:

**Rule 5:** For each surface  $i$  adhering to rule 4, if that surface has 2<sup>nd</sup>-level space boundary type 2a and its corresponding space boundary does not have a relating space, then for that surface the following hold: OutsideBoundaryCondition, OutsideBoundaryConditionObject, SunExposure, WindExposure properties values are equal to Outdoors, EXTERNAL, SunExposed and WindExposed, respectively.

In case that surface's corresponding space boundary has as relating space the earth, then for that surface the following hold: OutsideBoundaryCondition, OutsideBoundaryConditionObject, SunExposure, WindExposure properties values are set to Ground, BOUNDARY = INPUT 1\*TGROUND, NoSun and NoWind, respectively; however, if corresponding space boundary has a relating space except the earth, then for that surface, the OutsideBoundaryCondition value is set to Zone, instead of Ground, while the OutsideBoundaryConditionObject is ADJACENT = the relating space GID of the corresponding space boundary.

## 4.6 Illustrative Example

CBIPtoIDF – EnergyPlus and CBIPtoIDF – TRNSYS3d transformation processes are implemented on the geometrical three dimensional data of CARTIF building (see Figure 4.3).

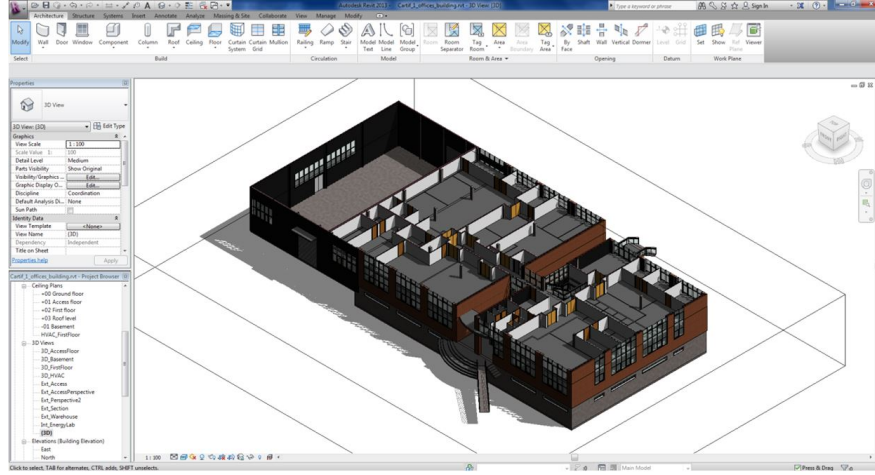


FIGURE 4.3: CARTIF building – BIM in Revit 2014

The geometrical data of the obtained surface pairs — outputs of the CBIP algorithm — are collected and transformed in IDD-classes objects in order to create the BEP model for EnergyPlus and TRNSYS. Since, almost identical transformation rules are adopted to develop the BEP simulation geometry in both processes, the resulted geometry is the same (see Figure 4.4), differing in the boundary condition properties’ definition.

As Figure 4.5 illustrates, where a section plane of the ground floor is depicted, the transformation processes correctly identify: (1) Building and Fenestration surfaces; (2) their boundary condition type; (3) the corresponding surfaces, when boundary condition type is “surface”; (4) the surface type; and (5) the 3d geometry representation of each surface. Moreover, the 2<sup>nd</sup>-level space boundaries of type 2b are neglected and not translated to BuildingSurface:Detail object. Hence, the aforementioned transformation rules properly translate the 2<sup>nd</sup>-level space boundary information to an EnergyPlus and/or TRNSYS simulation’s geometry.

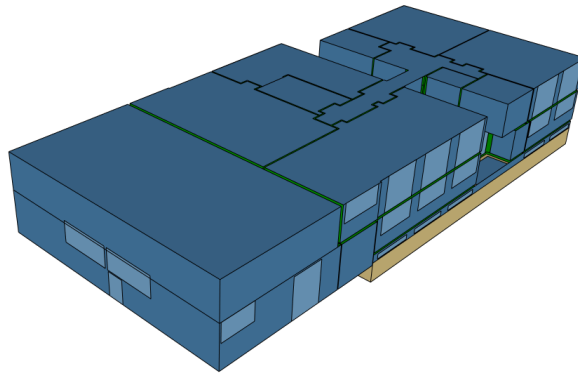


FIGURE 4.4: CARTIF building – BEP simulation model geometry

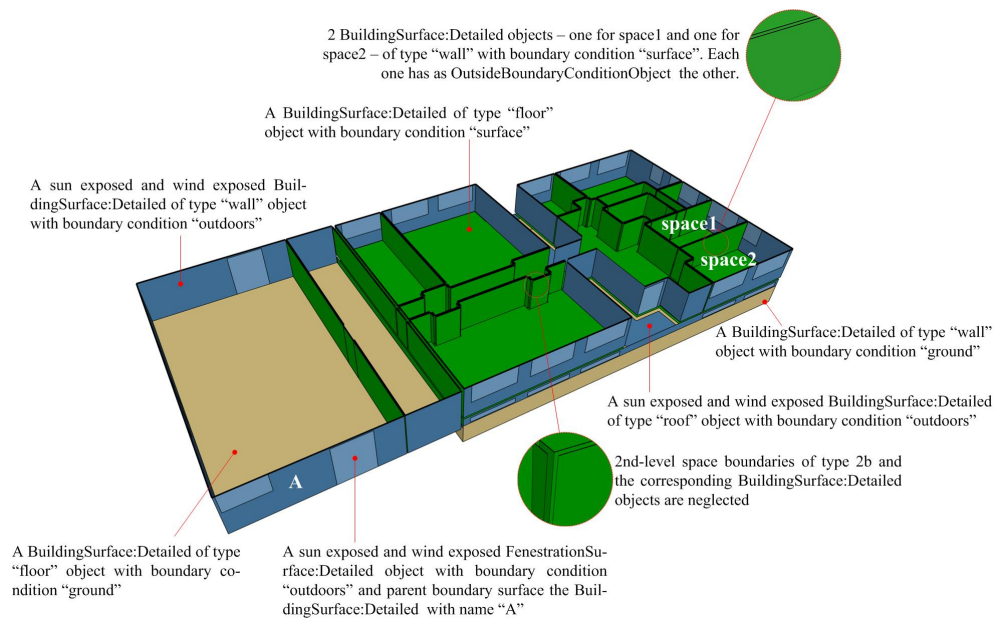


FIGURE 4.5: Illustrative examples of transformation rules’ correctness – CARTIF building

## Chapter 5

# Simulation Speed-up Approaches

According to building thermal simulation 3D zonal-models, a full-scale building consists of one or more thermal zones which are coupled with each other and with the environment. A zone consists of an air volume of a uniform temperature and all surfaces bounding or inside that air volume. The basis for the zone air temperature estimation is the formulation of energy and moisture balances for the zone air and the solution of the resulting ordinary differential equations. Hence, using the Heat Balance method and the Predictor/Corrector approach, briefly presented in Chapter 3, the simulation engine calculates the actual zone mean air temperature.

A BEP simulation model derived by applying the aforementioned model generation process is a full-scale model, since geometry and zoning approximation are considered of high detail. Their simulation is predominantly slowed down by the increasing number of zones, windows and surfaces [36].

In this Chapter, BEP simulation speed-up approaches, required for Control-Design tasks (computationally demanding tasks), are presented and categorized to Geometry and Zoning Reduction Approaches. In Zoning Reduction Approaches, creation of the thermal simulation model can be performed automatically and as such an automatic process for generating speed-up models based on zoning reduction approaches is presented.

### 5.1 Geometry Reduction Approaches

Since high complexity and prohibitive simulation runtime are predominantly due to the full-scale, detailed, geometry representation of the demonstration buildings, geometry reduction approaches are required. Leveraging the building geometry's periodicity, a commonly used approach is to define a representative building block, cut-out from the

full-scale geometry representation and able to reproduce the whole building. Its walls where actually other zones adjoin are defined as “boundary walls”.

A significant difficulty of this approach is to determine proper boundary conditions at each boundary wall, as they are naturally derived by the heat balance equation on a boundary wall. The heat balance equation on exterior faces, described by Equation 4.5, remains valid. However, it is the heat balance on interior faces, corresponding to a boundary wall, that deviates from the one formulated by Equation 4.4.

For these faces, two types of boundary conditions could be defined:

- *Adiabatic boundary condition* - is a special case of *Neumann* conditions, where the surface is assumed to be perfectly insulated. Thus, any heat transfer through boundary walls is excluded for all  $k = 1, \dots$ :

$$-\kappa (\nabla T_k \cdot \hat{n}) = 0; \quad (5.1)$$

where  $\kappa$  is a scalar expressing the wall’s thermal conductivity,  $\nabla T_k$  the temperature gradient of the given wall at the time  $k$ , and  $\hat{n}$  is a transposed normal vector of the wall. For instance, if we consider a floor, the transposed normal vector would be  $\hat{n} = (0, 0, -1)$  and the Equation 5.1 states that there is no heat flux to the ground.

- *Dirichlet boundary condition* - describes a situation for which the surface is maintained at a fixed temperature profile  $T_{sch,k}$  at each simulation time step  $k$ . The boundary condition is formulated as:

$$T_{s,k} = T_{sch,k} \quad (5.2)$$

for all  $k = 1, \dots$ .

The efficiency of two Geometry Reduction Approaches is investigated for two buildings: the FIBP Building, simulated using TRNSYS; and the TUC Building, simulated using EnergyPlus. The proposed Geometry Reduction Approaches are evaluated on these buildings, with respect to the accuracy of the simulation results and their impact (positive or negative) to the computational effort. Towards investigating the accuracy, both the detailed and reduced (speedup) model of the two buildings are simulated for a certain time interval and the temperature deviation is selected as the accuracy measure. For the simulation speedup on the other hand, scenarios with different run periods are investigated and the respective speedup times are presented. In all cases, the speedup models exhibit significantly reduced execution times.

### 5.1.1 FIBP Building – Leveraging Geometry’s Periodicity

As detailed simulation of indoor temperatures of the full-scale model increases computational complexity, a “tower” of three offices upon each other is cut-out, since the whole building can be viewed as a parallel expansion of the simple “tower” sub-model in one dimension. Here, the outer surfaces are actually adjoining rooms’ surfaces, which are defined as “boundary walls”, and the adjacent constructions that have a shading effect on the tower, but are not included in the thermal simulation, are modeled as shading groups. The tower is shown in Figure 5.1.

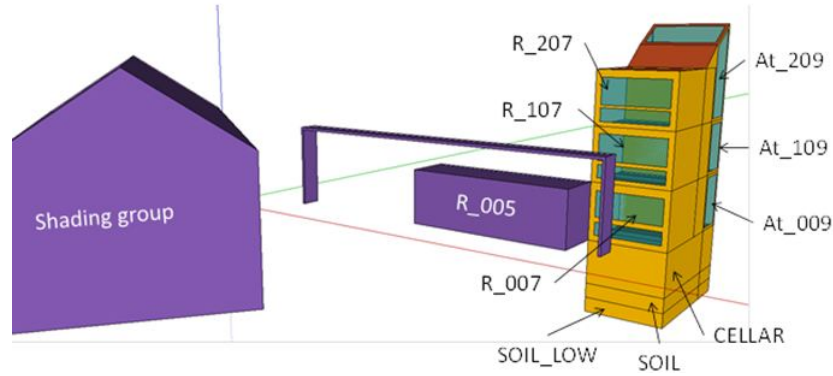


FIGURE 5.1: FIBP Building – tower simulation model

The tower concept is preferred against a horizontal cut-out to keep the essential effect of vertical temperature stratification.

Initially, a *Dirichlet* boundary condition with variable temperature and fixed at each simulation timestep is supplied as boundary condition for each wall. In this case, the boundary condition at the outer plane of contact surfaces is defined by Equation 5.2, where the temperature schedule for the outer plane of a boundary wall ( $T_{sch}$ ) is the actual room temperature of adjoining rooms available from measurements. This approach has limited perspective from a practical point of view, since forecast simulations would require predictions of the temperatures in the adjoining rooms, which there is no chance to be available. Since room temperature trajectories are to be simulated by the model this would be a vicious circle.

Using an *Adiabatic* boundary condition, the aforementioned hindrance can be overcome; however, there are cases, such as the tower operating differently from the rest building, that the assumption of a zero heat transfer could lead to an over- or underestimation of the zone air temperatures.

### 5.1.2 TUC Building – Co-Simulation Approach

For the TUC building on the other hand, its triangular geometry does not define a building block, like the concept of “towers” above, being able to reproduce the whole building, so the periodicity/adiabatic technique can no longer be applied. In this case, an alternative option would be to divide the building into three sub-buildings, each of which can be simulated separately, establishing the connection by using compatible *Dirichlet* boundary conditions. Such splitting of the building, along with the parallel simulation in existing multi-core computer architectures, can yield a sizable reduction in simulation run-time. The use of co-simulation, described next, can yield an effective method, for the transfer of boundary conditions between the sub-buildings, establishing their thermal interaction. For the particular case at hand, the TUC building, since the two main corridors (that include the atrium) are connected through horizontal holes, the need for correctly computing radiant gains in the atrium, forces including both corridors in one of the sub-buildings. With this prerequisite, the whole building is divided to three sub-buildings, shown in Figure 5.2.

Regarding the boundary conditions, a first approach would be to use similar *Dirichlet* boundary conditions to that applied to the “tower” of FIBP building (adjacent rooms temperature). A more complex approach would be to force as boundary condition for each sub-building contact surface the temperature of the corresponding surface, resulting from the adjacent sub-building’s simulation. In other words, suppose that wall  $A$  is a common surface of sub-buildings 1 ( $A_1$ ) and 2 ( $A_2$ ). Then, at the end of sub-building:1 simulation, the temperature profile of surface  $A_1$  is applied as boundary condition to surface  $A_2$  so as to simulation of sub-building:2 run. Thus, a set of *Dirichlet* boundary conditions needs to be defined.

Aiming at control the temperature of a surface that is adjacent to an area that is not included to the simulation model (part of a boundary wall), “OtherSideCoefficients” feature in EnergyPlus [22] is used.

Other side coefficients affect the other side of a surface for all time steps  $k = 1, \dots$  as follows:

$$T_{s,k} = c_1 T_{az,k} + c_2 T_{sch,k} + c_3 T_{\infty,k} + c_4 T_{g,k} + c_5 T_{\infty,k} v_{w,k} \quad (5.3)$$

where  $T_{s,k}$  refers to the surface temperature,  $T_{az,k}$  to the temperature of the zone being simulated,  $T_{\infty,k}$  to the dry-bulb temperature of the outdoor air,  $T_{sch,k}$  to the temperature schedule for the outer plane of the surface,  $T_{g,k}$  to the temperature of the ground and  $v_{w,k}$  to the outdoor wind speed. Here, a temperature profile ( $T_{s,k}$ ) is provided as boundary condition.

In the present work, the boundary condition at the outer plane of each boundary wall is defined by Equation 5.3, with the following coefficients' combination:  $c_1 = 0$ ;  $c_2 = 1$ ;  $c_3 = 0$ ;  $c_4 = 0$ ;  $c_5 = 0$ ; where the temperature schedule for the outer plane of a sub-building boundary surface ( $T_{sch}$ ) is the temperature profile of the inner plane of the corresponding surface of the adjacent sub-building. This situation corresponds to Equation 5.2 and is illustrated in Figure 5.3.

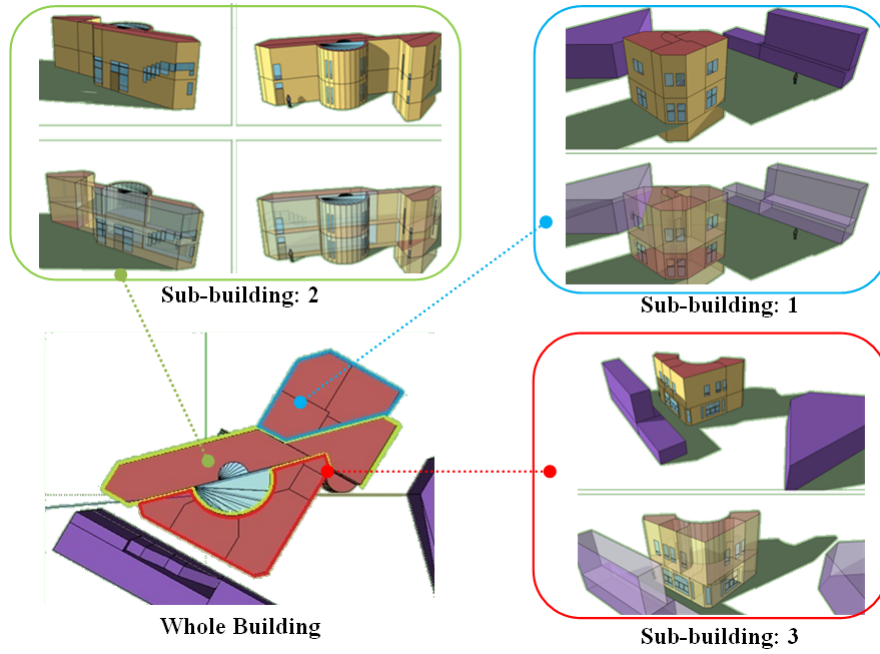


FIGURE 5.2: Dividing the whole building to 3 sub-buildings

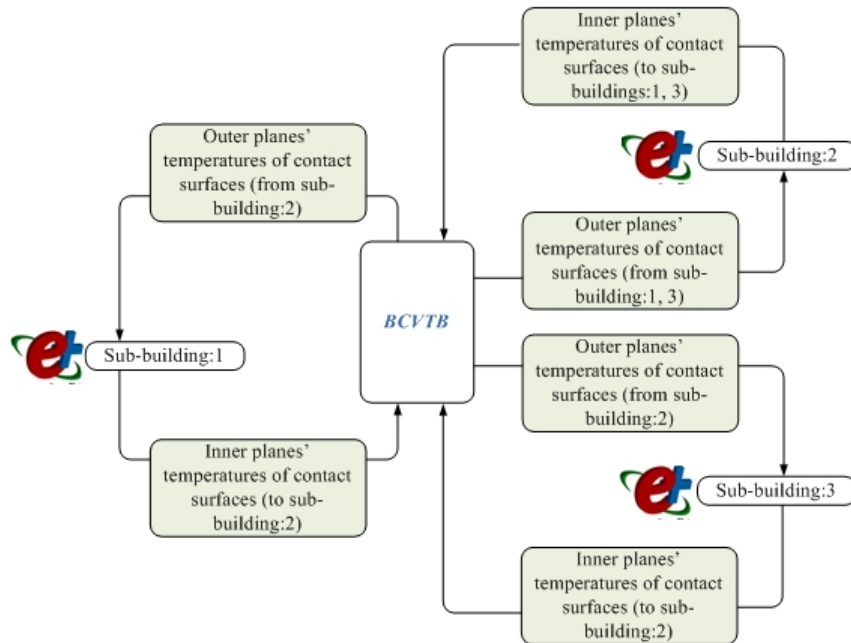


FIGURE 5.3: Data exchange between the sub-building through BCVTB

In order to exchange these boundary conditions in a proper way, a dynamic communication between the three sub-buildings is required and can be achieved by using EnergyPlus with External Interfaces and especially with the Building Controls Virtual Test Bed (BCVTB) [105].

Through such a dynamic communication, a co-simulation between three different thermal models, developed in EnergyPlus, is achieved and the boundary conditions on “boundary walls” (contact surfaces) are defined as follows:

- The state variables  $x_1$  contain the temperatures of inner planes of all that surfaces which belong to sub-building:1 and which are temperatures of outer planes of the corresponding surfaces which belong to sub-building:2.
- The state variables  $x_2$  contain the temperatures of inner planes of all that surfaces which belong to sub-building:2 and which are temperatures of outer planes of the corresponding surfaces which belong to sub-building:1 and sub-building:3.
- The state variables  $x_3$  contain the temperatures of inner planes of all that surfaces which belong to sub-building:3 and which are temperatures of outer planes of the corresponding surfaces which belong to sub-building:2.
- Suppose that the functions  $f_1$ ,  $f_2$  and  $f_3$  compute the next state variable of the simulator 1 and 2 and 3, respectively. The simulator 1 computes, for each time step  $k$ , the sequence:  $x_{k+1,1} = f_1(x_{k,1}, x_{k,2}, x_{k,3})$  and similarly, simulators 2 and 3 compute the sequences  $x_{k+1,2} = f_2(x_{k,1}, x_{k,2}, x_{k,3})$  and  $x_{k+1,3} = f_3(x_{k,1}, x_{k,2}, x_{k,3})$  respectively, with initial conditions on  $x_{0,1}$ ,  $x_{0,2}$  and  $x_{0,3}$ .
- To advance from time  $k$  to  $k + 1$ , each simulator uses its own time integration algorithm. At the end of the time step, the simulator 1 sends the new state  $x_{k+1,1}$  through the BCVTB and receives the state  $x_{k+1,2}$  through the BCVTB. The same procedure is done with the simulators 2 and 3. The BCVTB synchronizes the data in such a way that it does not matter which of the three simulators is called first.

### 5.1.3 Simulation Results

Towards investigating the accuracy of the simulation results, both the full-scale and the reduced (after the implementation of a speed-up technique) model are simulated for a predefined time interval (run period), and standard deviation of the difference between the resulting zone air temperatures are calculated according to the following equations.

To start, the temperature difference between the full-scale and the reduced model simulation results is given for the  $i$ th temperature and  $k$ th time step by:

$$\Delta T_{k,i} = T_{\text{full-scale},k,i} - T_{\text{reduced},k,i}. \quad (5.4)$$

Consecutively, the mean is calculated according to:

$$\overline{\Delta T}_i = \frac{1}{n_{\text{data}}} \sum_{k=1}^{n_{\text{data}}} \Delta T_{k,i} \quad (5.5)$$

while, the sample standard deviation is calculated as follows:

$$s_i = \sqrt{\frac{1}{n_{\text{data}} - 1} \sum_{k=1}^{n_{\text{data}}} (\Delta T_{k,i} - \overline{\Delta T}_i)^2}. \quad (5.6)$$

The confidence bound is given as  $\overline{\Delta T}_i \pm s_i$ . A large  $\overline{\Delta T}_i$  points in the direction of a general failure, whereas a large  $s_i$  indicates a high uncertainty for the reproducibility. The reason for a general failure can be either a wrong model or wrong parameterizations.

For the simulation speedup on the other hand, the simulation runtimes for a number of cases, performed to record the reduction in simulation runtime due to the implementation of a speed-up technique, are presented.

### 5.1.3.1 FIBP Building

The required simulation time of a *basic* building simulation in TRNSYS scales up with increasing number of thermal zones according to the polygon shown in Figure 5.4 — the relevant calculation time in *seconds* is on the left axis. The right axis refers to the simulation time in minutes required to simulate the full-scale and the simplified (tower) FIBP building model. The simulation time of the full-scale model including all thermal zones (25), is approximately 800 minutes for one year. The tower (9 zones) takes approximately 50 minutes for the same simulation interval. Since the models were designed for the purpose of co-simulation the following considerations are relevant.

Co-simulation requires repeated simulations for the prediction horizon in connection with a stochastic optimization algorithm; the number of zones increases the dimension of the parameter space and hence prolongs the convergence time and the number of iterative simulations required by the algorithm. A prediction horizon of 72 hours including a

one day settling phase requires approximately 7 minutes simulation time for the full-scale model. Assuming 100 iterations — which is a conservative estimation for a 54 dimensional parameters space — to find an optimum, a controller design would take approximately 11 hours. This allows for only two complete controller design runs a day. Using the tower model requiring only 24 s for the relevant prediction horizon, at least 24 runs per day are possible.

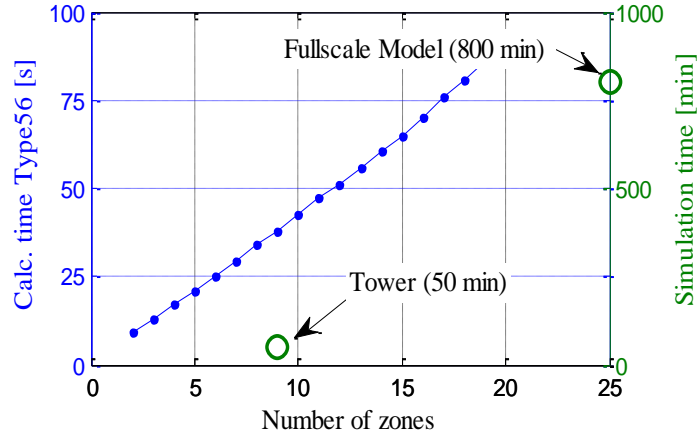


FIGURE 5.4: Upscaling of the simulation runtime for increasing number of thermal zones in TRNSYS Type 56 — base interval is one year. Left axis: general building model, right axis: fullscale FIBP & tower building model

In order to compare the impact on the results for this simplification, two simulations for a winter month (January) and two simulations for a summer month (June) are conducted. The setpoint temperatures (blue dashed line) — available from real measurement data — are applied as room set temperatures for the three zones R007, R107 and R207. Figure 5.6 shows that, for winter months, although a phase shift is clearly visible, temperature patterns are very similar.

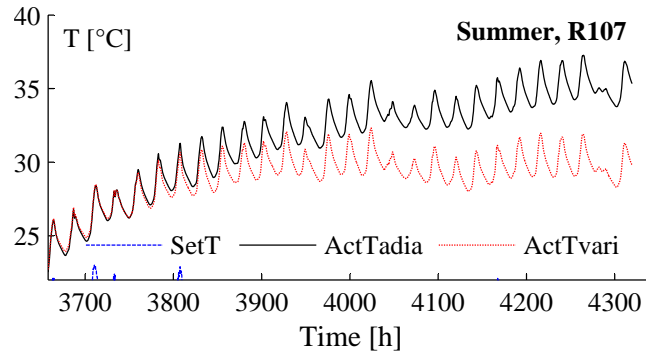


FIGURE 5.5: FIBP tower building model – different boundary conditions, for a simulation interval of a summer month

However, in summer we have a clear overestimation of the temperatures which is due to the fact that neighboring rooms receive some sort of cooling e.g. night ventilation, while for the tower zones no window opening or any other user influence was considered. In addition, the summer interval shows the significance of the thermal stratification in vertical direction; the difference between the bottom and the top zone is approximately 5K.

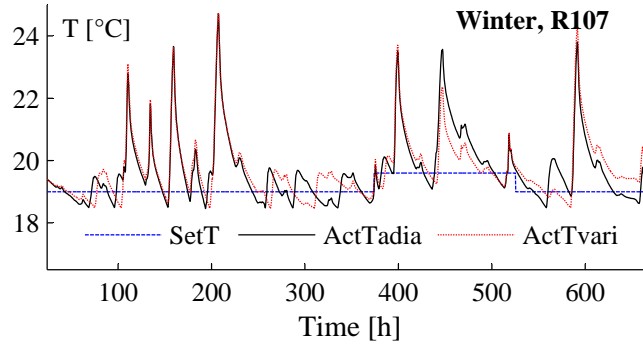


FIGURE 5.6: FIBP tower building model – different boundary conditions, for a simulation interval of a winter month

Hence, if rooms are operated similarly as is the case for winter, the two optional boundary condition lead to nearly the same results. But if the neighboring room is operated differently, as is the case for the summer month, the thermal coupling leads to systematic over- or underestimation.

The mean and the standard deviation of the expression  $\Delta T_k = \text{ActTvaria}_k - \text{ActTadia}_k$ , characterize the similarity of the temperature trajectories for different boundary conditions. Variable boundary conditions ( $\text{ActTvaria}_k$ ) are considered providing the same conditions for the tower as would be prevalent for these zones embedded in the full-scale model. Results of the deviation are presented in Table 5.1.

TABLE 5.1: Mean and standard deviation results for rooms R207, R107 and R007 of the FIBP tower building model

Thermal Zone $i$	Winter month		Summer month	
	$\overline{\Delta T}_i$	$s_i$	$\overline{\Delta T}_i$	$s_i$
R207	-0.020	0.145	-2.14	1.91
R107	0.055	0.476	-2.51	1.92
R007	-0.005	0.527	-1.96	1.40

### 5.1.3.2 TUC Building

Moving to the TUC Building simulation experiments setup, a whole year simulation time interval is selected to investigate the deviation, proving that a parallel connection between the sub-buildings is able to assimilate zone air temperature trends of a full-scale model, regardless of the neighboring zones thermal conditions (temperature, humidity, etc.). Thus, assuming all zones to be free floating, the conditions during this period are as follows: shading devices are completely open; internal gains are equal to zero, since lights and electric equipment are switched off; there is no HVAC system available to control the temperature of the zone; and natural ventilation is considered at night, during summer months only.

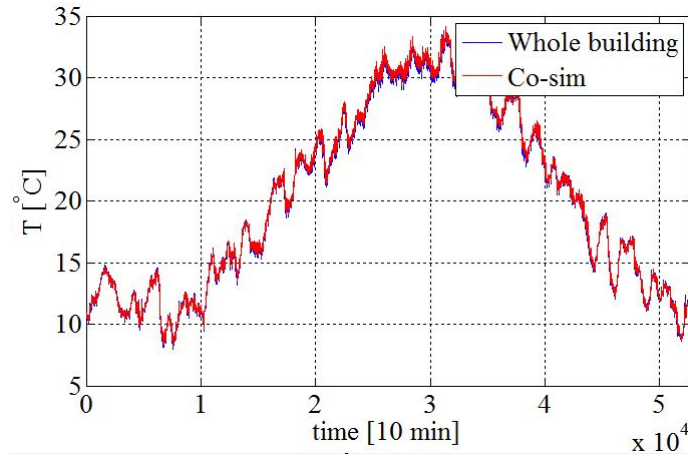


FIGURE 5.7: Simulated Air Temperature values in an office room (office 11) — whole building model and sub-building:1 model (Co-sim)

Observing the temperature results in Figure 5.7, the similarity of trends (almost identical) for a randomly selected office room (office 11 – part of sub-building:3) is obvious. Although the temperature differences are near to zero during the winter months, slight differences, which are within acceptable limits, arise during the summer months, proving that the above types of boundary conditions are efficient enough to describe the dynamic behavior of the full scale building. The mean and the standard deviation for the office rooms are presented in Table 5.2.

TABLE 5.2: Mean and standard deviation results for each office room of TUC building

Thermal Zone $i$	Whole year	
	$\overline{\Delta T}_i$	$s_i$
office 1	0.168	0.028
office 2	0.238	0.033
office 3	0.153	0.025
office 4	0.251	0.027
office 5	0.132	0.075
office 8	0.251	0.060
office 9	0.225	0.051
office 10	0.251	0.053
office 11	0.251	0.082
office 13	0.173	0.065

Three scenarios are investigated to prove the effectiveness of the parallel connection, regarding the simulation runtime for different run periods. In Table 5.3, Full is considered to be the full-scale TUC Building model, Sub:1, Sub:2 and Sub:3 are the simulation models of the sub-buildings presented in Figure 5.2, running independently with boundary conditions described by Equation 5.3 and coefficients  $c_1 = 0.5, c_2 = 0.5, c_3 = c_4 = c_5 = 0$ ; and Co-sim describes the parallel simulation which was presented above.

While one might expect that the runtime of the parallel simulation should coincide with the maximum value comparing Sub:1, Sub:2 and Sub:3 simulation runtimes at each run period (one day, one week and one month), Table 5.3 shows that, when the run period of simulation is growing, the resulting runtime of the dynamic connection is always greater than the corresponding runtime of sub-buildings. This is due to the large size of data exchanged through the co-simulation.

TABLE 5.3: Simulation runtimes for each sub-building (Sub:1, Sub:2 and Sub:3), the parallel simulation (Co-sim) and the whole building (Full)

Run Period	sub:1 (sec)	sub:2 (sec)	sub:3 (sec)	Co-sim (sec)	Full (sec)
1 day	13.6	9.9	13.2	15.4	98.9
1 week	20.5	21.9	17.3	37.8	201.1
1 month	47.4	75.5	33.5	120.9	612.9

Nevertheless, it is estimated that, dividing the whole building to three sub-buildings leads to a reduction of simulation runtime by 80%. Such a result was expected since the

division of the original building has significantly reduced the number of thermal zones, walls and windows, as shown in Table 5.4.

TABLE 5.4: Number of zones, walls and windows for each sub-building (Sub:1, Sub:2 and Sub:3) and the whole building (Full-scale)

Model	Full-scale	Sub:1	Sub:2	Sub:3
Zones	30	8	14	8
Walls	281	101	107	99
Windows	105	33	51	21

## 5.2 Zoning Reduction Approaches

In common practice, a full-scale thermal simulation model treats each room of a building as an individual thermal zone. Such an assumption increase significantly the simulation runtime, since computational effort is more than proportional to the number of zones; as mentioned earlier, increased number of zones corresponds to increased number of ordinary differential equations to be solved. Hence in many cases, building simulation modelers incorporate the HVAC zones definition, where each zone consists of one or more rooms and a thermostat assigned to that zone. At this level of detail, the thermal simulation model, where each HVAC zone is a thermal zone, can be still expensive for computationally demanding tasks. Concerning a further zoning reduction, building simulation experts are able to reduce the number of HVAC-thermal zones, but such a reduction is usually based on some similarity between the regions being combined (e.g. similar internal loads).

Towards an automatic methodology to reduce the number of zones, utilizing simulation results of a full-scale, validated, thermal simulation model, in this Section, two approaches are presented. The first approach utilizes the Hierarchical Clustering theory [64], while the second approach adopts the Koopman modes theory [67]. The Koopman modes, as a systematic approach to zoning and model reduction, has recently been proposed in [27], where motivational results are presented for a real building.

Implementations of these approaches are illustrated in the full-scale model of CARTIF Building and their results are presented.

### 5.2.1 Hierarchical Clustering Analysis

Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Hierarchical clustering strategies are divided into agglomerative and divisive.

Given a data-set consisting of a number of objects — for example, time-series of thermal zones' air temperatures, derived from a thermal model simulation for a predefined simulation run-period — the agglomerative technique considers that each object starts in its own cluster and pairs of clusters are merged at each step, until all objects are placed in one cluster (bottom-up). The divisive technique works vice versa; assuming that all objects belong to a cluster, it divides this cluster until each cluster contains a unique object (top-down).

Here, the agglomerative strategy for hierarchical clustering is adopted, since considering a large number of thermal zones, a reduced number of thermal zones must be obtained.

In a broad sense, the agglomerative technique is performed as follows [64]:

1. A single cluster is defined for each object of the data-set; for instance, if there is a set of  $n_x$  objects to be merged, there are  $n_x$  clusters, where each cluster contains an object of  $n_x$ ;
2. The similarity or dissimilarity between every pair of objects in the data set is estimated; here, the distance between every objects' pair is calculated. Some commonly norms (metric) to compute this distance are: Euclidean, Squared Euclidean, and Maximum distances.
3. The pair with the minimum distance is selected; this pair is merged into one cluster, leading to newly formed clusters.
4. The newly formed clusters are grouped into larger clusters until a hierarchical tree is formed. To calculate the distance between clusters that include more than one objects a linkage criterion is selected. The linkage criterion determines the distance between sets of objects as a function of the pairwise distances between objects. Some commonly used linkage criteria between two sets of objects are: Maximum, Minimum and Average linkages.
5. Finally, the point to cut the hierarchical tree into clusters is determined. In this step, branches off the bottom of the hierarchical tree are pruned, and all objects below each cut are assigned to a single cluster.

The choice of an appropriate metric (in step 2.) will influence the shape of the clusters, as some elements may be close to one another according to one distance and further away according to another.

To explain the aforementioned metrics and linkage criteria, suppose that we have a data-set represented as a matrix  $G \in \mathbb{R}^{n_x, n_{\text{data}}}$ , where  $n_x$  stands for number of objects (e.g.

$n_x$  zones' operative temperature timeseries) and  $n_{\text{data}}$  stands for the number of time steps when the values were measured. Moreover, two clusters  $A$  and  $B$  are considered. Cluster  $A$  consists of  $n_A$  objects where each of them is element of  $\{1, 2, \dots, n_x\}$ , while cluster  $B$  consists of  $n_B$  objects  $\{B_1, B_2, \dots, B_{n_B}\}$ , where  $n_A + n_B \leq n_x$ . The distance  $\delta_{i,j}$  between objects  $i \in A$  and  $j \in B$  can be calculated by one of the following equations:

- According to the Euclidean distance:

$$\delta_{i,j} = \sqrt{\sum_{k=1}^{n_{\text{data}}} (G_{i,k} - G_{j,k})^2}; \quad (5.7)$$

- or according to the squared Euclidean distance:

$$\delta_{i,j} = \sum_{k=1}^{n_{\text{data}}} (G_{i,k} - G_{j,k})^2 \quad (5.8)$$

- according to the maximum distance:

$$\delta_{i,j} = \max_k |G_{i,k} - G_{j,k}|. \quad (5.9)$$

The distance between cluster  $A$  and cluster  $B$ ,  $\delta_{AB}$ , is defined as follows:

- According to the Maximum linkage criterion:

$$\delta_{AB} = \min_{i,j} \delta_{i,j} \quad (5.10)$$

- according to Minimum linkage criterion:

$$\delta_{AB} = \max_{i,j} \delta_{i,j} \quad (5.11)$$

- according to Average linkage criterion:

$$\delta_{AB} = \frac{\sum_{i \in A} \sum_{j \in B} \delta_{i,j}}{n_A \cdot n_B}. \quad (5.12)$$

In the present work, the choice of linkage criterion is not of high importance, since the point to cut the hierarchical tree into clusters is defined as the point where the zones' pair with the minimum distance are linked (see Figure 5.8).

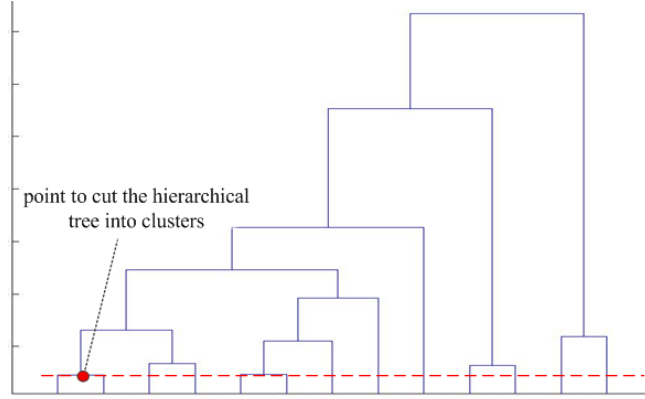


FIGURE 5.8: Hierarchical tree – the point to cut the hierarchical tree into clusters is defined as the point where the zones' pair with the minimum distance are linked

Such a definition stems from the following interpretation: When two thermal zones are merged into a new one, the new zone's resulted temperature profile could be conspicuously different from the temperature profiles of the initial zones; hence, the data-set should be updated whenever a two zones' merging is performed. To accomplish this, the agglomerative technique is performed repeatedly in the following manner: (1) Set the desired number of zones; (2) Consider a data-set consisting of time series, derived from the full-scale zonal model simulation; (3) Apply the agglomerative technique, where the point to cut the hierarchical tree into clusters is defined as the point where the zones' pair with the minimum distance are linked; (4) Create a newly formed zonal model according to the results of step 3 and simulate it to receive the updated data-set; (5) Using the updated data-set, repeat steps 3 and 4, until the number of zones is equal to the desired one.

### 5.2.2 Koopman Mode Analysis

Because the equations describing a building model can be of a high dimension and are often not accessible analytically, methods are required which are measurement, or time-series based, to study such systems. In this context, the Koopman operator can be applied to building models for the visualization and analysis of these systems. By projecting the data-set (time-series of simulated objects) from a building simulation onto eigenfunctions of the operator, spatial features of the system being studied can be extracted.

The procedure of merging thermal zones from a full-scale model using Koopman modes is:

1. Simulate the full-scale thermal simulation model resulting to an objects' data-set of interest (in this work thermal zones' air temperatures).
2. Calculate Koopman eigenvalues and modes by projecting the objects' data-set onto eigenfunctions of the Koopman operator.
3. Merge thermal zones with Koopman modes of similar amplitude and phase at frequencies (modes) of interest.

There are several methods available for calculating Koopman modes [67, 96]. Here, Arnoldi algorithm [96] is selected, since in [27] it is presented as an efficient algorithm to decompose building simulation time series into Koopman modes, able to capture the thermal behavior of a building. Arnoldi algorithm is described below.

Suppose again that we have the data matrix  $G$  as introduced above. Then, empirical Ritz values (Koopman eigenvalues)  $\lambda_k$  and empirical Ritz vectors (Koopman modes, eigenfunctions)  $v_k$ ,  $\forall k = 1, 2, \dots, n_{\text{data}}$  are calculated by the following procedure:

1. Find constants  $c_k$ ,  $\forall k = 1, \dots, n_{\text{data}} - 1$  such that  $\mathbf{r} \perp G$ , where:

$$\mathbf{r}_i = G_{i, n_{\text{data}}} - \sum_{i=1}^{n_{\text{data}}-1} c_i G_{i, k}. \quad (5.13)$$

for all  $i = 1, \dots, n_x$ . Since  $\mathbf{r} \perp G$  the following equation holds:

$$G'_{\bullet, k} \mathbf{r} = 0 \quad (5.14)$$

for all  $k = 1, \dots, n_{\text{data}} - 1$  where  $G_{\bullet, k}$  stands for  $k$ th column of matrix  $G$ .

Suppose that  $A = \{a_{k, \tilde{k}}\} \in \mathbb{R}^{n_{\text{data}}-1, n_{\text{data}}-1}$ , where  $a_{k, \tilde{k}} = G'_{\bullet, k} G_{\bullet, \tilde{k}}$  and  $b_k = G'_{\bullet, k} G_{\bullet, n_{\text{data}}}$  and  $c = \{c_i\} \in \mathbb{R}^{n_{\text{data}}-1}$ . Then, constants  $c_k$  can be found by solving the following system of linear equations:

$$Ac = b \quad (5.15)$$

2. Define the companion matrix  $C$ :

$$C = \begin{bmatrix} 0 & 0 & \cdots & 0 & c_1 \\ 1 & 0 & \cdots & 0 & c_2 \\ 0 & 1 & \cdots & 0 & c_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & c_{n_{\text{data}}-1} \end{bmatrix} \quad (5.16)$$

and find its eigenvalues  $\lambda_k$ ,  $\forall k \in \{1, 2, \dots, n_{\text{data}} - 1\}$ .

3. Define the Vandermonde matrix  $T$  as follows:

$$T = \begin{bmatrix} 1 & \lambda_1 & \lambda_1^2 & \dots & \lambda_1^{n_{\text{data}}-2} \\ 1 & \lambda_2 & \lambda_2^2 & \dots & \lambda_2^{n_{\text{data}}-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \lambda_{n_{\text{data}}-1} & \lambda_{n_{\text{data}}-1}^2 & \dots & \lambda_{n_{\text{data}}-1}^{n_{\text{data}}-2} \end{bmatrix}. \quad (5.17)$$

4. Finally, calculate the matrix  $V = GT^{-1}$ , columns of which are the Koopman modes. Note that  $V$  might contain complex numbers because of inversion of  $T$ .

From the calculated Koopman modes of a full-scale building model data, thermal zones are merged if their amplitudes and phases within the Koopman modes considered are within some tolerance. Combining zones using this approach physically corresponds to combining zones which behave similarly due to internal heat generation and environmental heat transfer. The following definition is used for comparing the amplitudes and phases of zones and creating merged zones approximations:

Select  $\epsilon_1, \epsilon_2 \geq 0$  and consider  $i, j \in \{1, 2, \dots, n_x\}$ , while the data' objects are the  $n_x$  zones' air temperatures. Then, zones  $i$  and  $j$  can be merged if:

$$|||v_{i,k}| - |v_{j,k}||| < \epsilon_1 \quad (5.18)$$

$$|\angle v_{i,k} - \angle v_{j,k}| < \epsilon_2 \quad (5.19)$$

for all the  $k$ -th Koopman modes of interest where  $|| \cdot ||$  stands for absolute value of a complex number and  $\angle$  stands for phase of a complex number. Koopman modes of interest correspond to the largest modes, calculated in Step 4. The main idea of investigating the validity of Equations 5.18 and 5.19 for Koopman modes of interest and not all Koopman modes is based on the fact that only several modes are required to describe important characteristics of the building's thermal response.

### 5.2.3 Automatic Generation of Models

Summarizing, given a full-scale zonal model and a representative data-set consisting of a zone's level output variable (for example, air temperature for all zones) on which zoning reduction relies: (1) in Hierarchical clustering approach, substantial recreations

of intermediate zonal models are required, before the recreation of the final speedup model that consists of the desired number of zones; (2) in Koopman modes approach, a zoning approximation, where Equations 5.18 and 5.19 hold for a predefined Koopman modes of interest number, is used to recreate the final speedup model.

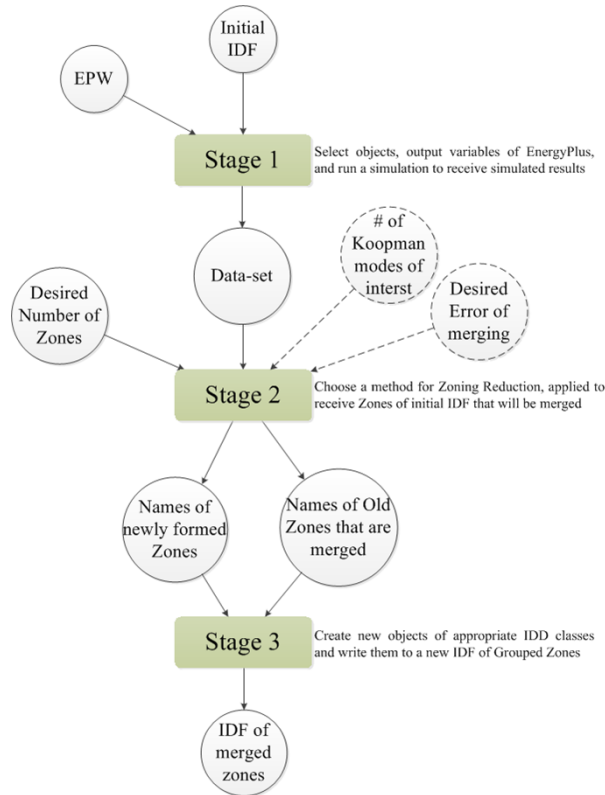


FIGURE 5.9: An overview of the Automatic Process for Generating Speed-up Model based on Zoning Reduction Approaches

In both approaches, recreation of a thermal simulation model is required, a tedious, slow and error-pruning process, commonly performed by a thermal simulation modeler manually. The benefit of an automatic process for generating speedup models based on Zoning Reduction Approaches would be twofold: (1) it would be orders of magnitude faster than manually recreating building geometry; and (2) it would be less susceptible to human error. Towards this direction, the three stages process presented in Figure 5.9 is proposed, utilizing EnergyPlus as the simulation engine to develop zonal models. In Stage 1, the objects of the data-set, output variables of EnergyPlus, are defined and a full-scale zonal model (initial model) simulation is performed to receive simulated results. A detailed representation of steps embedded in Stage 1 is shown in Figure 5.10.

In EnergyPlus, model input data are supplied by two ASCII (text) files: the Input Data Dictionary (IDD) and the Input Data File (IDF).

As previously mentioned, all possible EnergyPlus classes, and a specification of the properties each class has, are defined in the IDD file, while an IDF file consists of

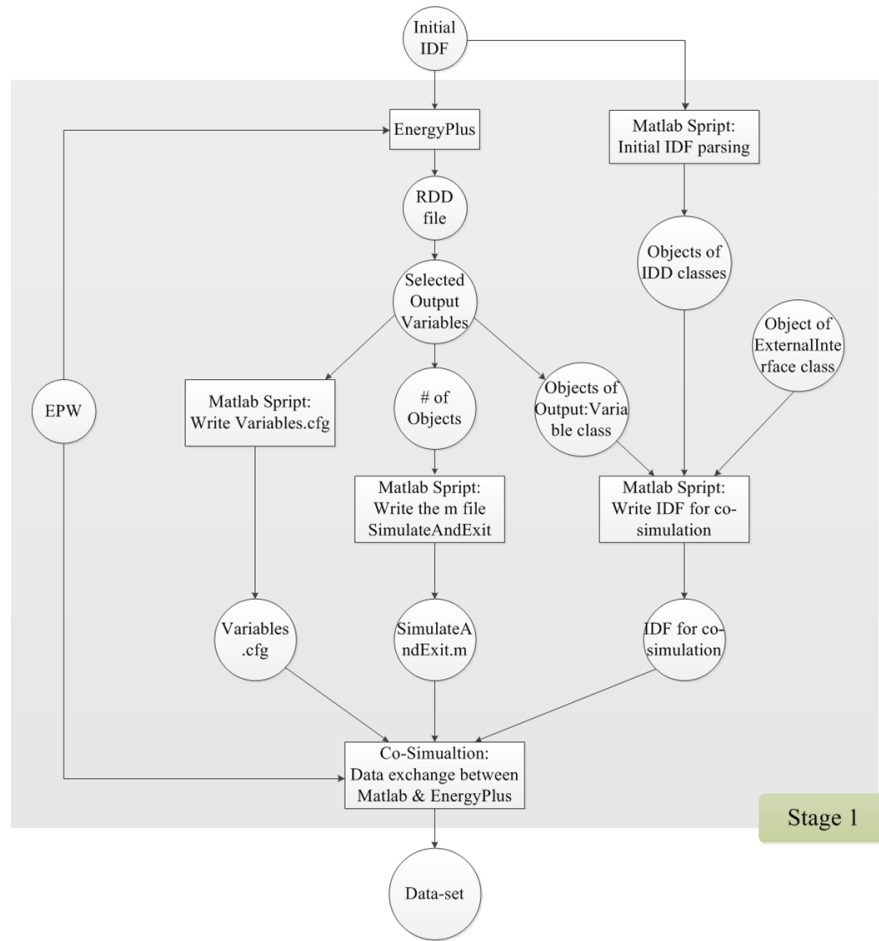


FIGURE 5.10: Stage 1 – Select objects — output variables of EnergyPlus — and run a simulation to receive simulated results

all the necessary IDD classes' objects to properly define a thermal simulation model of a certain building. Each thermal simulation model has a different IDF file. This information is parsed by two Matlab scripts: the first script identifies the version of the IDF file, parses the appropriate IDD file, and creates a library (MatlabIDD<sub>xx</sub>, where xx is the EnergyPlus version) of Matlab classes, corresponding to EnergyPlus classes; the second script identifies the version of a certain IDF file and parses this file conducting to MatlabIDD<sub>xx</sub> objects definition.

Beyond a wide variety of EnergyPlus output variables, particular variables can be reported depending on the actual simulation problem described in the IDF. The Report Data Dictionary (RDD) is a text file listing those variables available for reporting during the simulation of a certain IDF, including possible objects of the data-set required for a Zoning Reduction Approach execution. For instance, Fanger's predicted mean vote could not be reported, if People class objects for all zones have not been defined. Selecting a zone's level output variable from that list, an object of the *Output:Variable* class is defined and imported in the initial IDF. A plethora of zone's level output variables

could exist in the RDD file, commonly including zone air temperature, zone operative temperature, zone relative humidity and zone Fanger Predicted Percentage of Dissatisfied people (Fanger PPD) to name but a few. However from this point forward, we assume that the selected output variable is the zone operative temperature.

After an initial IDF — enriched with the selected *Output:Variable* — simulation run, the resulted data-set of the selected variable is printed in a comma separated text by a semi column, where each column corresponds to a unique zone's variable time-series (zone air temperature with reporting frequency equal to the simulation timestep). The order of columns is alphabetical, according to the names of the zones, and as such not allowing the user to set a preferable order. To achieve this, EnergyPlus could be used in conjunction with the Building Controls Virtual Test Bed.

To configure the data exchange between EnergyPlus and Matlab through BCVTB, the following four steps (described further in Section 6.1) are required:

- Enrich the initial IDF with the selected *Output:Variable* object and an object of the *ExternalInterface* class.
- Develop an xml file, named variables.cfg, that defines the mapping between EnergyPlus and BCVTB variables.
- Create an m file, named simulateAndExit.m, to determine the data exchange between MATLAB and the BCVTB variables.
- Create a Ptolemy model.

In order to automatically create — when required — the enriched IDF, the variables.cfg and the simulateAndExit.m files, three respective Matlab scripts have been developed.

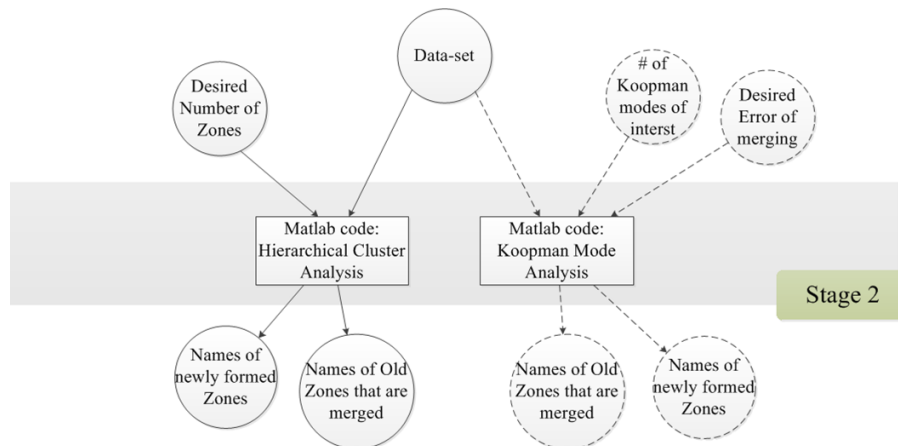


FIGURE 5.11: Stage 2 – Choose a method for Zoning Reduction to receive Zones of initial IDF that are merged

The data-set consisting of the simulated results (outputs of Stage 1) along with the desired number of zones, or the desired errors of zones' merging  $\epsilon_1$ ,  $\epsilon_2$  and the Koopman modes of interest number, are forwarded to Stage 2, where a Zoning Reduction Approach is chosen and is applied to receive groups of zones that will be merged (see Figure 5.11).

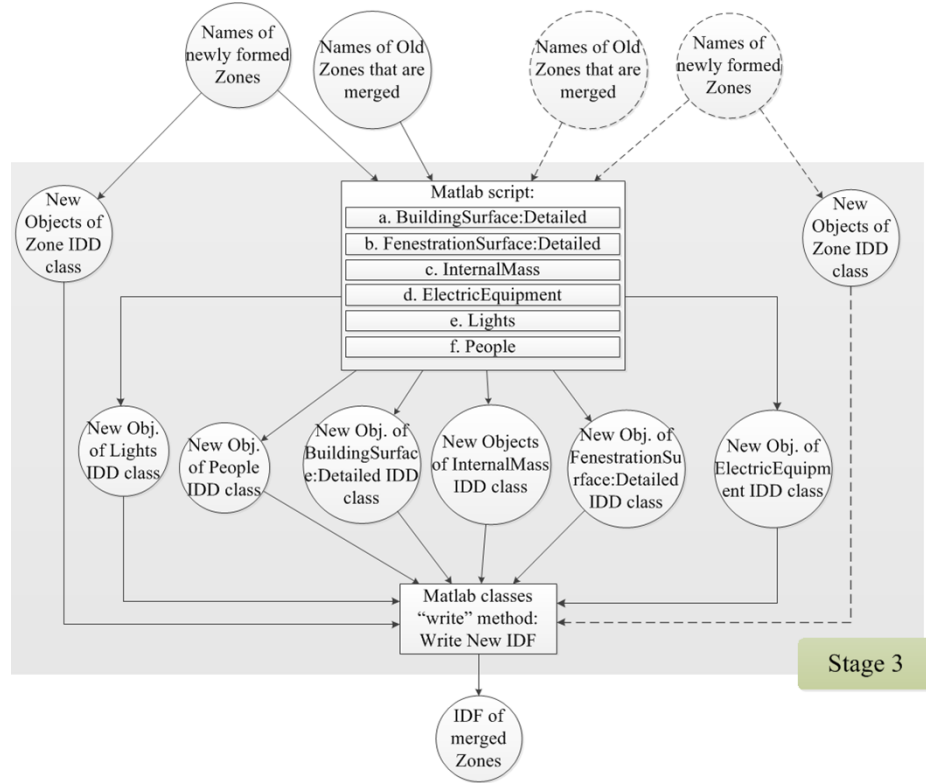


FIGURE 5.12: Stage 3 – Create new objects of appropriate IDD classes and write them to a new IDF of Grouped Zones

Names of newly formed zones and names of initial model's zones that belong to each newly formed zone, are finally used to automatically recreate the final speedup model in Stage 3 (see Figure 5.12), in the following manner:

- For each newly formed zone, a new object of the *Zone* class is determined.
- When zones are merged forming a new zone, shared walls and openings (doors, windows) are no more *BuildingSurface:Detailed* and *FenestrationSurface:Detailed* class objects, respectively, but *InternalMass* class objects acting as thermal mass for the new zone. Moreover, internal loads from electric-equipment, lighting, and occupancy, objects of *ElectricEquipment*, *Lights* and *People* classes, respectively, are combined for the newly formed zone. With the initial IDF parsed objects of *BuildingSurface:Detailed*, *FenestrationSurface:Detailed*, *IntrnalMass*, *ElectricEquipment*, *Lights* and *People* classes as inputs, a Matlab script has been developed and is applied to properly determine new objects of these classes.

- For each new object of *Zone*, *BuildingSurface:Detailed*, *FenestrationSurface:Detailed*, *InternalMass*, *ElectricEquipment*, *Lights* and *People* classes the respective class “write” method is called, implementing the writing operations performed on objects of the corresponding class to create the new IDF of merged zones.

#### 5.2.4 Simulation Results

The implementation of Zoning Reduction approaches is performed by a proposed Automatic Process described in Section 5.2.3, which has been evaluated for a real office building, the CARTIF Building. The main objective of the test case presented here is to verify applicability of the process and to evaluate the proposed Zoning Reduction Approaches, with respect to the accuracy of the simulation results and their impact (positive or negative) to the computational effort.

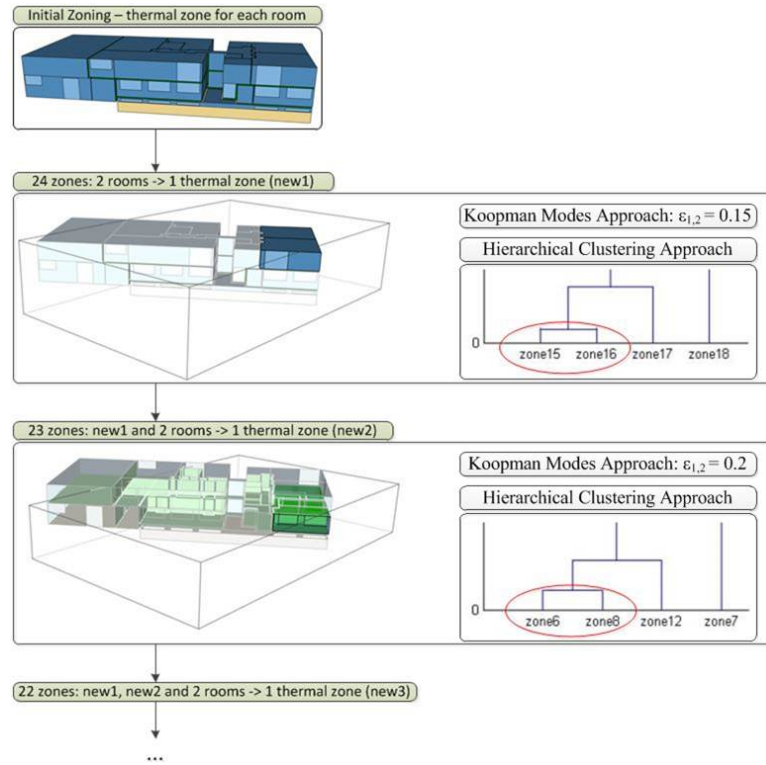


FIGURE 5.13: Zoning approximations for the first two steps of Hierarchical Clustering Approach and the respective  $\epsilon_{1,2}$  values of Koopman Modes Approach in CARTIF Building, using as data-set variables the zones’ operative temperatures

A set of experiments were performed for each Zoning Reduction Approach, selecting as data-set the simulated zones’ air temperature. A whole year simulation time interval is selected, while assuming all zones to be free floating, the conditions during this period are as follows: shading devices are completely open; internal gains are equal to zero,

since lights and electric equipment are switched off; there is no HVAC system available to control the temperature of the zone; and natural ventilation is not considered.

Evaluating the proposed Automatic Process, the main positive result is that both Zoning Reduction Approaches lead to the same zoning approximations, confirming thereby the effectiveness of the Hierarchical Clustering Approach, since the Koopman Modes Approach has been verified in the context of a recent work [27]. For intuition on how those approaches reduce the number of zone, the 3D geometries of the 25 zones (the initial full-scale model), 24 zones and 23 zones models are depicted in Figure 5.13.

Although both Zoning Reduction Approaches are equally effective, the ease of use of the Hierarchical Clustering Approach, lies in the fact that the only input to the algorithm is the desired number of zones; in contrast, the difficulty of use of the Koopman Modes Approach, lies in the difficulty of selecting proper values of  $\epsilon_1$ ,  $\epsilon_2$  and the number of modes of interest, since such selection does not have a physical interpretation, but it is based on experience the modeler has with this Approach.

Towards investigating the impact of the generated zoning approximations on model accuracy, initially a uniform criterion is used, the building's Heating/Cooling (H/C) energy demands. Here, all zones of each evaluated speedup model are simulated for a whole year simulation time interval with heating and cooling. The accuracy of a speedup model is measured by comparing the total energy demands required to maintain the zones' air temperature at the the range  $[20, 25]^\circ\text{C}$ , while the conditions during this period are as follows: shading devices are completely open; internal gains are equal to zero, since lights and electric equipment are switched off; an ideal HVAC system is available at each zone to control the zone air temperature; and natural ventilation is not considered.

Moreover, the speed-up models are designed for the purpose of Control-Design. Control-Design requires repeated simulations for the prediction horizon in connection with a stochastic optimization algorithm; a prediction horizon of one day along with six days of the warming up phase, reveals a time interval of one week to be set for each simulation, and as such the simulation runtime of each zoning approximation is compared to that of the full-scale model, considering a one-week time interval.

As Table 5.5 shows, for the Koopman Modes Approach, setting the number of Koopman modes of interest to 10 constantly, as increased values of  $\epsilon_1$  and  $\epsilon_2$  are used to check if Equations 5.18 and 5.19 hold, more zones are determined to be sufficiently similar to each other, reducing the total number of zones. As the number of zones reduces, the calculated H/C energy demands' error increases, while the simulation runtime decreases. In words

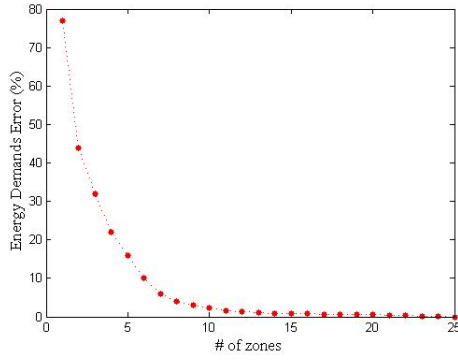


FIGURE 5.14: H/C energy demands' error in prediction as the number of zones increases

TABLE 5.5: Results of Zoning Reduction Approaches — Different Koopman modes tolerances  $\epsilon_{1,2}$  result to different H/C energy demands' error and simulation runtimes

# of zones	$\epsilon_{1,2}$	Error %	Runtime (sec)
25	0	0	382
20	0.3	0.5	296
15	0.68	0.8	223
10	0.7	2.3	160
5	0.83	16	115
1	1	77	92

of the Hierarchical Clustering Approach, as the desired number of zones decreases, the model's accuracy increases, while the simulation is less computationally expensive.

According to Figure 5.14, the H/C energy demands' error in prediction is inversely proportional to the number of zones.

The effectiveness of each zoning approximation to predict the zone's operative temperature, in terms of the root mean square error ( $rmse$ ) and the maximum deviation indicators ( $\delta_{\max}$ ), is also investigated, where both the full-scale model and a group of reduced models — specifically, 25 zones, 20 zones, 15 zones, 10 zones, 5 zones and 1 zone models — are simulated for a whole year time interval (run period).

Let  $h$  be a function that assigns to an index of zone in the original settings the index of the an aggregated zone. Let  $T_{\text{full-scale},k,i}$  be the zone  $i$  of the full-scale model operative temperature at time step  $k$  and  $T_{\text{reduced},k,h(i)}$  be the zone  $h(i)$  of the zoning reduced model, part of which is the zone  $i$  of the full-scale model, operative temperature. The  $rmse_i$  between the full-scale model consisting of  $N$  zones and a reduced model consisting of  $M < N$  simulation results is given by:

$$rmse_i = \sqrt{\sum_{k=1}^{n_{\text{data}}} (T_{\text{full-scale},k,i} - T_{\text{reduced},k,h(i)})^2 / n_{\text{data}}} \quad (5.20)$$

where  $n_{\text{data}}$  is the total number of timesteps (e.g.  $n_{\text{data}} = 52560$ , for a whole year simulation with 10min timestep), while the  $\delta_{\max,i}$  is calculated as follows:

$$\delta_{\max,i} = \max_k |T_{\text{full-scale},k,i} - T_{\text{reduced},k,h(i)}|. \quad (5.21)$$

Table 5.6 summarize the change in model predictive capability of operative temperature as the number of zones is reduced. The *rmse* and the  $\delta_{\max}$  results, clearly confirm what was discussed above. The accuracy of the model decreases exponentially as the number of zones is reduced. Thus, a critical point in the number of zones exists, beyond which accuracy does not substantiate the use of zonal models as surrogates of real buildings. Any number of zones greater than this critical point is acceptable, but the following trade-off is obvious: for more accurate results much more number of zones must be provided and calculations are more time demanding, therefore it should carefully be considered which is the computational complexity for a specific task and what is its required level of accuracy: one-size-fits-all does not apply here.

TABLE 5.6: Error in model predictive capability of zones' operative temperature as the number of zones is reduced, in terms of the *rmse* and the  $\delta_{\max}$

Initial IDF zone name	root mean square error					maximum deviation				
	20 zones	15 zones	10 zones	5 zones	1 zone	20 zones	15 zones	10 zones	5 zones	1 zone
zone 1	0.0235	0.0311	0.0452	0.3419	1.2311	0.0448	0.0647	0.1067	0.7687	2.2385
zone 2	0.0213	0.0319	0.3793	0.3945	0.8915	0.0454	0.0725	1.0913	1.0269	1.8302
zone 3	0.0064	0.0096	0.1487	0.5426	0.5339	0.0129	0.0192	0.4061	1.2614	1.7699
zone 4	0.1090	0.1692	0.1935	0.3842	1.5404	0.3179	0.4484	0.4627	0.8759	2.8924
zone 5	0.1338	0.1448	0.1172	0.2381	1.6115	0.3926	0.5088	0.4401	0.6902	2.6272
zone 6	0.1286	0.1306	0.2054	0.6142	2.5790	0.3137	0.3179	0.4579	1.2131	3.7374
zone 7	0.0739	0.1711	0.6941	0.5365	2.2538	0.1081	0.5436	1.3722	1.0363	3.8359
zone 8	0.0896	0.0981	0.1629	0.6103	2.5425	0.1795	0.2342	0.3973	1.2083	3.7845
zone 9	0.0776	0.2338	0.5129	0.4205	2.1229	0.1369	0.6879	1.2964	1.1483	3.0774
zone 10	0.0714	0.0844	0.4137	0.3204	2.5445	0.1229	0.2349	0.7189	0.8309	3.8221
zone 11	0.0685	0.3629	0.3684	0.3764	1.9367	0.1138	1.0128	1.0954	1.0858	3.8671
zone 12	0.1673	0.1843	0.2106	0.5963	2.5178	0.5787	0.6077	0.6199	1.2752	3.9272
zone 13	0.0411	0.1646	0.2002	0.2836	1.7620	0.0612	0.5328	0.6045	0.7692	3.3716
zone 14	0.0314	0.0523	0.6994	0.9008	1.1261	0.0822	0.1614	1.3544	1.8000	2.5878
zone 15	0.1681	0.2129	0.2276	0.3976	4.2139	0.3373	0.4042	0.5463	0.7732	5.3185
zone 16	0.1910	0.2376	0.2389	0.4282	4.1810	0.4409	0.5197	0.5949	0.9343	5.2351
zone 17	0.0889	0.2910	0.3275	0.4242	4.2731	0.2387	1.0085	1.0045	1.3040	5.3474
zone 18	0.0507	0.0689	0.0957	0.4654	4.7275	0.0720	0.1986	0.2732	1.2421	5.6199
zone 19	0.0533	0.2971	0.3355	0.9223	3.4709	0.0892	0.7344	0.8387	2.7311	5.4492
zone 20	0.1423	0.1908	0.2139	0.2363	2.9260	0.3835	0.4894	0.5486	0.8117	3.3700
zone 21	0.1327	0.1420	0.1839	0.3211	3.0695	0.3777	0.4468	0.5088	0.7459	3.6459
zone 22	0.0745	0.1264	0.3980	0.7728	2.6029	0.2533	0.4184	1.2235	2.5238	3.8112
zone 23	0.0344	0.0574	0.3464	0.8766	2.1281	0.0708	0.1519	0.7460	2.1886	2.5584
zone 24	0.0706	0.1387	0.1555	0.6730	3.1931	0.1320	0.3947	0.4241	2.3259	4.7794
zone 25	0.0141	0.0210	0.1499	0.2301	2.1884	0.0286	0.0426	0.2984	0.8617	3.2731

## Chapter 6

# Dynamic Data to BEP simulation

Improving BEP simulation models' accuracy by enriching them with sensed measurements of dynamic data is twofold: a) in-building sensed data can be forwarded to BEP simulation model through a co-simulation set-up; and b) in-situ measured weather data and/or weather predictions can be used to create the weather file required for the BEP simulation.

In this Chapter, Section 6.1 presents BCVTB as a co-simulation methodology, used during a Control-Design process, since that process representatively introduces the necessity of co-simulation towards improving the BEP simulation model's accuracy and designing effective controllers as well.

In Sections 6.2 - 6.4, Radiation Models briefly referred in Chapter 1, are presented. These models can be adopted and be performed as Global Solar Radiation Virtual Sensors (GSR-Virtual Sensor) or Direct/Diffuse Solar Radiation Virtual Sensors (DDSR-Virtual Sensor).

In many studies, for which results are presented in Section 6.5, the different established models and correlations that calculate hourly solar radiation components have been selected, performed and tested to decide which model is recommended.

The question of the best method for estimating the diffuse (or direct) fraction of global solar radiation are not fully settled.

### 6.1 Co-Simulation

The Control-Design process consists of a Warming-up and a Forecast Phase [28]. During the Warming-up Phase, a BEP simulation model of the building is used – combined with

past in-building sensor measurements and weather data – to estimate the actual thermal state of the building at the beginning of the Forecast Phase. Subsequently, when the Warming-up Phase finishes, the Forecast Phase initiates. Here, a stochastic optimization algorithm is used to solve a constraint optimization problem that requires minimization of the energy consumption, while preserving user thermal comfort levels [52, 53]. The optimization algorithm, starting from a provided initial controller constructs series of candidate controllers, which are evaluated using the “warmed-up” simulation model of the building along with weather and occupancy forecasts, to design a new controller.

According to [30], the Control Design methodology towards designing efficient controllers relies on the availability of accurate enough models, able to predict the thermal state of the building. Additionally, the proposed optimization iterative algorithms have to converge to a “good” controller after a finite number of iterations. Convergence-time heavily depends on the size of the problem at hand: for small problems convergence can be achieved after few iterations, while for more complex settings the number of iterations required increases. An increase in the number of iterations, allows algorithms to converge to better solutions, thus, simulation-time has to be minimum. To that direction, the reduced BEP simulation models, derived from approaches described in Chapter 5, are used.

With such a simulation model of the building at hand, the dynamic interaction between the model, past sensed and forecast data, and the Control-Design algorithm has to be defined using co-simulation, which enables the use of different software for run-time coupling. For the sake of example, the dynamic connection between EnergyPlus, where the model of the building has been developed, and Matlab, where the control logic has been implemented has to be effectively utilized. Such a connection can be achieved using EnergyPlus with External Interfaces and especially with the Building Controls Virtual Test Bed (BCVTB).

Hitherto, the BCVTB environment has been used as a co-simulation tool for boundary conditions data exchange in Section 5.1.2 and to allow the user to set a preferable order of BEP simulation outputs in Section 5.2.3. Here, the BCVTB is used as a central communication node that enables the coupling of different software codes for distributed simulation, by allowing simulation of the building envelope and HVAC system in EnergyPlus (or TRNSYS) and implementation of the control logic in Matlab (or other general purpose programming languages), facilitating dynamic data exchange between the two programs at each time step of the simulation.

Figure 6.1 shows the system architecture and the data exchange paths that establish BCVTB as the central communication node in EnergyPlus-Matlab connection. During the Warming-up Phase, a Matlab script requests past weather and in-building sensor

data from the real building, which forwards them to the EnergyPlus simulation model through BCVTB, thus using the one-way data exchange path, shown in the upper part of Figure 6.1. Consecutively, during the Forecast Phase a set of candidate controllers produced by the optimization algorithm are evaluated on the simulation model. Here, these controllers are implemented in Matlab and require information on specific building states, in order to produce control decisions in each simulation time-step. So, in every time-step of the simulation, a vector of building states (e.g. room temperature, outside humidity, etc.) is transmitted from the EnergyPlus model to the control logic in Matlab, through BCVTB. Subsequently, the new control decisions are communicated from Matlab back to EnergyPlus using the BCVTB (Figure 6.1). When the simulation ends, the performance of the control strategy, in terms of energy consumption and user comfort levels, is acquired by the optimization algorithm, again using the BCVTB (Figure 6.1).

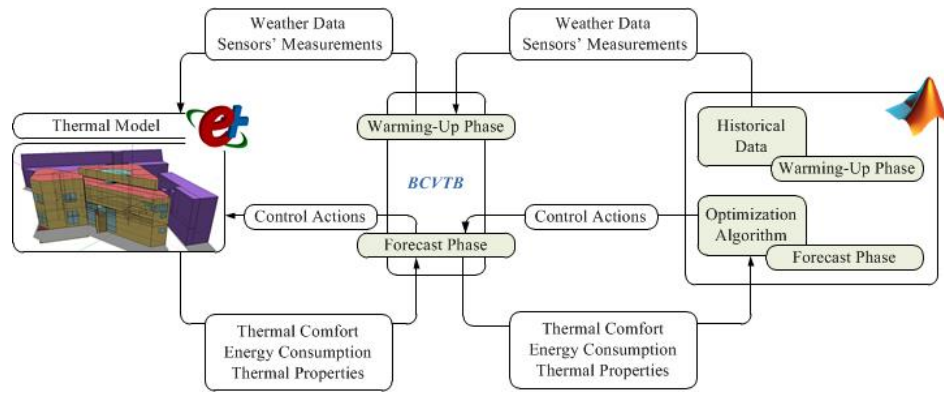


FIGURE 6.1: Architecture of the connection between EnergyPlus and the BCVTB and the connection between MATLAB and BCVTB during Control Design

Facilitating the controllers/simulation models coupling, the simplified simulation zonal (developed in EnergyPlus or TRNSYS following approaches presented in Chapter 5) models can be included as actors in BCVTB, as shown in Figure 6.2. Here, a Ptolemy model which is a Ptolemy II flow chart diagram is defined. Regarding the Simulator actor, configuring the flow chart diagram consists of three elements, enumerated in Figure 6.2:

1. *SDF Director*: In Ptolemy II, different models of computations can be used to define how the different actors interact with each other. The model of computation is defined by a director that needs to be included in the Ptolemy II flow chart diagram. For the BCVTB, the Synchronous Data Flow (SDF) Director is used.
2. *beginTime, timeStep, endTime*: These three parameters have units of seconds and needs to be equal to the start time, time step and final time that are used in the simulation program. These parameters are used to configure the SDF Director.

3. *Simulator actor*: The Simulator actor conducts the data exchange with the simulation program. Here, three types of data exchange need to be configured: (1) EnergyPlus - BCVTB; (2) TRNSYS - BCVTB; and (3) Matlab - BCVTB data exchange. The requirements' definition of such configurations is the main topic of this Section and described below.

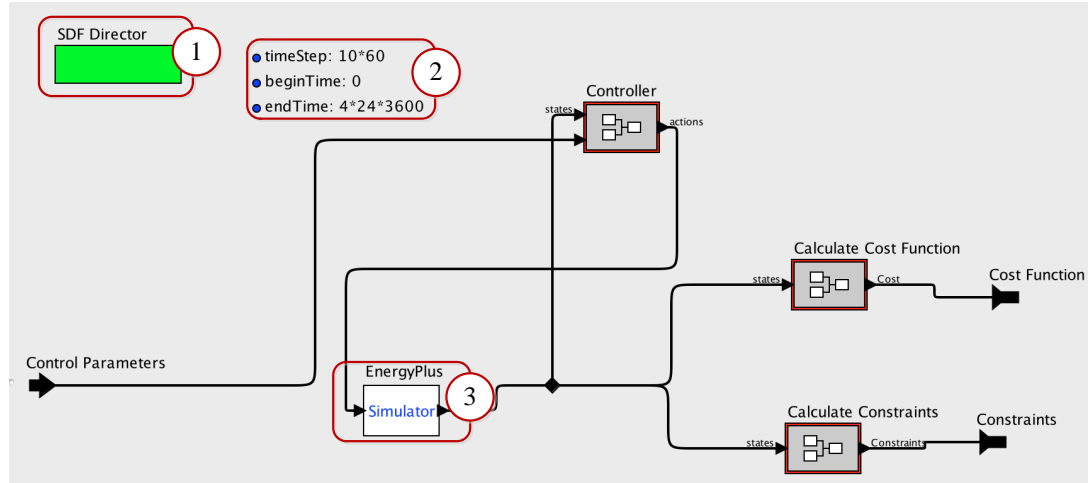


FIGURE 6.2: Integration of the Control Design and the Simplified Simulation Models within BCVTB

### 6.1.1 EnergyPlus – BCVTB data exchange

To configure the data exchange between EnergyPlus and BCVTB, the following three steps are required:

- Enrich the simplified IDF with the essential objects of the variables we would like to exchange.
- Develop an xml file, named `variables.cfg`, that defines the mapping between EnergyPlus and BCVTB variables.
- Define a Simulator actor to the Ptolemy II model.

#### 6.1.1.1 IDF file

With the simplified IDF at hand, modification of it must be performed in order to achieve the EnergyPlus — BCVTB data exchange. The first object needs to be filled is an *ExternalInterface* class object, in which the name property is set to PtolemyServer

so as to activate the BCVTB. Objects of the *Output:Variable* class can be used to send data from EnergyPlus to the BCVTB, while objects of the following IDD classes can be used to receive data from the BCVTB, at each zone time step:

1. ExternalInterface:Schedule;
2. ExternalInterface:Actuator;
3. ExternalInterface:Variable.

#### 6.1.1.2 XML file

The data mapping between EnergyPlus thermal model and the BCVTB is defined in an xml file called variables.cfg. This file needs to be in the same directory as the EnergyPlus idf file.

The file begins with the following line:

```
<?xml version="1.0" encoding="ISO-8859-1"? ><!DOCTYPE BCVTB-variables SYSTEM "variables.dtd">
```

<BCVTB-variables> and </BCVTB-variables> define the start and the end of the xml file, respectively. In between them, we need to specify how the exchanged data is mapped to EnergyPlus objects. They contain all the elements which define the variable mapping. The order of the elements which are defined between them matters and it needs to be the same as the order of the elements in the input and output signal vector of the BCVTB actor that calls EnergyPlus.

The exchanged variables are declared in elements that are called “variable” and have a corresponding source. The BCVTB can send data to objects of three classes, *ExternalInterface:Schedule*, *ExternalInterface:Actuator* and *ExternalInterface:Variable*. For these objects, the source needs to be set to Ptolemy, because they are computed in Ptolemy. The xml elements for these objects look as follows:

For the ExternalInterface:Schedule objects, NAME needs to be the same as the EnergyPlus schedule name.

```
<variable source="Ptolemy"><EnergyPlus schedule="NAME" /></variable>
```

For the ExternalInterface:Actuator objects, NAME needs to be the same as the EnergyPlus actuator name.

```
<variable source="Ptolemy"><EnergyPlus actuator="NAME" /></variable>
```

For the ExternalInterface:Variable objects, NAME needs to be the same as the EnergyPlus Energy Runtime Language variable name.

```
<variable source="Ptolemy"><EnergyPlus variable="NAME" /></variable>
```

As we discussed above, the BCVTB can also read data from any Output:Variable object. For each object, the source attribute is set to EnergyPlus, because it is computed by EnergyPlus. The xml elements for each object look as follows:

```
<variable source="EnergyPlus"><EnergyPlus name="NAME" type="TYPE" /></variable>
```

Output:Variable: NAME needs to be the same as the EnergyPlus “Variable Name” and TYPE needs to be the same as the corresponding EnergyPlus “Key Value”.

### 6.1.1.3 Simulator actor – EnergyPlus

A Simulator actor is used to conduct the data exchange with EnergyPlus. The parameters of the Simulator actor are as shown in Figure 6.3 and a short description is provided below:

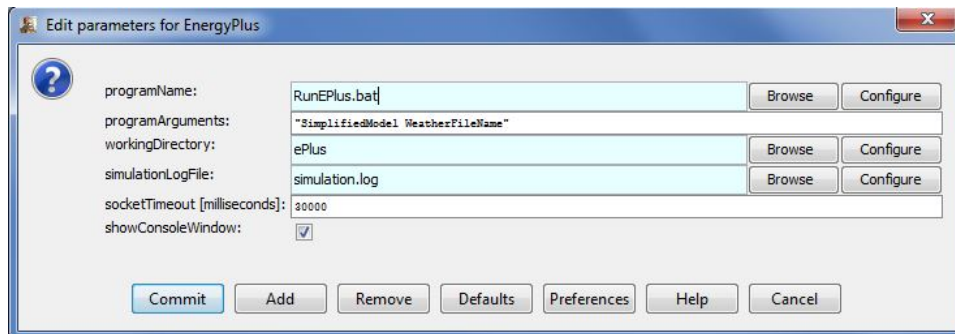


FIGURE 6.3: Simulator actor – EnergyPlus and its parameters

- **programName:** The name of the executable that starts the simulation is RunEPlus.bat.
- **programArguments:** Arguments needed by EnergyPlus. In this field we set the idf file name (SimplifiedModel) and the weather file name (WeatherFileName).
- **workingDirectory:** Working directory of EnergyPlus is ePlus folder.
- **simulationLogFile:** Name of the file to which the BCVTB will write the console output and error stream that it receives from EnergyPlus. This file typically shows what may have caused an error.

- `socketTimeout`: Time out in milliseconds for the initial socket connection. At the start of the simulation, the BCVTB waits for EnergyPlus to connect through a socket connection to the BCVTB. If EnergyPlus does not connect within the here specified time, the BCVTB will stop with an error.

### 6.1.2 Matlab – BCVTB data exchange

To configure the data exchange between Matlab and BCVTB, the following three steps are required:

- Create an m file, named `simulateAndExit.m` preferably, to determine the data exchange between the MATLAB and the BCVTB variables.
- Define a Simulator actor to the Ptolemy II model.

#### 6.1.2.1 M file

The `simulateAndExit.m` file has the following structure:

1. Initialize variables
2. Add path to BCVTB Matlab libraries:

```
addpath( strcat(getenv('BCVTB_HOME'),'/lib/matlab'));
```

3. Establish the socket connection:

```
sockfd = establishClientSocket('socket.cfg');
```

4. Exchange data (called at each timestep):

```
[retVal, flaRea, simTimRea, dblValRea] = ...  
... exchangeDoublesWithSocket(sockfd, flaWri, nDblRea, simTimWri, dblValWri);
```

The input arguments are:

- `sockfd`: Socket file descriptor.
- `flaWri`: Communication flag to write to the socket stream. It is set to zero for normal operation, or to a negative value to stop the exchange.
- `nDblRea`: Number of double values which will be read.
- `simTimWri`: Current simulation time in seconds to write to BCVTB.
- `dblValWri`: Vector of double values to write to BCVTB.

The return values are:

- retVal: Has to be a non negative value if the data exchange was successful or a negative value if an error occurs.
  - flaWri: communication flag read from the socket stream. A negative value indicates that the BCVTB will stop due to an error and not send any more data. A value equal to zero indicates normal operation and a value equal to one means that the final simulation time has been reached and no more data will be exchanged.
  - simTimRea: current simulation time in seconds read from the socket.
  - dblValRea: vector of double values read from the socket.
5. Close socket at the end of the simulation:  
`closeIPC(sockfd);`
  6. Exit Matlab:  
`exit`

### 6.1.2.2 Simulator actor – Matlab

A Simulator actor is used to conduct the data exchange with Matlab. The parameters of the Simulator actor are shown in Figure 6.4 and a short description is provided below:

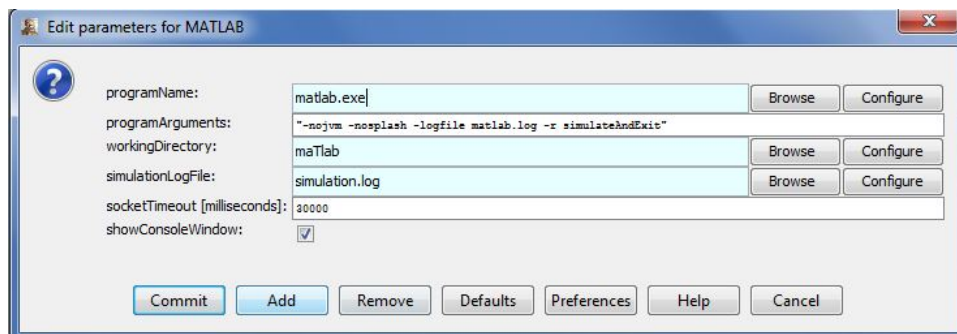


FIGURE 6.4: Simulator actor – Matlab and its parameters

- programName: The name of the executable that starts the simulation is matlab.
- programArguments: Arguments needed by the simulation. Text arguments need to be enclosed in apostrophes.
- workingDirectory: Working directory of Matlab is maTlab folder.
- simulationLogFile: Name of the file to which the BCVTB will write the console output and error stream that it receives from Matlab. This file typically shows what may have caused an error.

- `socketTimeout`: Time out in milliseconds for the initial socket connection. At the start of the simulation, the BCVTB waits for Matlab to connect through a socket connection to the BCVTB. If EnergyPlus does not connect within the here specified time, the BCVTB will stop with an error.

### 6.1.3 TRNSYS - BCVTB data exchange

Any TRNSYS simulation model is developed in the Simulation Studio (user interface of TRNSYS). The Simulation Studio creates not only the trnsys project file (tpf), but also the basic input file, named deck file (dck), a text file that contains all the information on the simulation but no graphical information. To configure the data exchange between TRNSYS and BCVTB, the following three steps are required:

- Enrich the simplified tpf with the BCVTB component (Type 6666).
- Create the dck file
- Define a Simulator actor to the Ptolemy II model.

#### 6.1.3.1 BCVTB component and dck file

The BCVTB component (Type 6666) controls how the variables are communicated between TRNSYS and the BCVTB. There are 3 parameters need to be defined: the number of variables passed to the BCVTB; the number of variables received from the BCVTB; and the number of TRNSYS timesteps per data exchange with the BCVTB. For the data mapping between TRNSYS thermal model and the BCVTB, the usual TRNSYS linking process is used.

Once the BCVTB component is added to the tpf file, the dck file for the project must be created. The BCVTB uses the dck file directly to run the TRNSYS simulation and not the Studio project file (tpf file). The dck file is written by the pen icon on the left side of the Simulation Studio window.

#### 6.1.3.2 Simulator actor – TRNSYS

A Simulator actor is used to conduct the data exchange with TRNSYS. The parameters of the Simulator actor are shown in Figure 6.5 and a short description is provided below:

- `programName`: The name of the executable that starts the simulation is `runTRNSYS.bat`.

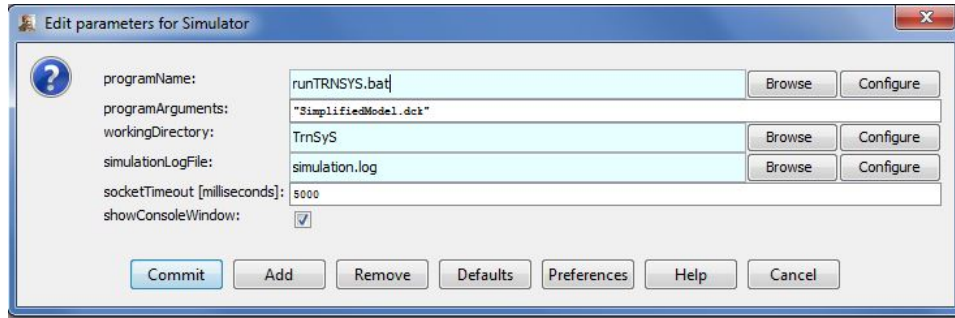


FIGURE 6.5: Simulator actor – TRNSYS and its parameters

- **programArguments:** Arguments needed by TRNSYS. In this field we set the dck file name (SimplifiedModel).
- **workingDirectory:** Working directory of TRNSYS is TrnSyS folder.
- **simulationLogFile:** Name of the file to which the BCVTB will write the console output and error stream that it receives from TRNSYS. This file typically shows what may have caused an error.
- **socketTimeout:** Time out in milliseconds for the initial socket connection. At the start of the simulation, the BCVTB waits for TRNSYS to connect through a socket connection to the BCVTB. If TRNSYS does not connect within the here specified time, the BCVTB will stop with an error.

#### 6.1.4 Experiment

Between numerous experiments that have been conducted on TUC building [51], two of them are selected to be presented here, highlighting the necessity of the co-simulation setup within the model-assisted control design context and towards improving the BEP simulation model's accuracy as well.

The first experiment indicates the significance of the Warming-up Phase for the simulation model accuracy. Here, the simulation model is provided with historical weather and in-building sensor data for 13 days (from December 10th to December 30th) and exploits them to assimilate the thermal state of the building at the beginning of the 14th day. After that, and for about 4 days, the real building is unoccupied and allowed to free-float (i.e. no actuating components are operated), while the simulated model is required to accurately predict the zone temperature values (model validation). Note here that during the warm-up the sensed air temperature of each office room is set as the thermostat temperature setpoint of the room. As Figure 6.7 depicts for zone O4 (see Figure 6.6), during the warming-up period simulated and measured temperature trends are (almost)

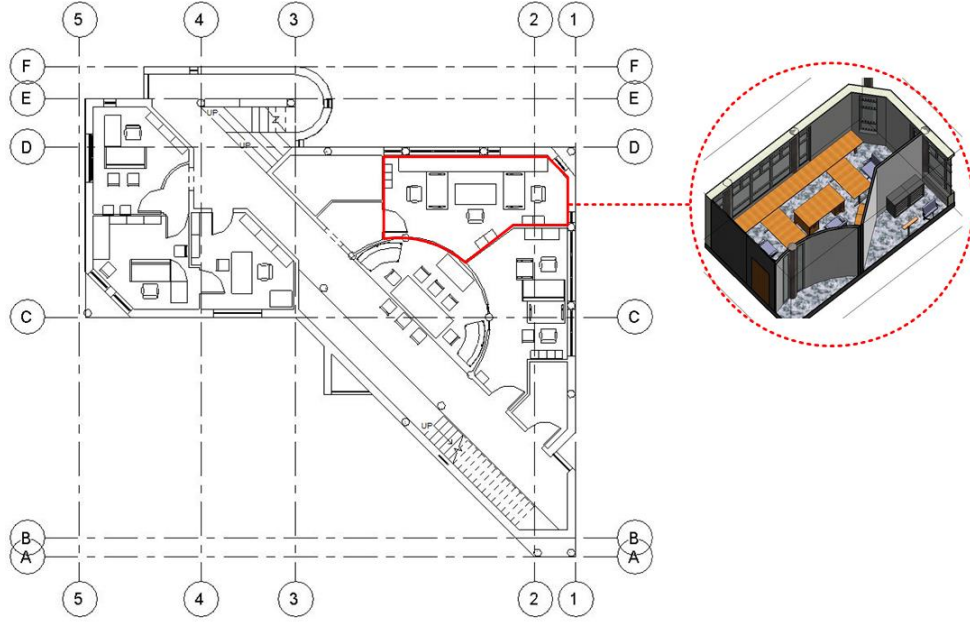


FIGURE 6.6: Floor plan of TUC building and test zone O4 (office 4)

identical, thus the historical sensor values provided to the simulation model through the middleware, using the co-simulation setup, enhance the accuracy of the model. As for the validation phase (days 14-17), the results on Figure 6.7 show that the maximum temperature difference between the simulated and real temperature schedules is  $0.61^{\circ}\text{C}$ , while the mean absolute error  $0.25^{\circ}\text{C}$ , indicating high model accuracy.

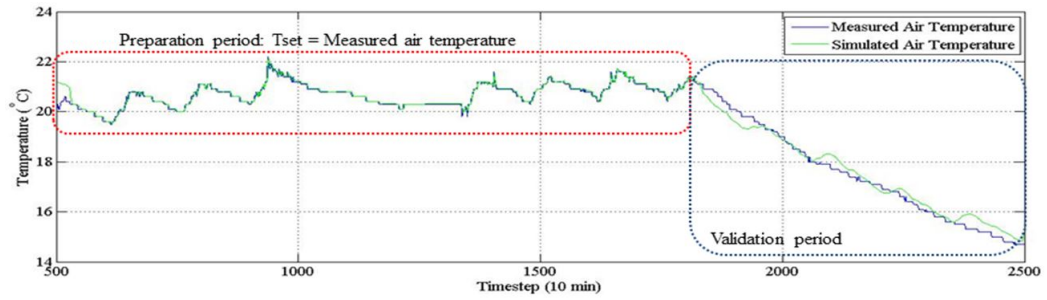


FIGURE 6.7: Warming up period and validation period: Comparison of simulated inside temperatures and measured inside temperatures for zone O4

The second experiment presents the quality of the controllers produced by applying the overall methodology to the real building. Here, for a hot summer weekend where the outside temperature rises up to  $27^{\circ}\text{C}$  during the day and drops as low as  $20^{\circ}\text{C}$  during the night, a new controller is designed using the Control Design process every 2 hours, while the produced controllers are applied to the real building every 10 minutes. The final control strategy applied to zone O4 is shown in Figure 6.8(a) along with the predicted room temperature values, while in Figure 6.8(b) the actual room conditions are presented. A closer look on the results reveals the intelligent behavior of the control strategy generated by the proposed approach. To start, the use of occupancy information

allows identifying unoccupied periods, thus shutting-down the cooling system. Moreover, the algorithm identifies that during the morning less cooling power is required and follows a conservative cooling policy, while during noon lowers the setpoint to maintain acceptable comfort levels in more demanding conditions. This behavior stems from the enhanced model accuracy, due to the incorporation of past and forecast data through co-simulation. This enhances the evaluation accuracy of candidate controllers tested by the optimization algorithm, thus assisting towards better control design.

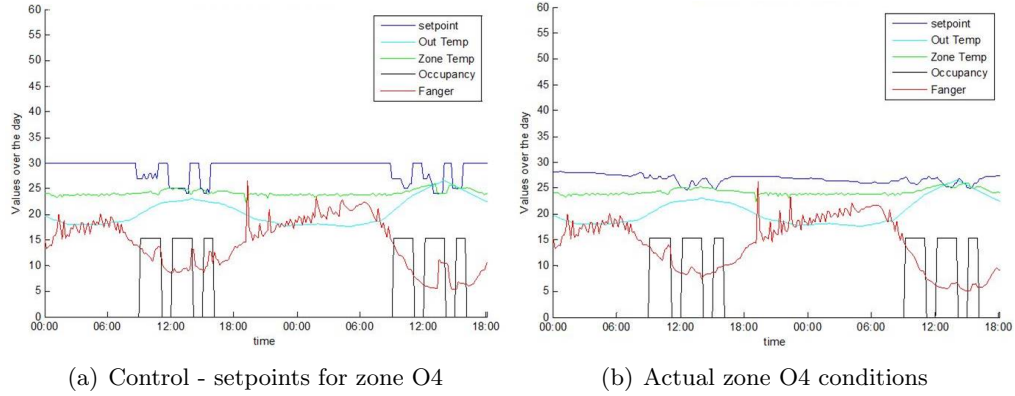


FIGURE 6.8: Control Design results – TUC building

## 6.2 GSR-Virtual Sensor: Global Solar Radiation prediction based on percentage of cloud coverage data

Towards the global solar radiation prediction from available cloud coverage data, the forecasting model described in [47] and the references within are used, where the global horizontal irradiance is determined by:

$$\frac{I}{I_{clear}} = 1 - 0.75(N/8)^{3.4}. \quad (6.1)$$

Here,  $N$  is the cloud cover in octas and  $I_{clear}$  is the clear sky global irradiance.

Beam clear sky irradiance could be calculated according to ASHRAE clear sky model [2]. According to ASHRAE clear sky model the beam clear sky irradiance received by a surface with orientation normal to the sun rays follows an exponential model:

$$I_{b,clear} = A \exp\left(-\frac{B}{\sin SE}\right) \quad (6.2)$$

Coefficients A,B are given by for each month in table 6.1.

TABLE 6.1: ASHRAE clear sky model coefficients [2]

Month	A	B	C
January	1202	0.141	0.103
February	1187	0.142	0.104
March	1164	0.149	0.109
April	1130	0.164	0.120
May	1106	0.177	0.130
June	1092	0.185	0.137
July	1093	0.186	0.138
August	1107	0.182	0.134
September	1136	0.165	0.121
October	1166	0.152	0.111
November	1190	0.144	0.106
December	1204	0.141	0.103

Additionally, the diffuse horizontal clear sky irradiance on a horizontal surface is given by:

$$I_{d,clear} = C I_{b,clear} \quad (6.3)$$

Coefficient C is also given in Table 6.1.  $SE$  is the solar elevation angle.

Adding complexity towards properly determining the clear sky global horizontal irradiance, various models exist [87]. For example, when Linke turbidity factor is available, instead of the ASHRAE clear sky model, the model adopted by [45] is used. Here, the global clear sky irradiance is quantified by the Linke turbidity factor as follows:

$$I_{clear} = 0.84 I_o \cos(Z) \exp(-0.02 m_a (f_1 + (TL - 1) f_2)), \quad (6.4)$$

with,  $f_1 = \exp(-alt/8000)$ ,  $f_2 = \exp(-alt/1250)$  and  $I_o$  be the extraterrestrial normal,  $Z$  be the solar zenith angle,  $m_a$  be the optical air mass,  $alt$  be the elevation above the sea level in meters and  $TL$  be the Linke turbidity factor which denotes the transparency of the cloudless atmosphere. A typical value of  $TL$  for Europe is 3. However, this value exhibits strong fluctuations in space and time. Thus, monthly values of  $TL$  factor, gathered from a publicly available web service (<http://www.soda-is.com>) for a specific location (TUC building's location) are presented in Table 6.2.

TABLE 6.2: Monthly values of Linke turbidity factor for TUC Building location

Month	Jan	Feb	Mar	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
<b>TL</b>	2.5	2.4	3	3.7	3.5	3.6	3.8	3.4	2.8	2.6	2.6

The optical air mass is estimated as derived by [47] (see [87] for more models quantifying the optical air mass)

$$m_a = \frac{p/p_o}{\cos Z + 0.50572 (96.07995 - Z \cdot 180/\pi)^{-1.6364}}. \quad (6.5)$$

Here,  $p/p_o$  is the atmospheric pressure ratio and it is estimated by:

$$p/p_o = \exp\left(-\frac{alt}{8434.5}\right). \quad (6.6)$$

The zenith angle can be found by the following equation:

$$\cos Z = \sin \phi \cdot \sin D + \cos \phi \cdot \cos D \cdot \cos \omega; \quad (6.7)$$

where,  $\phi$  is the local latitude,  $D$  is the solar declination and  $\omega$  is the hour angle. The solar declination's calculation is based on the equation of Cooper:

$$D = 23.45 \cdot \sin\left(\frac{360 \cdot (284 + n)}{360}\right). \quad (6.8)$$

Here,  $n$  is the day of the year and the hour angle  $\omega$ :

$$\omega = \pm 1/4nm. \quad (6.9)$$

The angle is negative in the morning and positive in the afternoon. Here,  $nm$  is the number of minutes from local solar noon.

The previous angles are part of a set of angles which are used to describe the position of the sun and are derived by [24]. According to the forecasting solar radiation model presented above, these angles are used to calculate the value of zenith angle. Thereupon, the value of zenith angle is substituted into Equation 6.5 and hence, the optical air mass is estimated while, given the Linke turbidity factor determined by Table 6.2, it is possible to calculate the clear sky global irradiance through Equation 6.4. Finally, the result from Equation 6.4 is substituted into Equation 6.1 to obtain the value of the cloudy sky global irradiance, given the forecasted cloud coverage.

### 6.3 DDSR-Virtual Sensor: Diffuse fraction of global solar radiation models

Existing polynomial correlation models are usually expressed in terms of first to forth order polynomial functions and relate the diffuse fraction of global solar radiation with the clearness index. The total range of clearness index is divided into three intervals.

$$f_d = \begin{cases} a_1 + a_2 \cdot k_t & \text{first range of } k_t \\ a_3 + a_4 \cdot k_t + a_5 \cdot k_t^2 + a_6 \cdot k_t^3 + a_7 \cdot k_t^4 & \text{second range of } k_t \\ a_8 & \text{third range of } k_t. \end{cases} \quad (6.10)$$

Table 6.3 presents the values of coefficients  $a_i$  that are used for eight different polynomial models. However, there are many other studies that are trying to calibrate these coefficients.

TABLE 6.3: Diffuse fraction-polynomial models coefficients and their second range of clearness index

Polynomial Model	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	$a_8$	2nd range of $k_t$
Orgill & Hol.	1	-0.249	1.557	-1.84	0	0	0	0.177	$0.35 \leq k_t \leq 0.75$
Reindl	1.02	-0.248	1.45	-1.67	0	0	0	0.147	$0.3 < k_t < 0.78$
Hawladar	0.915	0	1.135	-0.942	-0.388	0	0	0.18	$0.225 < k_t < 0.775$
Miguel	0.995	-0.081	0.724	2.738	-8.32	4.96	0	0.19	$0.21 < k_t < 0.76$
Karatasou	0	0	0.999	-0.05	-2.415	1.492	0	0.78	$0 < k_t \leq 0.78$
Jacovides et al.	0.987	0	0.94	0.937	-5.01	3.32	0	0.177	$0.1 < k_t \leq 0.8$
Erbs	1	-0.09	0.951	-0.164	4.388	-16.638	12.336	0.165	$0.22 < k_t \leq 0.8$
Oliveira	1	0	0.97	0.8	-3	-3.1	5.2	0.17	$0.17 < k_t < 0.75$

In [86] apart from the polynomial model shown in Table 6.3, the authors described a model relating the diffuse fraction not only with the clearness index but also with the sun altitude ( $SE$ ), the ambient air temperature ( $T_\infty$ ) and the relative humidity ( $H_\infty$ ). In this approach the diffuse fraction is calculated as follows:

$$f_d = \begin{cases} 1.0 - 0.232k_t + 0.0239 \sin SE - 0.000682T_\infty + 0.019H_\infty & 0 \leq k_t \leq 0.3 \\ 1.329 - 1.716k_t + 0.267 \sin SE - 0.00357T_\infty + 0.106H_\infty & 0.3 < k_t \leq 0.78 \\ 0.426k_t + 0.256 \sin SE - 0.00349T_\infty + 0.0734H_\infty & \text{else.} \end{cases} \quad (6.11)$$

Analyzing the functional relationship between the experimental diffuse fraction and the clearness index, observed data are commonly filtered by a moving average method. Using a moving average method Bolland and Ridley [14] proposed a mathematical model,

described by Equation 6.12. It is obvious that in the proposed model only two parameters are needed.

$$f_d = \frac{1}{1 + e^{b_1(k_t + b_2)}} \quad (6.12)$$

Bolland and Ridley [14] claimed that the values 8.6 and -0.581 of  $b_1$  and  $b_2$  respectively, can have a general validity.

As mentioned earlier, in Ridley et al. model [88], a persistence index is introduced. Here, Equation 6.13 is used to obtain the diffuse fraction.

$$f_d = \frac{1}{1 + e^{-5.38 + 6.63k_t + 0.006AST - 0.007SE + 1.75K_t + 1.31\psi}}. \quad (6.13)$$

$AST$  is the apparent solar time, in decimal hours;  $k_t$  is the hourly clearness index;  $K_t$  is the daily clearness index;  $SE$  is the solar elevation in degrees and  $\psi$  is the persistence index, calculated with (6.14).

$$\psi = \begin{cases} \frac{k_{t-1} + k_{t+1}}{2} & \text{sunrise} < t < \text{sunset} \\ k_{t+1} & t = \text{sunrise} \\ k_{t-1} & t = \text{sunset}. \end{cases} \quad (6.14)$$

## 6.4 DDSR-Virtual Sensor: Direct Irradiance models

Maxwell model [66] is a quasi-physical model in which the direct irradiance is estimated as follows:

$$I_b = I_o[\lambda - (d_4 + d_5 e^{m_a d_6})]; \quad (6.15)$$

where,

$$\lambda = 0.866 - 0.122m_a + 0.0121m_a^2 - 0.000653m_a^3 + 0.000653m_a^4 + 0.000014m_a^5; \quad (6.16)$$

$$m_a = m_r \frac{p}{1013.25}; \quad (6.17)$$

and the air mass at standard pressure approximated by Kasten's formula [46]. as:

$$m_r = \frac{1}{\cos \theta_z + 0.15(93.885 - \theta_z)^{-1.253}} \quad (6.18)$$

Coefficients  $d_4$ ,  $d_5$  and  $d_6$  are functions of the clearness index, derived by Equations 6.19 – 6.21.

$$d_4 = \begin{cases} 0.512 - 1.56k_t + 2.286k_t^2 - 2.222k_t^3 & k_t \leq 0.6 \\ -5.743 + 21.77k_t - 27.49k_t^2 + 11.56k_t^3 & k_t > 0.6 \end{cases}, \quad (6.19)$$

$$d_5 = \begin{cases} 0.37 + 0.962k_t & k_t \leq 0.6 \\ 41.4 - 118.5k_t + 66.05k_t^2 + 31.9k_t^3 & k_t > 0.6 \end{cases}, \quad (6.20)$$

$$d_6 = \begin{cases} -0.28 + 0.923k_t - 2.048k_t^2 & k_t \leq 0.6 \\ -47.01 + 184.2k_t - 222k_t^2 + 73.81k_t^3 & k_t > 0.6 \end{cases}. \quad (6.21)$$

In [92] the authors described a model relating the direct irradiance with the clearness index, the sun altitude ( $SE$ ), the ambient air temperature ( $T_\infty$ ), the relative humidity ( $H_\infty$ ), and the monthly average global radiation ( $G$ ). In this approach the direct irradiance is calculated as follows:

$$I_b = \frac{G(1 - \lambda)}{\sin SE}, \quad (6.22)$$

where,

$$\lambda = \begin{cases} 1 & k_t < c_1 \\ 1 - (1 - d_1)[d_2\sqrt{c_3} + (1 - d_2)c_3^2] & c_1 \leq k_t \leq 1.09c_2 \\ 1 - 1.09c_2 \frac{1-\xi}{k_t} & k_t > 1.09c_2 \end{cases}. \quad (6.23)$$

Coefficients  $d_1, d_2, d_3, c_1, c_2, c_3$  and  $\xi$  are calculated by Equations 6.24 – 6.31.

$$c_1 = 0.2, \quad (6.24)$$

$$c_2 = 0.87 - 0.56e^{-0.06a}, \quad (6.25)$$

$$c_4 = k_t - c_1, \quad (6.26)$$

$$d_1 = 0.15 + 0.43e^{-0.06a}, \quad (6.27)$$

$$d_2 = 0.27, \quad (6.28)$$

$$d_3 = c_2 - c_1, \quad (6.29)$$

$$c_3 = 0.5[1 + \sin(\pi(c_4/d_3 - 0.5))], \quad (6.30)$$

$$\xi = 1 - (1 - d_1)(d_2 c_3 / 2 + (1 - d_2) c_3^2). \quad (6.31)$$

The Dirint model [78] is a modification of the Maxwell model. In this case, the direct irradiance calculated by the Maxwell model ( $I_b$ ) is multiplied by a coefficient from a 6 x 6 x 5 x 7 look-up table ( $X(k'_t, Z, W, \Delta k'_t)$ ) that depends on the following parameters: the clearness index, the sun elevation, the dew point temperature and a variability index for considering the dynamics of the process. This model presents two operational modes: on the one hand, the 4-D and on the other hand, the 3-D, which does not need the dew point temperature data.

The basic expression used to obtain the normal direct irradiance ( $I_{b,Dirint}$ ) is described in Equation 6.32.

$$I_{b,Dirint} = I_b \cdot X(k'_t, Z, W, \Delta k'_t); \quad (6.32)$$

where,  $k'_t$  is the correction of the clearness index to make it independent on the sun's position and estimated by Equation 6.33,  $Z$  is the solar zenith angle,  $W$  is the precipitable water in the atmosphere (obtained from the dew point temperature and not used in the 3-D version of the model), and  $\Delta k'_t$  an index that depends on the  $k'_t$  values corresponding to the previous, current and following hour (see Equation 6.34).

$$k'_t = \frac{k_t}{1.031e^{-1.4/(0.9+9.4/m)} + 0.1}; \quad (6.33)$$

$$\Delta k'_t = 0.5(|k'_{t_i} - k'_{t_{i+1}}| + |k'_{t_i} - k'_{t_{i-1}}|). \quad (6.34)$$

## 6.5 Solar Radiation Models: Comparative studies

Most existing work on diffuse fraction models' comparison and validation has been based on weather stations' data from North America, Canada and Australia. Nevertheless, in

this section models' comparison studies in the European countries are presented.

In the Jacovides et al. study [41], polynomial correlations were developed to establish a relationship between the hourly diffuse fraction and the hourly clearness index using hourly global and diffuse irradiation measurements on a horizontal surface performed at Athalassa, Cyprus. The proposed correlations were compared against ten models available in the literature in terms of the root mean square error (*rmse*), mean bias error (*mbe*) and *t* – *test* indicators:

$$rmse = \frac{100}{\bar{D}} \sqrt{\sum_{i=1}^N (D_{im} - D_{ir})^2 / N}; \quad (6.35)$$

$$mbe = \frac{100}{\bar{D}} \sum_{i=1}^N (D_{im} - D_{ir}) / N; \quad (6.36)$$

$$t_{stat} = \sqrt{\frac{(N-1)mbe^2}{rmse^2 - mbe^2}}; \quad (6.37)$$

where,  $N$  is the number of data,  $D_{ir}$  is the  $i$ th estimated value,  $D_{im}$  is the  $i$ th measured value and  $\bar{D}$  is the mean of the measured values. The analysis was based on hourly radiation data collected at Athalassa (Cyprus) for a 5-year period (1 January 1998–31 December 2002). Results from this analysis, are presented in Table 6.4 and Figure 6.9.

TABLE 6.4: Results of the Jacovides et al. hourly model's comparison study [41]

Models	mbe (%)	rmse (%)	$t_{stat}$
Orgill and Hollands	4.47	30.2	0.638
Reindl et al.	3.93	29.5	0.88
Boland et al.	6.95	30.5	0.923
Hawladar	1.88	29.2	0.697
Miguel et al.	4.52	29.8	0.893
Karatasou et al.	1.38	29.2	0.719
Erbs et al.	3.5	30.9	0.664
Chandrasekaran and Kumar	6.15	30.6	1.214
Oliveira	-2.82	29.4	1.005
Soares	-8.21	30.9	1.263
Jacovides et al.	1.22	28.4	0.835

Table 6.4 indicates that the Karatasou et al. [43], Hawladar et al.[31] and Oliveira et al. [76] models provide the best (lower) *mbe* and *rmse* values. Figure 6.9(a) shows that, a good agreement is indicated between the first-order correlations [14, 77, 86] and the measured diffuse fraction values. However, these models tend to underestimate diffuse fraction values on  $k_t > 0.4$  values. On the other hand, as Figure 6.9(b) depicts, second-

and third-order correlations [19, 31, 43] give better results. Nevertheless, the Miguel et al. model [19], clearly disagrees with the measured values in the range  $0.25 \leq k_t \leq 0.60$ .

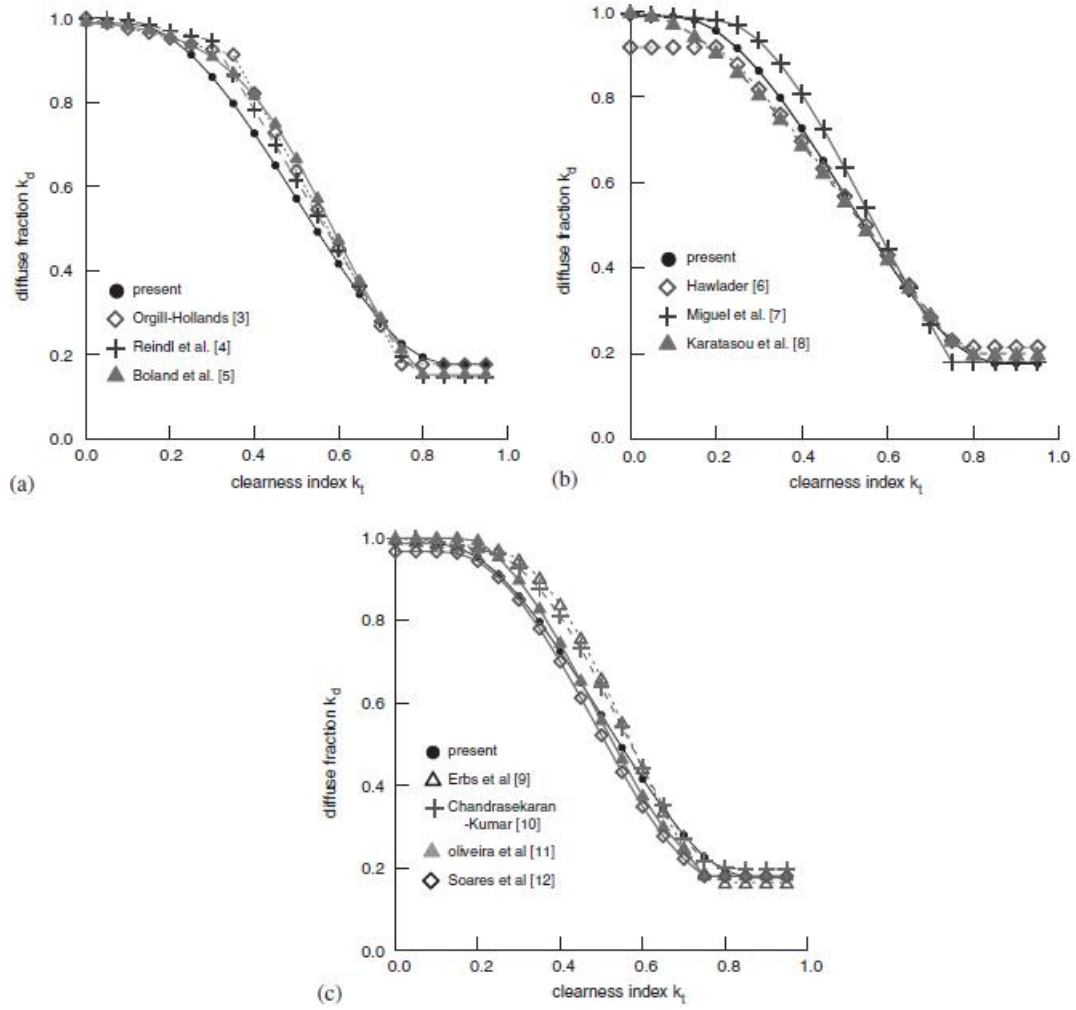


FIGURE 6.9: The hourly diffuse fraction vs clearness index for different polynomial correlations in Cyprus: (a) first-order; (b) second-third order; and (c) fourth-order polynomial correlations [41]

As Figure 6.9(c) shows, Oliveira et al. model [76] allows a better fitting with the measured data in the range  $k_t < 0.5$  while for larger values of  $k_t$ , the diffuse fraction tends to be underestimated. Apart from the Oliveira et al. model, the higher-order standard correlations concluded to higher *rmse* and *mbe* values. Nevertheless, the general conclusion of this study reveals that most of the polynomial correlations are equally accurate for hourly diffuse irradiation predictions.

Miguel et al. [19] selected, performed and tested different established models and correlations that calculate hourly and daily diffuse solar irradiation on horizontal surface to decide which model is recommended. Model and correlation studies were classified in three groups. In the first group, selected correlations which calculate daily diffuse

from global daily irradiation were tested. In the second, three models which calculate hourly diffuse from hourly global irradiation were performed and a new correlation was proposed. The third group contains four models which calculate hourly diffuse from daily diffuse irradiation values.

In our case, hourly global solar radiation's measurements (past weather sensed data) or estimations (based on cloud coverage prediction) are available and as such the conclusions of the second group is of our interest. Hence, in that group Maxwell [66], Hollands and Crha [35], Macagnan et al. [60] and Miguel et al. [19] models were tested and evaluated.

Measured values of hourly global and diffuse horizontal irradiation were collected from eleven different European cities, belonging to the Mediterranean belt area, for such a purpose: Athens (Greece); Lisbon (Portugal); Coimbra (Portugal); Evora (Portugal); Faro (Portugal); Porto (Portugal); Carpentras (France); Pau (France); Perpignan (France); Madrid (Spain); and Seville (Spain). The *rmse* and *mbe* were used to indicate how closely the correlation agrees with the data and their seasonal variation (see Equations 6.35 and 6.36).

TABLE 6.5: Results of the Miguel et al. hourly model's comparison study [19]

Correlations	Athens (Greece)		Porto (Portugal)		Seville (Spain)	
	rmse (%)	mbe (%)	rmse (%)	mbe (%)	rmse (%)	mbe (%)
Macagnan et al.	54.4	3.16	45.82	-14.6	45.69	-4.76
Hollands and Crha	53.89	29.54	36.04	2.73	38.05	14.75
Maxwell	56.09	17.71	51.43	-24.85	48.63	-7.84
Miguel et al.	58	11.81	36.25	-2.88	38.01	6.23

Some representative results are shown in Table 6.5. It is obvious that the Maxwell model's estimated values do not fit well the real data. Hollands and Crha's model obtained better results due to the fact that its parameters were recalculated, in order to fit the data of the locations belonging to the Mediterranean Belt. The comparison between statistical characteristics of the real and estimated data in the cases studied shows that the recalibration of the parameters allows an improvement in the performance of the model at the cost of narrow range of validity. With regard to the statistical characteristics, Hollands and Crha's original model is the best in reproducing the mean, the standard deviation and the standard error. For these reasons, the Hollands and Crha model is recommended, taking into account that a recalibration of the coefficients might substantially improve the performance of the model.

Torres et al. [99] presented a comparison among seventeen different proposals for estimating the hourly diffuse fraction of irradiance in Pamplona (Navarre); a city located in

the Spanish region. Twelve of them are polynomial correlations of different orders, two are based on an exponential function and three consider the diffuse irradiance values in the previous and posterior hour to that of the calculation.

For this research, experimental global, direct and diffuse irradiance data were collected by a Pamplona's weather station, from October 2006 up to May 2008.

TABLE 6.6: Results of the Torres et al. hourly model's comparison study [99]

Models	mbe (%)	rmse (%)
Orgill and Hollands	7.05	35.14
Reindl et al.	1.52	34.36
1st order calibrated	-1	36.1
Hawladar	-0.5	35.85
2nd order calibrated	-2.94	35.57
Miguel et al.	2.62	34.26
Karatasou et al.	-0.01	37.98
3rd order calibrated	-2.69	35.62
Erbs et al.	0.13	35
Oliveira	-5.47	37.43
Jacovides et al.	0.02	36.85
4th order calibrated	3.47	35.77
Boland et al.	5.49	35.51
Logistic function calibrated	5.49	35.51
Dirint	2.57	29.38
Skartveit et al.	5.98	30.13
Ridley et al.	1.49	31.4

Table 6.6 shows the absolute and percentage values of *mbe* and *rmse* resulting from comparing the experimental hourly average diffuse irradiance with the one calculated by applying the different analyzed models. The applied statistics indicate that an increase in the correlation order does not significantly improve the diffuse irradiance estimation. Models that consider the calibrated coefficients for the diffuse fraction in Pamplona (1st, 2nd, 3rd, 4th order and logistic function calibrated models in Table 6.6) do not give better results for the diffuse irradiance than their respective correlations of equivalent order obtained from the literature. Moreover, the logistic model does not improve the results of the correlations. Regarding the three models that consider the dynamics of the process [78, 88, 93], although they exhibit higher *mbe* values than some correlations, they also achieve better *rmse* values than the rest of the models, being even lower than 30% in the case of the Dirint model. In conclusion, authors recommended that the models Dirint and Ridley et al. for the data of this study.

Various models tested and their performances evaluated using solar data collected on the Mediterranean site of Ajaccio in Notton et al. study [71]. In fact, seven relations

applied, validated and compared using the mbe and rmse statistical test indicators to quantify their accuracy.

TABLE 6.7: Results of the Notton et al. hourly model's comparison study [71]

Models	mbe (%)	rmse (%)
Orgill and Hollands	5.386	37
Erbs et al.	-0.652	37.048
Hollands	0.441	37.414
Hollands and Crha	7.789	39.422
Miguel et al.	1.341	36.524
Skartveit and Olseth	9.949	36.898
Maxwell	-3.056	38.058

The results presented in Table 6.7, indicate that there is not a model largely better than another; the use of correlations taking into account the influence of solar elevation [66, 93] lead to a modest improvement of the determination of the hourly diffuse irradiation. Finally, the model showing the best performances for the Ajaccio solar data is the Miguel et al. model [19].

Dervishi and Mahdavi [20] compared eight models [25, 54, 59, 66, 77, 86, 92, 102] for estimating diffuse fraction of irradiance based on a database of measured irradiance from Vienna, Austria.

Two sets of measured data were used for this study. The first set, from January 2009 to May 2010, was used to compare the models' performance while the second set, from January 2007 to December 2008, was used to derive the local (Vienna) version of the three better performing models.

Towards the first comparison of these eight models, the relative error indicator was used:

$$re = \frac{100}{D_{ir}}(D_{ir} - D_{im}) \quad (6.38)$$

It revealed that three models [25, 77, 86] reproduce measurement results more accurately. About 62% of the results derived based on these three models display are of less than  $\pm 20\%$ .

TABLE 6.8: Results (%) with corresponding maximum relative error

Models	$\pm 5\%$	$\pm 10\%$	$\pm 15\%$	$\pm 20\%$	$\pm 25\%$	$\pm 30\%$	$\pm 35\%$	$\pm 40\%$
Erbs et al.	32.4	46.3	55.6	62.4	67.4	71.3	74.9	78.2
Reindl et al.	38.1	48.7	57.5	64.7	70.2	74.5	78.1	81.1
Orgill and Hollands	31.8	49.5	56	61.2	65.7	69.3	72.6	75.6
Lam and Li	19.4	31.9	42.8	52.7	60.8	67.3	72.3	75.9
Skatrveit et al.	18	23.6	27.9	31.6	35.3	38.9	42.6	45.9
Louche et al.	11	24.9	38	48.7	57.7	65.1	71	75.9
Maxwell	19.2	33.8	44.6	53.6	60.9	67.2	72.8	77.4
Vignola et al.	8.26	12.4	18.7	28.6	38.3	47.6	56.3	63.7

Using the second set of data, these three models were calibrated to explore the potential for the performance improvement. The model calibration (via derivation of new values for coefficients) resulted only in a modest improvement of the models' predictive performance.

## 6.6 Solar Radiation Model's Impact on BEP simulation

Investigating the impact of each solar model on the thermal simulation modeling, all the aforementioned models (described in Sections 6.3 and 6.4) have been developed in Matlab. For the sake of example, an exemplary building, consisting of three office rooms, is considered and described in this section. Using EnergyPlus a detailed thermal simulation model of the building is constructed. For our purpose and use of relevant weather data, the building is supposed to be located at several regions defined in Table B.2. Numerous test cases were conducted and defined in this section. In Appendix C their results are presented.

### 6.6.1 Building's Geometry and Zoning

A detailed representation of the building geometry, shown in Figure 6.10, created using the OpenStudio plugin for Google SketchUp.

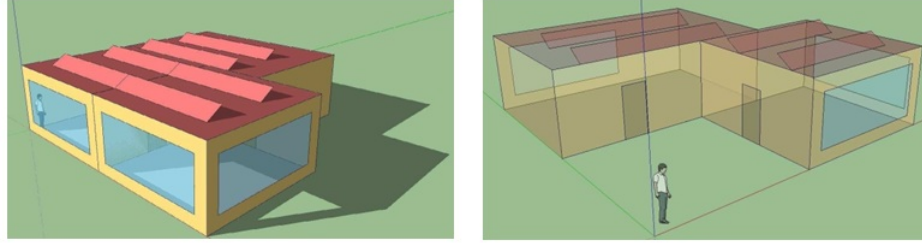


FIGURE 6.10: Geometry of the sample building created in OpenStudio plugin for Google SketchUp

The office building comprises 3 office rooms and each room of the building is defined as a separate thermal zone (see Figure 6.11 and Table 6.9).

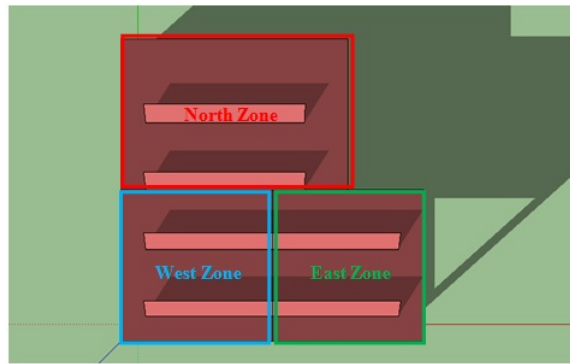


FIGURE 6.11: Zoning of the sample building

TABLE 6.9: Thermal Zones - Area and Height

	Area (m <sup>2</sup> )	Height (m)
West Zone	37	3
East Zone	37	3
North Zone	56	3
Whole Building	130	3

Since the glazing area principally affects the solar heat gains of the building, two scenarios of the glazing area are considered as follows:

- Glazing area scenario 1: each window's surface area equals to 10 m<sup>2</sup>,
- Glazing area scenario 2: each window's surface area equals to 5 m<sup>2</sup>.

For the Glazing area scenario 1, the surface of each opening in each side of the building is shown in Figure 6.12.

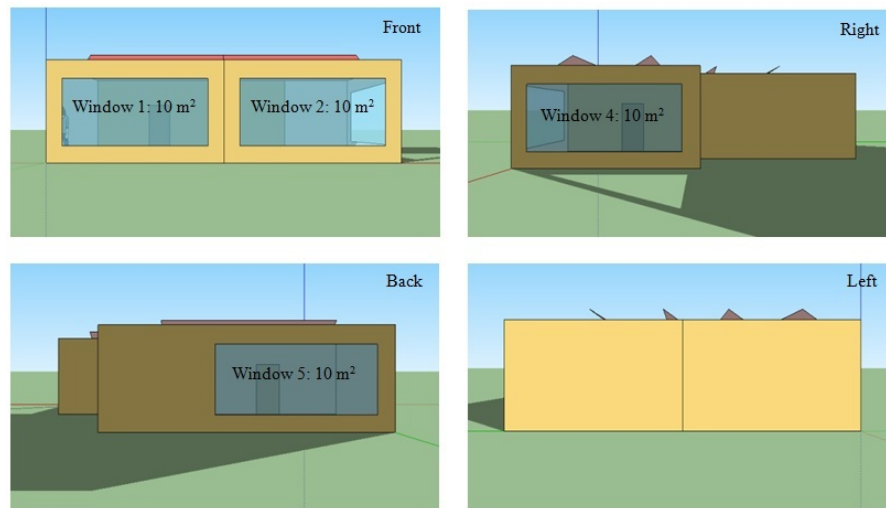


FIGURE 6.12: Glazing area in the front, right, back and left side of the building

### 6.6.2 Climate Data

The weather file used for the thermal simulation of the building contains data of a “typical year” for 10 different locations. The weather data are collected from 10 stations of the National Solar Radiation Data Base and their locations are shown in Figure 6.13. Please refer to Appendix B for further explanations on weather data collected.

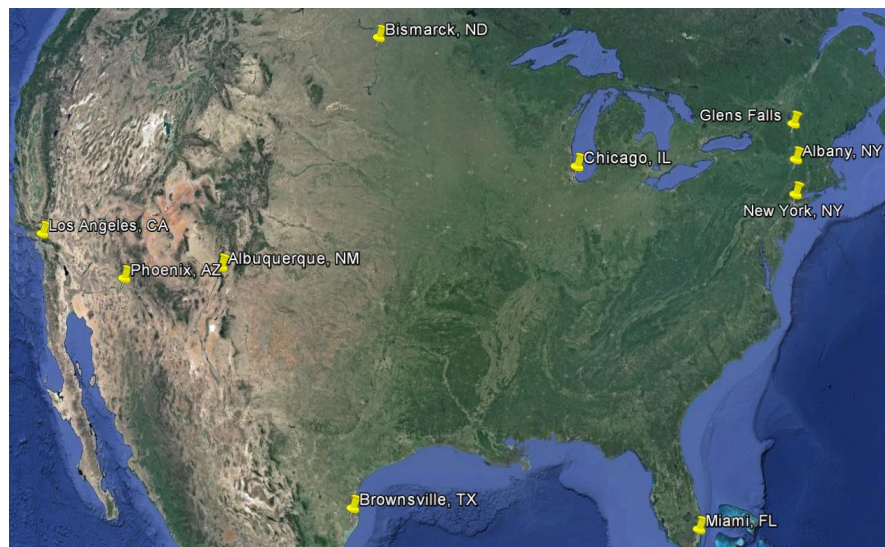


FIGURE 6.13: Locations of the weather station and the sample building at the University Campus

### 6.6.3 Shading

Blinds acting as shading devices, are located on the inside of the windows (“interior blinds”). When a window’s blinds are activated, they are assumed to cover all the glazed part of the window. The plane of the blinds is assumed to be parallel to the glazing. When the blinds are disabled, they are assumed to cover none of the window. The slat angle varies from  $0^\circ$ , when the front of the blinds is parallel to the glazing and faces toward the outdoors, to  $90^\circ$ , when the blinds are perpendicular to the glazing, to  $180^\circ$ , when the front of the blinds is parallel to the glazing and faces toward the indoors (see Figure 6.14).

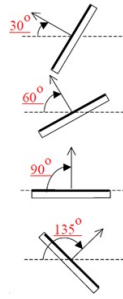


FIGURE 6.14: Blinds orientations for representative blind angles [22]

Three scenarios for the operation of the blinds (blind’s angle) are defined by the blind’s operation schedule as follows:

- Shading scenario 1: Blinds are always on – blind’s angle equals to  $0^\circ$ ;
- Shading scenario 2: Blinds are always off – blind’s angle equals to  $90^\circ$ ;
- Shading scenario 3: Blinds are on if beam plus diffuse solar radiation incident on the windows exceeds SetPoint ( $100 \text{ W/m}^2$ ).

## 6.7 Solar Radiation Models Comparison and Results

In our attempt to perform a comparison of the aforementioned solar radiation models, it was important to gather data of worldwide stations, covering different climate, geographical and meteorological conditions. Real efforts have been performed to provide irradiation values in various part of the world in a satisfactory way. In Appendix B the whole experiment set-up is presented, where initially the selected measured data source is defined and aiming at a homogeneous dataset, a data quality control is finally applied. Results of our investigation are summarized in Table 6.10.

TABLE 6.10: Statistical errors (%) of eight different solar radiation models for ten different locations

Location	Model	rmse	mbe	Location	Model	rmse	mbe
<b>Albany</b>	Bolland	24.48	3.83	<b>Albuquerque</b>	Bolland	32.18	5.36
	Dirint	20.31	1.53		Dirint	30.34	-1.66
	Erbs	24.82	4.63		Erbs	34.24	2.64
	Hawladar	27.65	10.87		Hawladar	36.28	3.34
	Jacovides	26.68	9.31		Jacovides	36.65	1.12
	Miguel	24.23	4.83		Miguel	35.68	-1.01
	Oliveira	28.29	12.09		Oliveira	37.66	7.38
	Orgill	24.01	4.19		Orgill	34.26	0.5
<b>New York</b>	Bolland	23.36	1.1	<b>Los Angeles</b>	Bolland	24.23	8.55
	Dirint	21.59	1.23		Dirint	22.26	7.29
	Erbs	24.79	2.94		Erbs	25.94	10.19
	Hawladar	23.58	6.24		Hawladar	25.07	11.26
	Jacovides	23.53	5.19		Jacovides	24.52	9.35
	Miguel	23.16	2.09		Miguel	24.03	8.56
	Oliveira	26.56	9.49		Oliveira	28.45	15.37
	Orgill	23.2	1.75		Orgill	24.13	8.62
<b>Chicago</b>	Bolland	21.74	1.67	<b>Miami</b>	Bolland	31.09	-1.29
	Dirint	20.91	1.17		Dirint	27.31	-1.12
	Erbs	22.29	2.42		Erbs	31.87	0.34
	Hawladar	23.34	8.26		Hawladar	29.44	5.79
	Jacovides	22.12	6.33		Jacovides	29.44	5.09
	Miguel	21	2.32		Miguel	30.22	0.33
	Oliveira	23.89	9.33		Oliveira	32.24	9.42
	Orgill	21.07	1.91		Orgill	30.44	-0.45
<b>Glens Falls</b>	Bolland	23.5	1.12	<b>Brownsville</b>	Bolland	30.07	-3.44
	Dirint	21.74	1.99		Dirint	27.93	-1.19
	Erbs	24.4	1.33		Erbs	29.7	-2.35
	Hawladar	25.77	8.12		Hawladar	28.05	3.28
	Jacovides	25.37	5.84		Jacovides	27.17	1.86
	Miguel	23.8	1.61		Miguel	28.41	-2.53
	Oliveira	27.06	7.38		Oliveira	27.99	5.76
	Orgill	23.9	1.86		Orgill	29.09	-2.98
<b>Phoenix</b>	Bolland	45.13	7.96	<b>Bismarck</b>	Bolland	31.98	1.35
	Dirint	40.38	-0.46		Dirint	30.59	0.8
	Erbs	41.44	2.43		Erbs	31.59	1.76
	Hawladar	41.13	0.76		Hawladar	32.49	5.21
	Jacovides	40.88	-0.38		Jacovides	32.24	3.31
	Miguel	41	-2.49		Miguel	31.15	0.6
	Oliveira	41.47	6.47		Oliveira	32.8	7.72
	Orgill	41.12	-0.76		Orgill	31.24	0.82

The results indicated that there is not a model largely better than another. Since statistical error differences are not high enough to highly recommend a model, the solar radiation model's selection is not expected to be of high impact. However, the model that considers the dynamics of the process (Dirint model), although it exhibits higher mbe values than some correlations in some cases, it always achieves better rmse values than the rest of the models and as such is recommended.

In Appendix [C](#), results of our investigation are presented thoroughly.

## Chapter 7

# Conclusions and Future Work

The focus of this thesis has been the development of a three-step methodology for (semi-) automated geometry's generation of thermal simulation models, including: a query on the building data model requesting geometry-related information of the data scheme by an open-source BIM repository; a processing of the acquired data by a 2<sup>nd</sup>-level boundary identification process called CBIP algorithm; and a transformation stage that converts the geometry and material properties information of IFC, along with the data obtained from the CBIP algorithm, to an EnergyPlus and/or TRNSYS input file. The presented methodology was applied on real buildings and the results demonstrated the ability in handling non-convex geometries and generating all the possible thermal, opening, shading and virtual elements. The 2<sup>nd</sup>-level space boundaries were identified and their space connectivity information was obtained accurately. The methodology facilitates significantly the overall process of energy simulation model creation from IFC geometric data. Of importance is the quality of the input IFC files; should geometric errors or other inaccuracies exist it might be hard to describe this process. It is for this reason that model-checking to ensure good quality of the IFC file is an important prerequisite.

Concerning building's geometry, the model derived applying the aforementioned methodology was of high detail and developed specifically for energy auditing purposes. For Control-Design tasks, efficient simulation was a prerequisite; the accuracy requirements of a proper model for such tasks are markedly different compared to the simulation-model for energy auditing purposes. Control-design requires a model that is able to capture the sensitivities and trends but no accuracy is necessary. As such, simulation speed-up approaches were proposed and their efficiency was investigated in a way that reduces complexity while maintaining features of the simulation. BEP simulation speed-up approaches were categorized to geometry and zoning reduction approaches. In

zoning reduction approaches, recreation of the thermal simulation model could be performed automatically, and as such an automatic process for generating speed-up models based on zoning reduction approaches was proposed. Regarding the geometry reduction approaches, two techniques aiming at reducing the computational cost of zonal-type building simulation models were presented: (i) geometry simplification for periodic geometries; and, (ii) the use of co-simulation to split a building into simpler sub-buildings, that can be evaluated in parallel and exchange boundary conditions data at each simulation timestep. These simulation speed-up approaches were evaluated, with respect to accuracy and computational effort, in two real test buildings. Regarding the computational effort, the efficiency of both techniques was supported by the results, which at the same time highlight the necessity of elaborate boundary conditions definition, since experiments showed that unrealistic boundary conditions can lead to substantial over- or under-estimation of zone air temperature. Geometry simplification for periodic geometries experiment showed that even though a significant simulation runtime speed-up was achieved, the inability to provide efficient boundary conditions, affected its efficiency. Co-simulation approach on the other hand, yielded an effective method, since its implementation led to a reduction of the simulation runtime up to 80% and at the same time managed to accurately define the boundary conditions between the sub-buildings.

For either energy auditing or Control-Design tasks, the availability of sensor data in a building were used to eliminate related uncertainties, further improving the corresponding thermal model. Establishing a link between the BEP simulation model's geometry and the building's sensed measurements, especially desirable also for the testing and design of control strategies, was adopted in this thesis, following a widely used co-simulation methodology, the Buildings Control Virtual Test Bed (BCVTB). Among a plethora of existing experiments on real building, one was selected to be presented, highlighting the necessity of the co-simulation setup within the model-assisted Control-Design context.

Finally, building's envelope thermal simulation calculations require weather data values in order to be executed. Most of these data are provided by weather files. However, it is quite common that weather data do not include information concerning solar radiation, but only a percentage of cloud coverage. Even if global radiation data are available, a crucial input in the simulation of building's energy performance is the availability of both diffuse and direct radiation data. Nevertheless, in most cases, measured data of diffuse and direct radiation are not available. Among a plethora of methods for estimating the diffuse (or direct) fraction of global solar radiation, the question of the best method has not been not fully settled, a task that this thesis tried to address. Results of our investigation indicated that there is not a method largely better than another. Since

statistical error differences are not high enough to highly recommend a method, the solar radiation model's selection was not of high impact. However, the model that considers the dynamics of the process (Dirint model), although it exhibited higher mbe values than some correlations in some cases, it always achieved better rmse values than the rest of the models and as such was recommended.

We conclude this section by giving some suggestions for future work:

- Development of a methodology to automatic translate IFC to BEP simulation's HVAC data: Concerning the HVAC data embedded to the BIM and their translation to a BEP simulation model's input data, IFC is the most widely used BIM schema, though suffering from limitations in the description of HVAC systems [89]. Commonly, HVAC modeling in BEP simulation engines requires further information than what is included in an IFC file. However, simulation engines, like EnergyPlus, offer to autosize this additional information. Currently, only the IFC HVAC Interface to EnergyPlus effectively exchanges HVAC information between IFC and EnergyPlus simulation [10].
- Development or use of an intermediate data model: Since the translation process to each simulation is specific to the dictionaries used, it might be conceivable that an intermediate data model (e.g. SimModel [74]) is populated and then one generator for each specific simulation engine is developed. For this case, most of the transformation rules described in this work would be applicable requiring only minor modifications.
- Co-Simulation through FMI: Lately, FMI utilization for co-simulation increasingly gains ground due to the fact that BCVTB requires expert knowledge, increasing the learning curve for a co-simulation setup. For the sake of example, a recent study shows impressively the coupling of EnergyPlus (building scale) and CitySim (urban scale) [98], where the two simulation engines, using the Functional Mock-up Interface (FMI) co-simulation framework, exchange data to improve the accuracy of both simulations. Therefore, following the future developments, the co-simulation setup described in this thesis will be replaced by a setup utilizing the FMI framework.
- Simulation speed-up based on order reduction approaches: Concerning the Control Design process, a number of approaches have been proposed in the literature; one such approach that has attracted a lot of interest is the Model Predictive Control paradigm (MPC), in which an order reduced (=simplified) model of the building is used, with specific mathematical properties (e.g. linearity or quasi-linearity). A criticism about MPC approaches is that the simplified models used are not

accurate enough and combined with prediction uncertainties optimality does not yield significant performance improvements. A suitable MPC building model can be developed utilizing the BRCM [95], which can be effectively improved through a data-exchange set-up with a more comprehensive EnergyPlus model [13].

## Appendix A

# CBIP Aglorithm Results in XML-format Files

The first stage of the three-stage process of building thermal simulation model creation involves the extraction of IFC's geometry as well as thermal material properties data, which is accomplished over the world wide web by the TNO BIM Server [\[11\]](#).

The extracted data are processed further by the CBIP algorithm, which outputs two xml files: (1) SB.xml file, which contains the 2<sup>nd</sup>-level space boundary information and (2) the Materials.xml file.

The xsd schema of Materials.xml and SB.xml files are displayed in Figure [A.1](#) and [A.2](#), respectively.

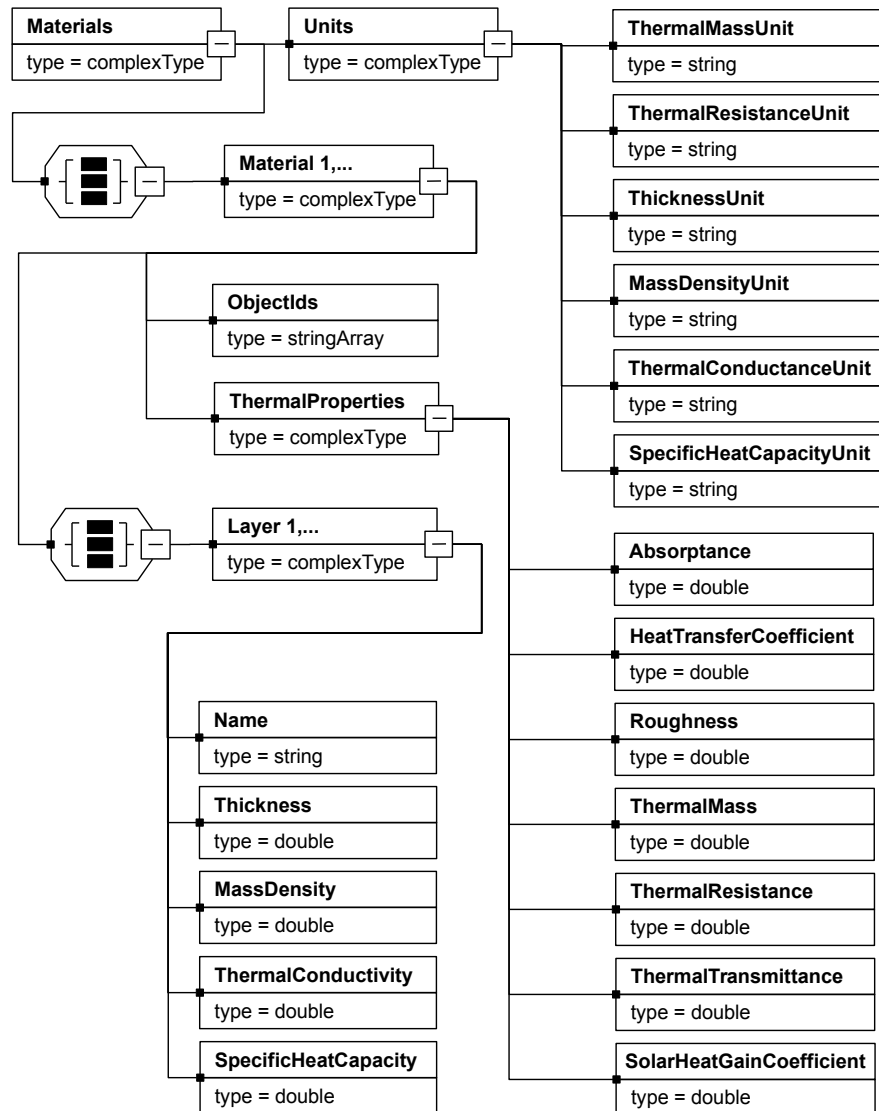


FIGURE A.1: XSD schema of Materials.xml file

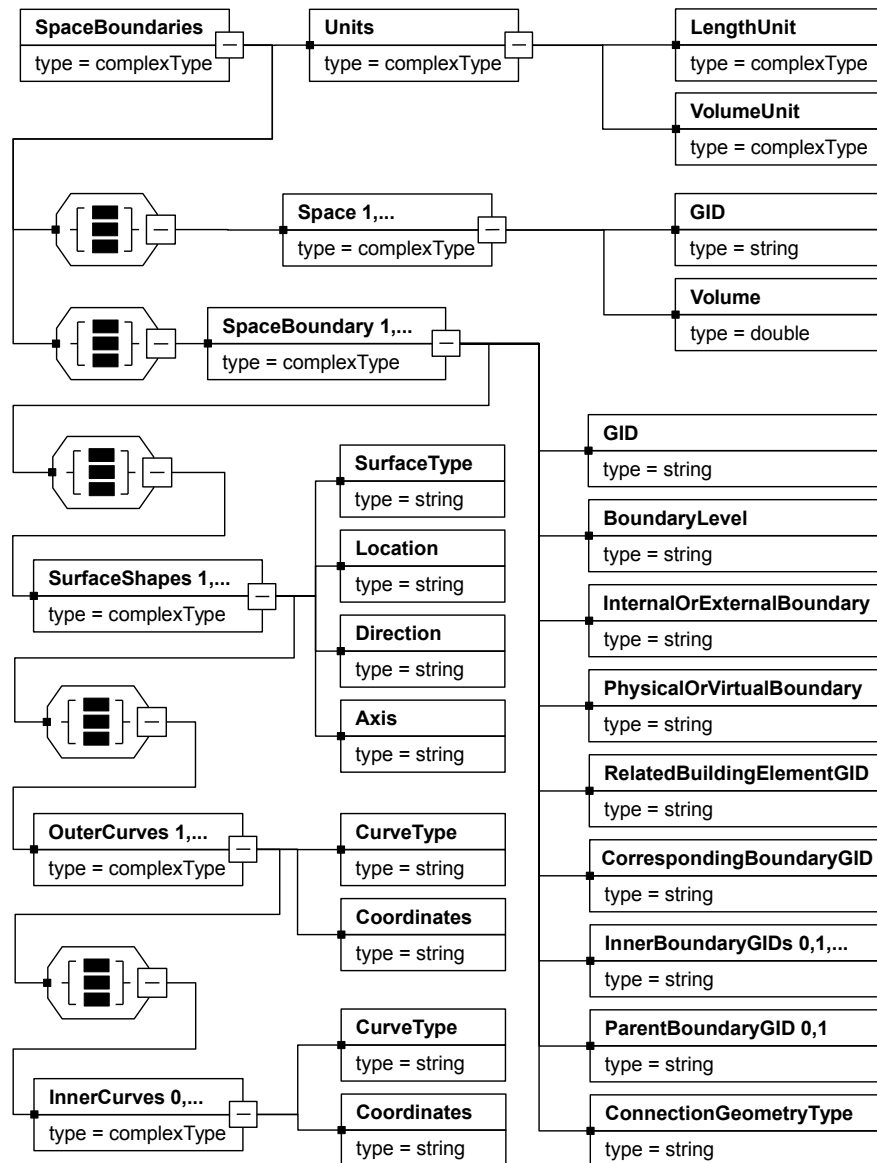


FIGURE A.2: XSD schema of SB.xml file

## Appendix B

# Solar Radiation Data

In order to perform a comparison of the aforementioned solar radiation models, it is important to gather data of worldwide stations, covering different climate, geographical and meteorological conditions. Real efforts have been performed to provide irradiation values in various part of the world in a satisfactory way. In this chapter, initially the selected measured data source is defined. In order to have a homogeneous dataset, a data quality control is finally applied.

### B.1 Databases Providing Solar Radiation Data

For this research, measured global and diffuse irradiance from locally recorded weather data, were not available. Hence, a survey on the main possibilities offered for retrieving solar radiation data held and its results are summarized in Table B.1.

TABLE B.1: Summary of the meteorological databases providing hourly solar radiation data

Database	Region	Source	Period	Variables					Availability
				I	Ib	Id	Ta	H	
<b>Meteonorm</b>	Worldwide	1700 Stations Interpolations	1995-2005	•	•	•	•	•	Software
<b>ESRA</b>	Europe		1981-1990	•	•	•	•	•	Software
<b>Satellite</b>	Europe	Meteosat	1996-2000	•	•	•			Web free
<b>US TMY3</b>	USA	1020 stations TMY	1991-2005	•	•	•	•	•	Web free
<b>SoDa (HelioClim-3)</b>	Europe, Africa	Meteosat	From 2004	•	•	•			Web restricted 2005 free
<b>WRDC</b>	Worldwide	1195 stations	1964-1993	•	•	•			Web free
<b>SolarGIS</b>	Worldwide	Meteosat	From 1994	•	•	•	•	•	Web, paid
<b>IWEC</b>	Worldwide	227 locations	1960-1990	•	•	•	•	•	Web free

Meteonorm (2000) is a digital solar atlas presented on CD-ROM. It contains a database of ground stations measurements made by a combination of several databases from different parts of the world (Swiss meteorological institute, GEBA, WMO). The main period of the measurement is from 1961 to 1990. Other climatological data useful for renewable energy engineering are also available (temperature, humidity, wind speed, precipitation). Meteonorm extrapolates hourly data from statistical data for a location. Where statistical data aren't available, Meteonorm interpolates from other nearby sites.

European Solar Radiation Atlas is also presented on CD-ROM. The CD-ROM contains the software to exploit the database. The input data are based on the period 1981-1990. This software uses either a "map" or a "station" mode at user choice. In the first case, any geographical site can be designated. In the second mode, only the available measuring stations can be selected. The data provided by these stations are actual hourly-sensed values of global horizontal radiation and its components, air temperature, humidity etc.

Satellite project offers a database of solar radiation data derived from satellite images, which can be accessed through the world wide web and produces value-added information mostly for daylighting purposes. Hence, Satellite provides hourly solar radiation data, estimated using satellite images, over 5 years (1996-2000), for any pixel of  $5 \times 7 \text{ km}^2$  in Europe.

The TMY3s are data sets of hourly values of solar radiation and meteorological elements for a 1-year period, for 1020 locations in the USA. TMY3s data sets are derived from the National Solar Radiation Data Base, a serially complete collection of hourly values of the three most common measurements of solar radiation (global horizontal, direct normal, and diffuse horizontal) and other climatological data (temperature, humidity, wind speed, precipitation).

Helioclim-3 provides global solar radiation data in hourly values, estimated using satellite images captured by Meteosat, since February 2004. These data are not free but data for 2005 are available for tests. Moreover, diffuse solar radiation data are not measured but estimated values using the equation of (Ruiz-Arias et al., 2010).

The World Radiation Data Center (WRDC) offers an on-line access to archived data for many countries. Measurements are most often made on a daily basis and not hourly basis, except for a limited number of countries. Users should find models that synthesize hourly irradiation from a daily irradiation. These models call upon statistical knowledge on the hourly profile that is not available at most stations and should be inferred from other stations performing hourly measurements.

SolarGIS provides access to solar and meteorological data covering the last 17 years and updated in real time. Hourly time series are available for any location in Europe, Africa, Asia, and parts of Australia and South America. These data are not free.

Since the principal aim of this chapter is to present a comparative study of existing solar radiation models where only the hourly global horizontal radiation is measured, hourly sensed values of global horizontal radiation and one of its components (either diffuse horizontal radiation or direct normal radiation) are required. With the global horizontal radiation and one of its components, the other component is obtained using the following equation:

$$I = I_d + I_b \sin \gamma \quad (\text{B.1})$$

Furthermore, the availability of measured dew point temperature data, which could be obtained from measured data of dry bulb temperature and relative humidity, is required to evaluate the Dirint model.

With these prerequisites, Meteonorm and IWECC are eliminated for use due to their lack of measured solar radiation data while, European Solar Radiation atlas and SolarGIS are rejected because of the very high acquisition cost of sensed solar radiation measurements. Due to the fact that Helioclim-3 and Satellicht provide solar data estimated by satellite images and not measured values, they are also eliminated. Although the World Radiation Data Center (WRDC) offers an on-line access to archived data for many countries, its measurements are most often made on a daily basis and not hourly basis and as such, not suitable for use in the present work. Accordingly, the TMY3s are the only source for free hourly measured, in regions of the United States of America though, solar radiation, air temperature and humidity data.

Data sets from 10 different stations, retrieved from the National Solar Radiation Data Base (TMY3) were used for this study. According to [78], the stations were selected properly to cover a wide range of climatic environments and Table B.2 briefly describes the climatic environment of each data set.

TABLE B.2: Climatic environment of each selected data set [78]

<b>Site</b>	<b>Climate</b>
<b>Albany, NY</b>	Humid continental
<b>New York, NY</b>	Humid continental Maritime influence
<b>Chicago, IL</b>	Humid continental Great Lakes basin
<b>Glens Falls, NY</b>	Humid continental Adirondack Mountains
<b>Phoenix, AZ</b>	Arid Low elevation
<b>Albuquerque, NM</b>	Arid High elevation
<b>Los Angeles, CA</b>	Arid Maritime influence Smog
<b>Miami, FL</b>	Tropical Low latitude
<b>Brownsville, TX</b>	Subtropical Low latitude High cloudiness
<b>Bismarck, ND</b>	Dry continental Extensive winter-time snow cover

## B.2 Data Quality Control

Data for night periods are excluded and the first constraint to the dataset is:

$$I \geq 0. \quad (\text{B.2})$$

For low values of solar altitude angle, instruments for the measurement of solar radiation have some limitation due to the cosine response and as such the following threshold for the solar altitude angle is considered:

$$\gamma \geq 5^\circ \quad (\text{B.3})$$

Moreover, the collected hourly irradiation values are checked against quality controls listed by [41] and proposed by European Commission—Daylight I, 1993:

$$I \geq 5\text{W/m}^2; \quad (\text{B.4})$$

$$\frac{I_d}{I} \leq 1.1; \quad (\text{B.5})$$

$$\frac{I}{I_o} \leq 1.2; \quad (\text{B.6})$$

$$\frac{I}{I_o} \leq 1.2; \quad (\text{B.7})$$

$$I_b \geq I_o; \quad (\text{B.8})$$

where,  $I$ ,  $I_b$ ,  $I_d$  and  $I_o$  are the global horizontal radiation, the direct normal radiation, the diffuse horizontal radiation and the horizontal extraterrestrial radiation, respectively.

Finally, two additional constraints are considered by [86] to identify particular cases of overcast and clear skies as follows:

$$\frac{I_d}{I} \geq 0.9 \quad \text{for } k_t < 0.20; \quad (\text{B.9})$$

$$\frac{I_d}{I} \leq 0.8 \quad \text{for } k_t > 0.60. \quad (\text{B.10})$$

## Appendix C

### Solar radiation models' Results

TABLE C.1: Statistical errors of ten variables for two solar radiation models – Albany

Location: Albany	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	20.45	-1.94	case_Glazing2_Blinds1	20.45	-1.96
DirintvsReal_HeatGain_East		21.62	-0.64		21.58	-0.65
DirintvsReal_HeatGain_North		19.49	1.78		19.26	1.73
DirintvsReal_RadiantTemp_West		1.56	-0.07		1.40	0.01
DirintvsReal_RadiantTemp_East		1.81	-0.23		1.44	-0.20
DirintvsReal_RadiantTemp_North		0.92	0.45		0.92	0.47
DirintvsReal_AirTemp_West		1.86	-0.11		1.54	-0.03
DirintvsReal_AirTemp_East		2.67	-0.29		1.88	-0.26
DirintvsReal_AirTemp_North		0.95	0.37		0.96	0.40
DirintvsReal_PV		7.89	-1.13		7.89	-1.13
OliveiravsReal_HeatGain_West		22.00	-4.03		22.04	-4.07
OliveiravsReal_HeatGain_East		21.43	-2.32		21.41	-2.39
OliveiravsReal_HeatGain_North		21.80	5.65		21.53	5.56
OliveiravsReal_RadiantTemp_West		1.94	-0.72		1.71	-0.60
OliveiravsReal_RadiantTemp_East		2.14	-0.79		1.77	-0.69
OliveiravsReal_RadiantTemp_North		0.93	0.39		0.91	0.40
OliveiravsReal_AirTemp_West		2.22	-0.78		1.85	-0.65
OliveiravsReal_AirTemp_East		2.93	-0.87		2.17	-0.77
OliveiravsReal_AirTemp_North		0.97	0.29		0.93	0.30
OliveiravsReal_PV		8.70	-2.28		8.70	-2.28
OliveiravsDirint_HeatGain_West		8.58	-2.05		8.59	-2.07
OliveiravsDirint_HeatGain_East		7.53	-1.67		7.35	-1.73
OliveiravsDirint_HeatGain_North		9.78	3.94		9.65	3.90
OliveiravsDirint_RadiantTemp_West		0.94	-0.65		0.83	-0.61
OliveiravsDirint_RadiantTemp_East		0.91	-0.55		0.76	-0.48
OliveiravsDirint_RadiantTemp_North		0.26	-0.05		0.25	-0.07
OliveiravsDirint_AirTemp_West		1.12	-0.67		0.96	-0.62
OliveiravsDirint_AirTemp_East		1.65	-0.58		1.21	-0.51
OliveiravsDirint_AirTemp_North		0.47	-0.08		0.46	-0.10
OliveiravsDirint_PV		3.44	-1.14		3.44	-1.14
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	20.52	-2.09	case_Glazing2_Blinds2	20.09	-2.11
DirintvsReal_HeatGain_East		21.74	-0.72		20.97	-0.78
DirintvsReal_HeatGain_North		22.84	2.36		22.20	2.30
DirintvsReal_RadiantTemp_West		1.96	-0.37		1.63	-0.18
DirintvsReal_RadiantTemp_East		2.18	-0.48		1.74	-0.34
DirintvsReal_RadiantTemp_North		0.97	0.42		0.94	0.45
DirintvsReal_AirTemp_West		1.97	-0.35		1.65	-0.16
DirintvsReal_AirTemp_East		2.60	-0.53		1.92	-0.39
DirintvsReal_AirTemp_North		0.94	0.29		0.94	0.35
DirintvsReal_PV		7.89	-1.13		7.89	-1.13
OliveiravsReal_HeatGain_West		22.22	-4.27		21.83	-4.28
OliveiravsReal_HeatGain_East		21.67	-2.51		20.97	-2.57
OliveiravsReal_HeatGain_North		25.31	6.67		24.64	6.54
OliveiravsReal_RadiantTemp_West		2.62	-1.23		2.14	-0.92
OliveiravsReal_RadiantTemp_East		2.72	-1.24		2.20	-0.99
OliveiravsReal_RadiantTemp_North		1.12	0.47		1.01	0.43
OliveiravsReal_AirTemp_West		2.60	-1.21		2.11	-0.89
OliveiravsReal_AirTemp_East		3.10	-1.28		2.35	-1.02
OliveiravsReal_AirTemp_North		1.03	0.26		0.99	0.28
OliveiravsReal_PV		8.70	-2.28		8.70	-2.28
OliveiravsDirint_HeatGain_West		8.63	-2.14		8.50	-2.12
OliveiravsDirint_HeatGain_East		7.42	-1.78		7.21	-1.78

OliveiravsDirint_HeatGain_North		11.25	4.42		10.96	4.34
OliveiravsDirint_RadiantTemp_West		1.27	-0.85		1.05	-0.74
OliveiravsDirint_RadiantTemp_East		1.16	-0.75		0.96	-0.65
OliveiravsDirint_RadiantTemp_North		0.33	0.06		0.27	-0.01
OliveiravsDirint_AirTemp_West		1.34	-0.86		1.10	-0.73
OliveiravsDirint_AirTemp_East		1.83	-0.75		1.31	-0.63
OliveiravsDirint_AirTemp_North		0.46	-0.03		0.47	-0.07
OliveiravsDirint_PV		3.44	-1.14		3.44	-1.14
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	20.37	-1.92	case_Glazing2_Blinds3	20.39	-1.95
DirintvsReal_HeatGain_East		21.49	-0.61		21.49	-0.63
DirintvsReal_HeatGain_North		19.40	1.77		19.20	1.73
DirintvsReal_RadiantTemp_West		1.56	-0.06		1.40	0.02
DirintvsReal_RadiantTemp_East		1.83	-0.22		1.44	-0.19
DirintvsReal_RadiantTemp_North		0.93	0.46		0.93	0.48
DirintvsReal_AirTemp_West		1.86	-0.11		1.56	-0.02
DirintvsReal_AirTemp_East		2.56	-0.30		1.87	-0.26
DirintvsReal_AirTemp_North		0.97	0.37		0.95	0.40
DirintvsReal_PV		7.89	-1.13		7.89	-1.13
OliveiravsReal_HeatGain_West		21.92	-4.01		21.97	-4.05
OliveiravsReal_HeatGain_East		21.31	-2.29		21.30	-2.36
OliveiravsReal_HeatGain_North		21.65	5.59		21.44	5.52
OliveiravsReal_RadiantTemp_West		1.94	-0.70		1.71	-0.59
OliveiravsReal_RadiantTemp_East		2.16	-0.76		1.77	-0.68
OliveiravsReal_RadiantTemp_North		0.95	0.41		0.92	0.41
OliveiravsReal_AirTemp_West		2.23	-0.77		1.86	-0.64
OliveiravsReal_AirTemp_East		2.86	-0.88		2.16	-0.77
OliveiravsReal_AirTemp_North		0.97	0.30		0.95	0.30
OliveiravsReal_PV		8.70	-2.28		8.70	-2.28
OliveiravsDirint_HeatGain_West		8.57	-2.05		8.59	-2.07
OliveiravsDirint_HeatGain_East		7.53	-1.67		7.33	-1.72
OliveiravsDirint_HeatGain_North		9.72	3.89		9.63	3.86
OliveiravsDirint_RadiantTemp_West		0.93	-0.63		0.83	-0.61
OliveiravsDirint_RadiantTemp_East		0.92	-0.54		0.76	-0.49
OliveiravsDirint_RadiantTemp_North		0.27	-0.05		0.25	-0.07
OliveiravsDirint_AirTemp_West		1.13	-0.66		0.98	-0.63
OliveiravsDirint_AirTemp_East		1.59	-0.58		1.21	-0.52
OliveiravsDirint_AirTemp_North		0.47	-0.07		0.46	-0.10
OliveiravsDirint_PV		3.44	-1.14		3.44	-1.14

TABLE C.2: Statistical errors of ten variables for two solar radiation models – Albuquerque

Location: Albuquerque	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	13.98	-0.02	case_Glazing2_Blinds1	13.99	0.04
DirintvsReal_HeatGain_East		18.34	2.04		18.42	2.11
DirintvsReal_HeatGain_North		23.69	-2.97		23.58	-2.97
DirintvsReal_RadiantTemp_West		1.24	-0.17		1.21	-0.22
DirintvsReal_RadiantTemp_East		1.33	0.53		1.04	0.45
DirintvsReal_RadiantTemp_North		0.71	-0.10		0.70	-0.09
DirintvsReal_AirTemp_West		1.35	-0.10		1.25	-0.17
DirintvsReal_AirTemp_East		2.09	0.48		1.34	0.41
DirintvsReal_AirTemp_North		0.76	-0.12		0.74	-0.11
DirintvsReal_PV		6.08	-0.28		6.08	-0.28
OliveiravsReal_HeatGain_West		13.78	-0.70		13.80	-0.65
OliveiravsReal_HeatGain_East		16.29	1.10		16.33	1.15
OliveiravsReal_HeatGain_North		24.66	-0.39		24.54	-0.38
OliveiravsReal_RadiantTemp_West		1.24	-0.40		1.21	-0.43
OliveiravsReal_RadiantTemp_East		1.18	0.28		0.92	0.23
OliveiravsReal_RadiantTemp_North		0.75	-0.13		0.73	-0.14
OliveiravsReal_AirTemp_West		1.36	-0.35		1.26	-0.40
OliveiravsReal_AirTemp_East		1.92	0.23		1.23	0.19
OliveiravsReal_AirTemp_North		0.79	-0.15		0.77	-0.16
OliveiravsReal_PV		6.15	-0.68		6.15	-0.68
OliveiravsDirint_HeatGain_West		4.57	-0.68		4.57	-0.69
OliveiravsDirint_HeatGain_East		5.27	-0.95		5.02	-0.98
OliveiravsDirint_HeatGain_North		7.47	2.50		7.44	2.51
OliveiravsDirint_RadiantTemp_West		0.42	-0.23		0.38	-0.22
OliveiravsDirint_RadiantTemp_East		0.46	-0.25		0.36	-0.23
OliveiravsDirint_RadiantTemp_North		0.15	-0.03		0.15	-0.04
OliveiravsDirint_AirTemp_West		0.54	-0.24		0.46	-0.23
OliveiravsDirint_AirTemp_East		1.07	-0.25		0.68	-0.22
OliveiravsDirint_AirTemp_North		0.27	-0.03		0.25	-0.05
OliveiravsDirint_PV		1.85	-0.40		1.85	-0.40
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	14.09	-0.13	case_Glazing2_Blinds2	13.89	-0.10
DirintvsReal_HeatGain_East		18.38	2.10		17.87	2.09
DirintvsReal_HeatGain_North		29.27	-3.26		28.63	-3.23
DirintvsReal_RadiantTemp_West		1.35	-0.12		1.27	-0.14
DirintvsReal_RadiantTemp_East		1.52	0.58		1.28	0.58
DirintvsReal_RadiantTemp_North		0.85	-0.18		0.76	-0.12
DirintvsReal_AirTemp_West		1.34	-0.07		1.26	-0.09
DirintvsReal_AirTemp_East		2.15	0.42		1.44	0.48
DirintvsReal_AirTemp_North		0.82	-0.18		0.77	-0.13
DirintvsReal_PV		6.08	-0.28		6.08	-0.28
OliveiravsReal_HeatGain_West		14.00	-0.84		13.82	-0.80
OliveiravsReal_HeatGain_East		16.37	1.12		15.92	1.13
OliveiravsReal_HeatGain_North		30.22	-0.24		29.57	-0.23
OliveiravsReal_RadiantTemp_West		1.38	-0.39		1.28	-0.40
OliveiravsReal_RadiantTemp_East		1.36	0.28		1.12	0.29
OliveiravsReal_RadiantTemp_North		0.91	-0.14		0.81	-0.13
OliveiravsReal_AirTemp_West		1.40	-0.34		1.28	-0.37
OliveiravsReal_AirTemp_East		2.06	0.17		1.30	0.22
OliveiravsReal_AirTemp_North		0.86	-0.17		0.80	-0.16
OliveiravsReal_PV		6.15	-0.68		6.15	-0.68
OliveiravsDirint_HeatGain_West		4.70	-0.71		4.61	-0.69
OliveiravsDirint_HeatGain_East		5.10	-1.00		4.94	-0.98

OliveiravsDirint_HeatGain_North		9.03	2.92		8.84	2.90
OliveiravsDirint_RadiantTemp_West		0.52	-0.27		0.46	-0.26
OliveiravsDirint_RadiantTemp_East		0.54	-0.30		0.45	-0.29
OliveiravsDirint_RadiantTemp_North		0.18	0.04		0.15	-0.01
OliveiravsDirint_AirTemp_West		0.64	-0.28		0.53	-0.27
OliveiravsDirint_AirTemp_East		1.51	-0.25		0.81	-0.26
OliveiravsDirint_AirTemp_North		0.26	0.01		0.23	-0.03
OliveiravsDirint_PV		1.85	-0.40		1.85	-0.40
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	13.97	-0.02	case_Glazing2_Blinds3	13.98	0.03
DirintvsReal_HeatGain_East		18.27	2.04		18.40	2.11
DirintvsReal_HeatGain_North		23.61	-2.94		23.55	-2.94
DirintvsReal_RadiantTemp_West		1.24	-0.17		1.21	-0.21
DirintvsReal_RadiantTemp_East		1.36	0.53		1.05	0.45
DirintvsReal_RadiantTemp_North		0.72	-0.10		0.70	-0.09
DirintvsReal_AirTemp_West		1.37	-0.11		1.26	-0.16
DirintvsReal_AirTemp_East		2.06	0.48		1.36	0.41
DirintvsReal_AirTemp_North		0.77	-0.13		0.74	-0.11
DirintvsReal_PV		6.08	-0.28		6.08	-0.28
OliveiravsReal_HeatGain_West		13.76	-0.70		13.79	-0.66
OliveiravsReal_HeatGain_East		16.24	1.11		16.32	1.16
OliveiravsReal_HeatGain_North		24.57	-0.39		24.50	-0.38
OliveiravsReal_RadiantTemp_West		1.24	-0.40		1.21	-0.43
OliveiravsReal_RadiantTemp_East		1.20	0.28		0.93	0.23
OliveiravsReal_RadiantTemp_North		0.76	-0.13		0.74	-0.13
OliveiravsReal_AirTemp_West		1.37	-0.36		1.27	-0.40
OliveiravsReal_AirTemp_East		1.93	0.24		1.23	0.18
OliveiravsReal_AirTemp_North		0.81	-0.16		0.77	-0.16
OliveiravsReal_PV		6.15	-0.68		6.15	-0.68
OliveiravsDirint_HeatGain_West		4.57	-0.67		4.57	-0.69
OliveiravsDirint_HeatGain_East		5.24	-0.95		5.02	-0.97
OliveiravsDirint_HeatGain_North		7.49	2.48		7.46	2.49
OliveiravsDirint_RadiantTemp_West		0.42	-0.23		0.38	-0.22
OliveiravsDirint_RadiantTemp_East		0.47	-0.25		0.37	-0.23
OliveiravsDirint_RadiantTemp_North		0.15	-0.03		0.15	-0.04
OliveiravsDirint_AirTemp_West		0.56	-0.25		0.48	-0.24
OliveiravsDirint_AirTemp_East		1.10	-0.25		0.67	-0.24
OliveiravsDirint_AirTemp_North		0.27	-0.03		0.22	-0.05
OliveiravsDirint_PV		1.85	-0.40		1.85	-0.40

TABLE C.3: Statistical errors of ten variables for two solar radiation models – Bismarck

Location: Bismarck	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	50.92	-1.44	case_Glazing2_Blinds1	50.88	-1.42
DirintvsReal_HeatGain_East		39.89	0.20		40.13	0.23
DirintvsReal_HeatGain_North		39.51	-2.42		39.12	-2.38
DirintvsReal_RadiantTemp_West		5.11	-1.23		4.93	-1.25
DirintvsReal_RadiantTemp_East		4.06	0.11		3.30	0.12
DirintvsReal_RadiantTemp_North		3.04	-0.81		3.07	-0.81
DirintvsReal_AirTemp_West		5.86	-1.19		5.23	-1.21
DirintvsReal_AirTemp_East		5.67	0.09		4.10	0.08
DirintvsReal_AirTemp_North		3.35	-0.82		3.36	-0.83
DirintvsReal_PV		22.75	-0.38		22.75	-0.38
OliveiravsReal_HeatGain_West		50.87	-3.28		50.86	-3.27
OliveiravsReal_HeatGain_East		38.99	-0.97		39.25	-0.98
OliveiravsReal_HeatGain_North		39.94	0.46		39.55	0.47
OliveiravsReal_RadiantTemp_West		5.35	-1.98		5.14	-1.95
OliveiravsReal_RadiantTemp_East		4.05	-0.42		3.29	-0.36
OliveiravsReal_RadiantTemp_North		3.01	-0.86		3.04	-0.88
OliveiravsReal_AirTemp_West		6.08	-1.95		5.43	-1.93
OliveiravsReal_AirTemp_East		5.61	-0.49		4.09	-0.43
OliveiravsReal_AirTemp_North		3.32	-0.90		3.33	-0.93
OliveiravsReal_PV		22.65	-1.33		22.65	-1.33
OliveiravsDirint_HeatGain_West		7.62	-1.81		7.63	-1.83
OliveiravsDirint_HeatGain_East		5.90	-1.17		5.77	-1.21
OliveiravsDirint_HeatGain_North		8.14	2.81		8.03	2.79
OliveiravsDirint_RadiantTemp_West		1.03	-0.74		0.92	-0.70
OliveiravsDirint_RadiantTemp_East		0.95	-0.53		0.80	-0.48
OliveiravsDirint_RadiantTemp_North		0.30	-0.05		0.29	-0.07
OliveiravsDirint_AirTemp_West		1.25	-0.76		1.05	-0.71
OliveiravsDirint_AirTemp_East		1.64	-0.59		1.35	-0.51
OliveiravsDirint_AirTemp_North		0.61	-0.08		0.55	-0.10
OliveiravsDirint_PV		3.15	-0.95		3.15	-0.95
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	48.65	-1.18	case_Glazing2_Blinds2	47.42	-1.08
DirintvsReal_HeatGain_East		40.76	0.58		39.29	0.74
DirintvsReal_HeatGain_North		45.02	-2.56		43.62	-2.44
DirintvsReal_RadiantTemp_West		4.89	-0.96		4.68	-1.06
DirintvsReal_RadiantTemp_East		4.31	0.62		3.62	0.44
DirintvsReal_RadiantTemp_North		2.86	-0.74		2.93	-0.76
DirintvsReal_AirTemp_West		5.13	-0.97		4.81	-1.08
DirintvsReal_AirTemp_East		4.60	0.50		3.71	0.35
DirintvsReal_AirTemp_North		3.12	-0.75		3.20	-0.77
DirintvsReal_PV		22.75	-0.38		22.75	-0.38
OliveiravsReal_HeatGain_West		48.56	-3.06		47.35	-2.94
OliveiravsReal_HeatGain_East		39.75	-0.67		38.30	-0.49
OliveiravsReal_HeatGain_North		45.27	0.86		43.87	0.91
OliveiravsReal_RadiantTemp_West		5.21	-1.88		4.95	-1.89
OliveiravsReal_RadiantTemp_East		4.15	-0.06		3.54	-0.17
OliveiravsReal_RadiantTemp_North		2.82	-0.70		2.89	-0.78
OliveiravsReal_AirTemp_West		5.45	-1.90		5.09	-1.89
OliveiravsReal_AirTemp_East		4.47	-0.22		3.66	-0.27
OliveiravsReal_AirTemp_North		3.09	-0.79		3.16	-0.85
OliveiravsReal_PV		22.65	-1.33		22.65	-1.33
OliveiravsDirint_HeatGain_West		7.60	-1.86		7.49	-1.84
OliveiravsDirint_HeatGain_East		5.92	-1.25		5.75	-1.24

OliveiravsDirint_HeatGain_North		9.66	3.33		9.40	3.27
OliveiravsDirint_RadiantTemp_West		1.31	-0.92		1.11	-0.82
OliveiravsDirint_RadiantTemp_East		1.15	-0.69		0.98	-0.61
OliveiravsDirint_RadiantTemp_North		0.36	0.04		0.30	-0.02
OliveiravsDirint_AirTemp_West		1.40	-0.92		1.16	-0.80
OliveiravsDirint_AirTemp_East		1.76	-0.73		1.37	-0.62
OliveiravsDirint_AirTemp_North		0.60	-0.04		0.58	-0.08
OliveiravsDirint_PV		3.15	-0.95		3.15	-0.95
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	50.75	-1.44	case_Glazing2_Blinds3	50.74	-1.41
DirintvsReal_HeatGain_East		39.69	0.23		39.97	0.23
DirintvsReal_HeatGain_North		39.05	-2.42		38.73	-2.38
DirintvsReal_RadiantTemp_West		5.10	-1.21		4.93	-1.24
DirintvsReal_RadiantTemp_East		4.13	0.15		3.32	0.14
DirintvsReal_RadiantTemp_North		3.04	-0.79		3.07	-0.80
DirintvsReal_AirTemp_West		5.91	-1.21		5.26	-1.23
DirintvsReal_AirTemp_East		5.65	0.03		4.08	0.05
DirintvsReal_AirTemp_North		3.39	-0.84		3.38	-0.84
DirintvsReal_PV		22.75	-0.38		22.75	-0.38
OliveiravsReal_HeatGain_West		50.70	-3.27		50.71	-3.26
OliveiravsReal_HeatGain_East		38.80	-0.92		39.08	-0.96
OliveiravsReal_HeatGain_North		39.49	0.41		39.16	0.43
OliveiravsReal_RadiantTemp_West		5.33	-1.95		5.13	-1.95
OliveiravsReal_RadiantTemp_East		4.11	-0.38		3.30	-0.34
OliveiravsReal_RadiantTemp_North		3.00	-0.84		3.03	-0.87
OliveiravsReal_AirTemp_West		6.12	-1.98		5.47	-1.95
OliveiravsReal_AirTemp_East		5.60	-0.56		4.09	-0.48
OliveiravsReal_AirTemp_North		3.37	-0.92		3.35	-0.94
OliveiravsReal_PV		22.65	-1.33		22.65	-1.33
OliveiravsDirint_HeatGain_West		7.61	-1.81		7.62	-1.82
OliveiravsDirint_HeatGain_East		5.89	-1.15		5.78	-1.20
OliveiravsDirint_HeatGain_North		8.13	2.77		8.03	2.75
OliveiravsDirint_RadiantTemp_West		1.03	-0.74		0.92	-0.70
OliveiravsDirint_RadiantTemp_East		0.97	-0.53		0.79	-0.48
OliveiravsDirint_RadiantTemp_North		0.31	-0.05		0.29	-0.07
OliveiravsDirint_AirTemp_West		1.23	-0.76		1.06	-0.71
OliveiravsDirint_AirTemp_East		1.59	-0.59		1.29	-0.52
OliveiravsDirint_AirTemp_North		0.54	-0.07		0.55	-0.10
OliveiravsDirint_PV		3.15	-0.95		3.15	-0.95

TABLE C.4: Statistical errors of ten variables for two solar radiation models – Brownsville

Location: Brownsville	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	19.93	-0.71	case_Glazing2_Blinds1	19.89	-0.62
DirintvsReal_HeatGain_East		24.20	4.90		24.25	4.85
DirintvsReal_HeatGain_North		20.77	-1.44		20.66	-1.43
DirintvsReal_RadiantTemp_West		1.01	-0.69		1.02	-0.73
DirintvsReal_RadiantTemp_East		0.83	0.38		0.71	0.34
DirintvsReal_RadiantTemp_North		0.57	-0.32		0.57	-0.33
DirintvsReal_AirTemp_West		1.00	-0.54		1.00	-0.60
DirintvsReal_AirTemp_East		0.97	0.18		0.78	0.20
DirintvsReal_AirTemp_North		0.61	-0.31		0.63	-0.32
DirintvsReal_PV		8.13	-0.82		8.13	-0.82
OliveiravsReal_HeatGain_West		19.56	-0.81		19.51	-0.73
OliveiravsReal_HeatGain_East		21.61	4.38		21.68	4.33
OliveiravsReal_HeatGain_North		20.42	2.18		20.33	2.18
OliveiravsReal_RadiantTemp_West		1.00	-0.67		1.01	-0.72
OliveiravsReal_RadiantTemp_East		0.82	0.39		0.69	0.35
OliveiravsReal_RadiantTemp_North		0.50	-0.19		0.50	-0.21
OliveiravsReal_AirTemp_West		1.01	-0.56		1.00	-0.62
OliveiravsReal_AirTemp_East		0.96	0.18		0.76	0.19
OliveiravsReal_AirTemp_North		0.56	-0.22		0.57	-0.24
OliveiravsReal_PV		8.22	-1.33		8.22	-1.33
OliveiravsDirint_HeatGain_West		6.24	-0.10		6.18	-0.10
OliveiravsDirint_HeatGain_East		6.44	-0.55		6.26	-0.54
OliveiravsDirint_HeatGain_North		8.80	3.57		8.74	3.56
OliveiravsDirint_RadiantTemp_West		0.21	0.01		0.20	0.01
OliveiravsDirint_RadiantTemp_East		0.22	0.01		0.19	0.00
OliveiravsDirint_RadiantTemp_North		0.17	0.13		0.15	0.12
OliveiravsDirint_AirTemp_West		0.28	-0.02		0.23	-0.02
OliveiravsDirint_AirTemp_East		0.53	0.00		0.39	-0.01
OliveiravsDirint_AirTemp_North		0.22	0.09		0.22	0.08
OliveiravsDirint_PV		2.07	-0.50		2.07	-0.50
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	21.20	-0.88	case_Glazing2_Blinds2	20.80	-0.79
DirintvsReal_HeatGain_East		24.89	4.84		24.25	4.90
DirintvsReal_HeatGain_North		25.07	-1.67		24.62	-1.63
DirintvsReal_RadiantTemp_West		1.07	-0.57		1.03	-0.64
DirintvsReal_RadiantTemp_East		1.21	0.71		0.95	0.54
DirintvsReal_RadiantTemp_North		0.59	-0.29		0.58	-0.30
DirintvsReal_AirTemp_West		1.01	-0.48		0.99	-0.54
DirintvsReal_AirTemp_East		1.04	0.36		0.84	0.29
DirintvsReal_AirTemp_North		0.59	-0.27		0.61	-0.28
DirintvsReal_PV		8.13	-0.82		8.13	-0.82
OliveiravsReal_HeatGain_West		20.93	-1.02		20.56	-0.93
OliveiravsReal_HeatGain_East		22.44	4.39		21.85	4.44
OliveiravsReal_HeatGain_North		24.81	2.60		24.39	2.62
OliveiravsReal_RadiantTemp_West		1.10	-0.58		1.03	-0.65
OliveiravsReal_RadiantTemp_East		1.15	0.65		0.92	0.50
OliveiravsReal_RadiantTemp_North		0.51	-0.09		0.50	-0.15
OliveiravsReal_AirTemp_West		1.06	-0.52		1.01	-0.58
OliveiravsReal_AirTemp_East		1.04	0.31		0.83	0.25
OliveiravsReal_AirTemp_North		0.51	-0.14		0.54	-0.18
OliveiravsReal_PV		8.22	-1.33		8.22	-1.33
OliveiravsDirint_HeatGain_West		6.70	-0.15		6.59	-0.14
OliveiravsDirint_HeatGain_East		6.71	-0.48		6.53	-0.49

OliveiravsDirint_HeatGain_North		10.31	4.20		10.15	4.18
OliveiravsDirint_RadiantTemp_West		0.34	-0.01		0.27	-0.01
OliveiravsDirint_RadiantTemp_East		0.30	-0.06		0.25	-0.05
OliveiravsDirint_RadiantTemp_North		0.25	0.20		0.20	0.16
OliveiravsDirint_AirTemp_West		0.35	-0.04		0.30	-0.04
OliveiravsDirint_AirTemp_East		0.55	-0.05		0.40	-0.04
OliveiravsDirint_AirTemp_North		0.24	0.12		0.24	0.10
OliveiravsDirint_PV		2.07	-0.50		2.07	-0.50
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	19.87	-0.71	case_Glazing2_Blinds3	19.86	-0.62
DirintvsReal_HeatGain_East		24.12	4.88		24.21	4.83
DirintvsReal_HeatGain_North		20.72	-1.45		20.65	-1.43
DirintvsReal_RadiantTemp_West		1.00	-0.66		1.01	-0.72
DirintvsReal_RadiantTemp_East		0.85	0.40		0.71	0.36
DirintvsReal_RadiantTemp_North		0.56	-0.31		0.57	-0.32
DirintvsReal_AirTemp_West		1.01	-0.55		1.01	-0.61
DirintvsReal_AirTemp_East		0.98	0.19		0.79	0.21
DirintvsReal_AirTemp_North		0.63	-0.31		0.63	-0.32
DirintvsReal_PV		8.13	-0.82		8.13	-0.82
OliveiravsReal_HeatGain_West		19.52	-0.81		19.48	-0.72
OliveiravsReal_HeatGain_East		21.56	4.37		21.65	4.33
OliveiravsReal_HeatGain_North		20.37	2.16		20.32	2.16
OliveiravsReal_RadiantTemp_West		0.98	-0.65		1.00	-0.71
OliveiravsReal_RadiantTemp_East		0.83	0.41		0.70	0.36
OliveiravsReal_RadiantTemp_North		0.50	-0.18		0.50	-0.21
OliveiravsReal_AirTemp_West		1.02	-0.57		1.01	-0.63
OliveiravsReal_AirTemp_East		0.97	0.19		0.78	0.20
OliveiravsReal_AirTemp_North		0.58	-0.22		0.57	-0.24
OliveiravsReal_PV		8.22	-1.33		8.22	-1.33
OliveiravsDirint_HeatGain_West		6.21	-0.10		6.17	-0.11
OliveiravsDirint_HeatGain_East		6.40	-0.53		6.25	-0.52
OliveiravsDirint_HeatGain_North		8.78	3.55		8.73	3.55
OliveiravsDirint_RadiantTemp_West		0.21	0.01		0.19	0.01
OliveiravsDirint_RadiantTemp_East		0.23	0.01		0.18	0.00
OliveiravsDirint_RadiantTemp_North		0.16	0.13		0.15	0.12
OliveiravsDirint_AirTemp_West		0.27	-0.02		0.25	-0.02
OliveiravsDirint_AirTemp_East		0.46	0.00		0.36	-0.01
OliveiravsDirint_AirTemp_North		0.23	0.09		0.22	0.08
OliveiravsDirint_PV		2.07	-0.50		2.07	-0.50

TABLE C.5: Statistical errors of ten variables for two solar radiation models – Chicago

Location: Chicago	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	17.44	-0.23	case_Glazing2_Blinds1	17.45	-0.25
DirintvsReal_HeatGain_East		18.92	-0.65		18.94	-0.61
DirintvsReal_HeatGain_North		17.44	0.47		17.24	0.45
DirintvsReal_RadiantTemp_West		1.38	0.39		1.23	0.43
DirintvsReal_RadiantTemp_East		1.67	-0.15		1.35	-0.14
DirintvsReal_RadiantTemp_North		0.77	0.38		0.77	0.40
DirintvsReal_AirTemp_West		1.61	0.34		1.35	0.39
DirintvsReal_AirTemp_East		2.33	-0.15		1.71	-0.13
DirintvsReal_AirTemp_North		0.83	0.34		0.83	0.36
DirintvsReal_PV		6.61	-0.48		6.61	-0.48
OliveiravsReal_HeatGain_West		16.48	-1.85		16.48	-1.88
OliveiravsReal_HeatGain_East		18.53	-2.01		18.52	-2.01
OliveiravsReal_HeatGain_North		18.43	4.10		18.21	4.04
OliveiravsReal_RadiantTemp_West		1.25	-0.09		1.09	-0.02
OliveiravsReal_RadiantTemp_East		1.73	-0.55		1.43	-0.51
OliveiravsReal_RadiantTemp_North		0.77	0.36		0.75	0.36
OliveiravsReal_AirTemp_West		1.50	-0.17		1.22	-0.08
OliveiravsReal_AirTemp_East		2.38	-0.58		1.79	-0.52
OliveiravsReal_AirTemp_North		0.81	0.29		0.78	0.30
OliveiravsReal_PV		6.65	-1.42		6.65	-1.42
OliveiravsDirint_HeatGain_West		8.16	-1.61		8.17	-1.63
OliveiravsDirint_HeatGain_East		7.00	-1.35		6.85	-1.39
OliveiravsDirint_HeatGain_North		9.33	3.65		9.21	3.61
OliveiravsDirint_RadiantTemp_West		0.77	-0.49		0.68	-0.46
OliveiravsDirint_RadiantTemp_East		0.77	-0.41		0.64	-0.37
OliveiravsDirint_RadiantTemp_North		0.24	-0.02		0.23	-0.03
OliveiravsDirint_AirTemp_West		0.93	-0.50		0.79	-0.47
OliveiravsDirint_AirTemp_East		1.33	-0.43		1.07	-0.39
OliveiravsDirint_AirTemp_North		0.43	-0.05		0.40	-0.06
OliveiravsDirint_PV		3.21	-0.94		3.21	-0.94
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	17.48	-0.36	case_Glazing2_Blinds2	17.11	-0.39
DirintvsReal_HeatGain_East		19.29	-0.72		18.64	-0.76
DirintvsReal_HeatGain_North		20.60	0.96		20.03	0.93
DirintvsReal_RadiantTemp_West		1.71	0.21		1.42	0.32
DirintvsReal_RadiantTemp_East		2.02	-0.32		1.63	-0.23
DirintvsReal_RadiantTemp_North		0.81	0.33		0.79	0.37
DirintvsReal_AirTemp_West		1.74	0.21		1.43	0.30
DirintvsReal_AirTemp_East		2.41	-0.29		1.76	-0.23
DirintvsReal_AirTemp_North		0.81	0.26		0.82	0.32
DirintvsReal_PV		6.61	-0.48		6.61	-0.48
OliveiravsReal_HeatGain_West		16.67	-2.04		16.35	-2.06
OliveiravsReal_HeatGain_East		18.91	-2.14		18.31	-2.19
OliveiravsReal_HeatGain_North		21.56	5.01		21.00	4.93
OliveiravsReal_RadiantTemp_West		1.69	-0.44		1.35	-0.24
OliveiravsReal_RadiantTemp_East		2.20	-0.91		1.77	-0.72
OliveiravsReal_RadiantTemp_North		0.93	0.41		0.83	0.38
OliveiravsReal_AirTemp_West		1.70	-0.45		1.35	-0.25
OliveiravsReal_AirTemp_East		2.59	-0.89		1.90	-0.71
OliveiravsReal_AirTemp_North		0.87	0.26		0.82	0.27
OliveiravsReal_PV		6.65	-1.42		6.65	-1.42
OliveiravsDirint_HeatGain_West		8.23	-1.67		8.10	-1.66
OliveiravsDirint_HeatGain_East		6.96	-1.41		6.75	-1.42

OliveiravsDirint_HeatGain_North		10.83	4.09		10.56	4.03
OliveiravsDirint_RadiantTemp_West		1.06	-0.65		0.86	-0.56
OliveiravsDirint_RadiantTemp_East		0.97	-0.58		0.81	-0.49
OliveiravsDirint_RadiantTemp_North		0.31	0.08		0.25	0.02
OliveiravsDirint_AirTemp_West		1.10	-0.66		0.91	-0.56
OliveiravsDirint_AirTemp_East		1.57	-0.59		1.06	-0.48
OliveiravsDirint_AirTemp_North		0.41	-0.01		0.38	-0.05
OliveiravsDirint_PV		3.21	-0.94		3.21	-0.94
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	17.37	-0.23	case_Glazing2_Blinds3	17.40	-0.24
DirintvsReal_HeatGain_East		18.84	-0.64		18.88	-0.59
DirintvsReal_HeatGain_North		17.36	0.48		17.19	0.48
DirintvsReal_RadiantTemp_West		1.38	0.40		1.23	0.43
DirintvsReal_RadiantTemp_East		1.69	-0.13		1.35	-0.13
DirintvsReal_RadiantTemp_North		0.78	0.39		0.77	0.39
DirintvsReal_AirTemp_West		1.62	0.35		1.36	0.39
DirintvsReal_AirTemp_East		2.31	-0.14		1.71	-0.14
DirintvsReal_AirTemp_North		0.85	0.35		0.82	0.35
DirintvsReal_PV		6.61	-0.48		6.61	-0.48
OliveiravsReal_HeatGain_West		16.43	-1.84		16.44	-1.88
OliveiravsReal_HeatGain_East		18.46	-2.00		18.46	-2.01
OliveiravsReal_HeatGain_North		18.31	4.07		18.14	4.05
OliveiravsReal_RadiantTemp_West		1.25	-0.08		1.09	-0.02
OliveiravsReal_RadiantTemp_East		1.75	-0.53		1.43	-0.48
OliveiravsReal_RadiantTemp_North		0.78	0.38		0.75	0.36
OliveiravsReal_AirTemp_West		1.49	-0.16		1.22	-0.08
OliveiravsReal_AirTemp_East		2.36	-0.58		1.78	-0.51
OliveiravsReal_AirTemp_North		0.83	0.30		0.78	0.29
OliveiravsReal_PV		6.65	-1.42		6.65	-1.42
OliveiravsDirint_HeatGain_West		8.15	-1.61		8.16	-1.63
OliveiravsDirint_HeatGain_East		7.04	-1.35		6.84	-1.40
OliveiravsDirint_HeatGain_North		9.26	3.61		9.20	3.59
OliveiravsDirint_RadiantTemp_West		0.77	-0.48		0.68	-0.45
OliveiravsDirint_RadiantTemp_East		0.78	-0.40		0.64	-0.35
OliveiravsDirint_RadiantTemp_North		0.24	-0.01		0.23	-0.03
OliveiravsDirint_AirTemp_West		0.94	-0.51		0.81	-0.47
OliveiravsDirint_AirTemp_East		1.32	-0.43		1.07	-0.38
OliveiravsDirint_AirTemp_North		0.44	-0.04		0.40	-0.06
OliveiravsDirint_PV		3.21	-0.94		3.21	-0.94

TABLE C.6: Statistical errors of ten variables for two solar radiation models – GlensFalls

Location: Glensfalls	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	19.97	-0.02	case_Glazing2_Blinds1	19.99	-0.03
DirintvsReal_HeatGain_East		20.48	-0.25		20.46	-0.22
DirintvsReal_HeatGain_North		18.98	0.15		18.80	0.12
DirintvsReal_RadiantTemp_West		1.61	0.44		1.48	0.47
DirintvsReal_RadiantTemp_East		1.61	-0.02		1.29	-0.02
DirintvsReal_RadiantTemp_North		0.77	0.39		0.78	0.41
DirintvsReal_AirTemp_West		1.92	0.40		1.64	0.45
DirintvsReal_AirTemp_East		2.59	-0.03		1.78	-0.04
DirintvsReal_AirTemp_North		0.86	0.36		0.83	0.39
DirintvsReal_PV		7.79	-0.10		7.79	-0.10
OliveiravsReal_HeatGain_West		19.12	-1.71		19.14	-1.73
OliveiravsReal_HeatGain_East		20.31	-1.74		20.23	-1.73
OliveiravsReal_HeatGain_North		20.10	3.08		19.92	3.05
OliveiravsReal_RadiantTemp_West		1.49	-0.10		1.34	-0.05
OliveiravsReal_RadiantTemp_East		1.69	-0.50		1.41	-0.46
OliveiravsReal_RadiantTemp_North		0.81	0.34		0.80	0.34
OliveiravsReal_AirTemp_West		1.80	-0.17		1.51	-0.10
OliveiravsReal_AirTemp_East		2.65	-0.53		1.86	-0.50
OliveiravsReal_AirTemp_North		0.85	0.30		0.82	0.30
OliveiravsReal_PV		7.73	-1.03		7.73	-1.03
OliveiravsDirint_HeatGain_West		8.70	-1.69		8.71	-1.70
OliveiravsDirint_HeatGain_East		7.63	-1.48		7.46	-1.51
OliveiravsDirint_HeatGain_North		9.55	2.94		9.48	2.94
OliveiravsDirint_RadiantTemp_West		0.91	-0.55		0.82	-0.52
OliveiravsDirint_RadiantTemp_East		0.90	-0.47		0.75	-0.44
OliveiravsDirint_RadiantTemp_North		0.27	-0.05		0.26	-0.06
OliveiravsDirint_AirTemp_West		1.14	-0.57		0.97	-0.55
OliveiravsDirint_AirTemp_East		1.69	-0.49		1.16	-0.46
OliveiravsDirint_AirTemp_North		0.42	-0.07		0.39	-0.09
OliveiravsDirint_PV		3.59	-0.93		3.59	-0.93
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	19.72	-0.05	case_Glazing2_Blinds2	19.30	-0.07
DirintvsReal_HeatGain_East		20.48	-0.26		19.80	-0.28
DirintvsReal_HeatGain_North		22.01	0.40		21.43	0.38
DirintvsReal_RadiantTemp_West		1.87	0.32		1.61	0.38
DirintvsReal_RadiantTemp_East		1.91	-0.12		1.53	-0.08
DirintvsReal_RadiantTemp_North		0.78	0.34		0.77	0.37
DirintvsReal_AirTemp_West		1.97	0.32		1.69	0.37
DirintvsReal_AirTemp_East		2.44	-0.11		1.78	-0.08
DirintvsReal_AirTemp_North		0.81	0.30		0.80	0.35
DirintvsReal_PV		7.79	-0.10		7.79	-0.10
OliveiravsReal_HeatGain_West		18.99	-1.77		18.60	-1.78
OliveiravsReal_HeatGain_East		20.31	-1.79		19.65	-1.81
OliveiravsReal_HeatGain_North		23.21	3.66		22.61	3.59
OliveiravsReal_RadiantTemp_West		1.83	-0.39		1.54	-0.23
OliveiravsReal_RadiantTemp_East		2.08	-0.77		1.69	-0.62
OliveiravsReal_RadiantTemp_North		0.89	0.37		0.84	0.36
OliveiravsReal_AirTemp_West		1.90	-0.40		1.60	-0.25
OliveiravsReal_AirTemp_East		2.61	-0.75		1.93	-0.62
OliveiravsReal_AirTemp_North		0.87	0.26		0.83	0.28
OliveiravsReal_PV		7.73	-1.03		7.73	-1.03
OliveiravsDirint_HeatGain_West		8.69	-1.72		8.55	-1.71
OliveiravsDirint_HeatGain_East		7.53	-1.52		7.33	-1.53

OliveiravsDirint_HeatGain_North		10.93	3.27		10.67	3.22
OliveiravsDirint_RadiantTemp_West		1.20	-0.71		0.99	-0.62
OliveiravsDirint_RadiantTemp_East		1.10	-0.65		0.91	-0.54
OliveiravsDirint_RadiantTemp_North		0.34	0.03		0.30	-0.02
OliveiravsDirint_AirTemp_West		1.31	-0.72		1.08	-0.62
OliveiravsDirint_AirTemp_East		1.70	-0.64		1.26	-0.53
OliveiravsDirint_AirTemp_North		0.45	-0.04		0.39	-0.07
OliveiravsDirint_PV		3.59	-0.93		3.59	-0.93
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	19.90	0.00	case_Glazing2_Blinds3	19.94	-0.02
DirintvsReal_HeatGain_East		20.38	-0.24		20.39	-0.18
DirintvsReal_HeatGain_North		18.85	0.14		18.69	0.11
DirintvsReal_RadiantTemp_West		1.61	0.45		1.48	0.47
DirintvsReal_RadiantTemp_East		1.64	-0.01		1.30	-0.03
DirintvsReal_RadiantTemp_North		0.79	0.40		0.79	0.41
DirintvsReal_AirTemp_West		1.92	0.42		1.65	0.45
DirintvsReal_AirTemp_East		2.59	-0.02		1.77	-0.04
DirintvsReal_AirTemp_North		0.87	0.37		0.84	0.39
DirintvsReal_PV		7.79	-0.10		7.79	-0.10
OliveiravsReal_HeatGain_West		19.05	-1.70		19.08	-1.73
OliveiravsReal_HeatGain_East		20.15	-1.72		20.16	-1.72
OliveiravsReal_HeatGain_North		19.94	3.04		19.77	3.01
OliveiravsReal_RadiantTemp_West		1.49	-0.09		1.35	-0.04
OliveiravsReal_RadiantTemp_East		1.72	-0.48		1.41	-0.45
OliveiravsReal_RadiantTemp_North		0.82	0.35		0.81	0.35
OliveiravsReal_AirTemp_West		1.81	-0.16		1.52	-0.09
OliveiravsReal_AirTemp_East		2.65	-0.53		1.85	-0.50
OliveiravsReal_AirTemp_North		0.86	0.30		0.83	0.30
OliveiravsReal_PV		7.73	-1.03		7.73	-1.03
OliveiravsDirint_HeatGain_West		8.69	-1.70		8.71	-1.71
OliveiravsDirint_HeatGain_East		7.60	-1.47		7.46	-1.53
OliveiravsDirint_HeatGain_North		9.57	2.91		9.50	2.90
OliveiravsDirint_RadiantTemp_West		0.92	-0.54		0.81	-0.52
OliveiravsDirint_RadiantTemp_East		0.90	-0.48		0.75	-0.43
OliveiravsDirint_RadiantTemp_North		0.28	-0.05		0.26	-0.06
OliveiravsDirint_AirTemp_West		1.14	-0.58		0.98	-0.55
OliveiravsDirint_AirTemp_East		1.64	-0.52		1.17	-0.46
OliveiravsDirint_AirTemp_North		0.45	-0.07		0.39	-0.09
OliveiravsDirint_PV		3.59	-0.93		3.59	-0.93

TABLE C.7: Statistical errors of ten variables for two solar radiation models – Los Angeles

Location: Los Angeles	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	13.93	2.08	case_Glazing2_Blinds1	13.86	2.04
DirintvsReal_HeatGain_East		16.57	1.61		16.25	1.54
DirintvsReal_HeatGain_North		21.61	1.19		21.57	1.16
DirintvsReal_RadiantTemp_West		0.75	0.33		0.76	0.33
DirintvsReal_RadiantTemp_East		0.90	0.25		0.77	0.27
DirintvsReal_RadiantTemp_North		0.48	0.20		0.47	0.20
DirintvsReal_AirTemp_West		1.11	0.24		1.05	0.26
DirintvsReal_AirTemp_East		2.10	0.14		1.53	0.21
DirintvsReal_AirTemp_North		0.59	0.17		0.56	0.18
DirintvsReal_PV		4.80	0.24		4.80	0.24
OliveiravsReal_HeatGain_West		13.14	0.56		13.07	0.57
OliveiravsReal_HeatGain_East		15.80	0.19		15.48	0.19
OliveiravsReal_HeatGain_North		24.35	4.56		24.30	4.53
OliveiravsReal_RadiantTemp_West		0.59	0.04		0.59	0.02
OliveiravsReal_RadiantTemp_East		0.87	0.10		0.69	0.08
OliveiravsReal_RadiantTemp_North		0.50	0.18		0.48	0.17
OliveiravsReal_AirTemp_West		1.03	-0.02		0.95	-0.04
OliveiravsReal_AirTemp_East		2.09	0.03		1.47	0.04
OliveiravsReal_AirTemp_North		0.61	0.13		0.56	0.13
OliveiravsReal_PV		4.79	-0.49		4.79	-0.49
OliveiravsDirint_HeatGain_West		7.09	-1.56		6.95	-1.51
OliveiravsDirint_HeatGain_East		7.22	-1.44		6.31	-1.37
OliveiravsDirint_HeatGain_North		8.79	3.40		8.72	3.41
OliveiravsDirint_RadiantTemp_West		0.45	-0.28		0.46	-0.31
OliveiravsDirint_RadiantTemp_East		0.49	-0.15		0.42	-0.18
OliveiravsDirint_RadiantTemp_North		0.15	-0.02		0.14	-0.04
OliveiravsDirint_AirTemp_West		0.93	-0.26		0.82	-0.30
OliveiravsDirint_AirTemp_East		1.82	-0.11		1.28	-0.17
OliveiravsDirint_AirTemp_North		0.36	-0.04		0.31	-0.06
OliveiravsDirint_PV		2.42	-0.74		2.42	-0.74
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	13.80	2.05	case_Glazing2_Blinds2	13.58	2.01
DirintvsReal_HeatGain_East		16.21	1.56		15.78	1.51
DirintvsReal_HeatGain_North		24.53	1.57		24.01	1.57
DirintvsReal_RadiantTemp_West		0.90	0.40		0.85	0.39
DirintvsReal_RadiantTemp_East		1.11	0.32		0.94	0.34
DirintvsReal_RadiantTemp_North		0.54	0.23		0.50	0.23
DirintvsReal_AirTemp_West		1.04	0.27		1.04	0.30
DirintvsReal_AirTemp_East		1.81	0.21		1.40	0.26
DirintvsReal_AirTemp_North		0.55	0.18		0.53	0.19
DirintvsReal_PV		4.80	0.24		4.80	0.24
OliveiravsReal_HeatGain_West		13.16	0.54		12.97	0.54
OliveiravsReal_HeatGain_East		15.51	0.24		15.12	0.21
OliveiravsReal_HeatGain_North		27.59	5.42		27.01	5.38
OliveiravsReal_RadiantTemp_West		0.75	0.03		0.64	0.03
OliveiravsReal_RadiantTemp_East		1.09	0.05		0.88	0.08
OliveiravsReal_RadiantTemp_North		0.61	0.29		0.53	0.23
OliveiravsReal_AirTemp_West		0.94	-0.04		0.91	-0.03
OliveiravsReal_AirTemp_East		1.80	0.04		1.36	0.06
OliveiravsReal_AirTemp_North		0.58	0.18		0.54	0.15
OliveiravsReal_PV		4.79	-0.49		4.79	-0.49
OliveiravsDirint_HeatGain_West		6.95	-1.54		6.84	-1.50
OliveiravsDirint_HeatGain_East		6.12	-1.34		5.96	-1.31

OliveiravsDirint_HeatGain_North		10.14	3.91		9.94	3.88
OliveiravsDirint_RadiantTemp_West		0.62	-0.36		0.56	-0.36
OliveiravsDirint_RadiantTemp_East		0.58	-0.26		0.51	-0.25
OliveiravsDirint_RadiantTemp_North		0.18	0.05		0.15	0.00
OliveiravsDirint_AirTemp_West		0.84	-0.31		0.83	-0.33
OliveiravsDirint_AirTemp_East		1.60	-0.18		1.17	-0.20
OliveiravsDirint_AirTemp_North		0.31	0.00		0.29	-0.04
OliveiravsDirint_PV		2.42	-0.74		2.42	-0.74
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	13.92	2.07	case_Glazing2_Blinds3	13.85	2.06
DirintvsReal_HeatGain_East		16.46	1.58		16.23	1.52
DirintvsReal_HeatGain_North		21.52	1.19		21.51	1.16
DirintvsReal_RadiantTemp_West		0.76	0.33		0.75	0.32
DirintvsReal_RadiantTemp_East		0.93	0.27		0.78	0.27
DirintvsReal_RadiantTemp_North		0.48	0.21		0.48	0.21
DirintvsReal_AirTemp_West		1.12	0.24		1.06	0.25
DirintvsReal_AirTemp_East		2.07	0.16		1.51	0.22
DirintvsReal_AirTemp_North		0.60	0.17		0.56	0.18
DirintvsReal_PV		4.80	0.24		4.80	0.24
OliveiravsReal_HeatGain_West		13.12	0.58		13.05	0.57
OliveiravsReal_HeatGain_East		15.76	0.23		15.46	0.19
OliveiravsReal_HeatGain_North		24.25	4.54		24.25	4.53
OliveiravsReal_RadiantTemp_West		0.60	0.03		0.57	0.01
OliveiravsReal_RadiantTemp_East		0.87	0.09		0.70	0.09
OliveiravsReal_RadiantTemp_North		0.50	0.18		0.48	0.17
OliveiravsReal_AirTemp_West		1.05	-0.04		0.95	-0.04
OliveiravsReal_AirTemp_East		2.11	0.02		1.45	0.04
OliveiravsReal_AirTemp_North		0.63	0.13		0.56	0.12
OliveiravsReal_PV		4.79	-0.49		4.79	-0.49
OliveiravsDirint_HeatGain_West		7.05	-1.53		6.92	-1.52
OliveiravsDirint_HeatGain_East		7.15	-1.37		6.24	-1.36
OliveiravsDirint_HeatGain_North		8.77	3.40		8.72	3.41
OliveiravsDirint_RadiantTemp_West		0.46	-0.30		0.46	-0.30
OliveiravsDirint_RadiantTemp_East		0.49	-0.18		0.42	-0.19
OliveiravsDirint_RadiantTemp_North		0.15	-0.02		0.14	-0.04
OliveiravsDirint_AirTemp_West		0.89	-0.28		0.82	-0.29
OliveiravsDirint_AirTemp_East		1.79	-0.14		1.24	-0.18
OliveiravsDirint_AirTemp_North		0.38	-0.04		0.31	-0.06
OliveiravsDirint_PV		2.42	-0.74		2.42	-0.74

TABLE C.8: Statistical errors of ten variables for two solar radiation models – Miami

Location: Miami	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	17.21	0.94	case_Glazing2_Blinds1	17.23	0.96
DirintvsReal_HeatGain_East		23.33	5.89		23.41	5.92
DirintvsReal_HeatGain_North		23.03	-0.75		23.08	-0.77
DirintvsReal_RadiantTemp_West		0.64	-0.37		0.65	-0.40
DirintvsReal_RadiantTemp_East		0.79	0.52		0.67	0.48
DirintvsReal_RadiantTemp_North		0.43	-0.12		0.43	-0.12
DirintvsReal_AirTemp_West		0.68	-0.30		0.68	-0.32
DirintvsReal_AirTemp_East		0.88	0.26		0.70	0.27
DirintvsReal_AirTemp_North		0.47	-0.16		0.48	-0.16
DirintvsReal_PV		6.75	0.04		6.75	0.04
OliveiravsReal_HeatGain_West		17.12	0.49		17.13	0.51
OliveiravsReal_HeatGain_East		21.01	5.18		21.06	5.19
OliveiravsReal_HeatGain_North		24.48	3.81		24.51	3.80
OliveiravsReal_RadiantTemp_West		0.66	-0.40		0.66	-0.43
OliveiravsReal_RadiantTemp_East		0.81	0.53		0.70	0.48
OliveiravsReal_RadiantTemp_North		0.43	0.00		0.42	-0.01
OliveiravsReal_AirTemp_West		0.70	-0.33		0.71	-0.37
OliveiravsReal_AirTemp_East		0.89	0.27		0.74	0.27
OliveiravsReal_AirTemp_North		0.45	-0.08		0.46	-0.09
OliveiravsReal_PV		6.68	-0.65		6.68	-0.65
OliveiravsDirint_HeatGain_West		6.79	-0.46		6.78	-0.46
OliveiravsDirint_HeatGain_East		7.01	-0.75		6.85	-0.77
OliveiravsDirint_HeatGain_North		10.09	4.53		10.10	4.54
OliveiravsDirint_RadiantTemp_West		0.22	-0.02		0.21	-0.03
OliveiravsDirint_RadiantTemp_East		0.25	0.00		0.21	0.00
OliveiravsDirint_RadiantTemp_North		0.16	0.12		0.15	0.11
OliveiravsDirint_AirTemp_West		0.29	-0.03		0.28	-0.04
OliveiravsDirint_AirTemp_East		0.57	0.01		0.45	0.00
OliveiravsDirint_AirTemp_North		0.20	0.07		0.21	0.07
OliveiravsDirint_PV		2.35	-0.69		2.35	-0.69
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	18.02	0.91	case_Glazing2_Blinds2	17.72	0.96
DirintvsReal_HeatGain_East		23.72	5.88		23.21	5.92
DirintvsReal_HeatGain_North		27.88	-1.03		27.47	-1.00
DirintvsReal_RadiantTemp_West		0.62	-0.20		0.60	-0.29
DirintvsReal_RadiantTemp_East		1.19	0.88		0.93	0.70
DirintvsReal_RadiantTemp_North		0.45	-0.08		0.43	-0.09
DirintvsReal_AirTemp_West		0.61	-0.19		0.63	-0.26
DirintvsReal_AirTemp_East		0.93	0.45		0.73	0.38
DirintvsReal_AirTemp_North		0.47	-0.11		0.47	-0.12
DirintvsReal_PV		6.75	0.04		6.75	0.04
OliveiravsReal_HeatGain_West		17.94	0.39		17.67	0.46
OliveiravsReal_HeatGain_East		21.52	5.22		21.03	5.25
OliveiravsReal_HeatGain_North		29.50	4.26		29.11	4.27
OliveiravsReal_RadiantTemp_West		0.70	-0.25		0.65	-0.34
OliveiravsReal_RadiantTemp_East		1.18	0.81		0.94	0.65
OliveiravsReal_RadiantTemp_North		0.51	0.12		0.45	0.06
OliveiravsReal_AirTemp_West		0.68	-0.24		0.68	-0.31
OliveiravsReal_AirTemp_East		0.95	0.41		0.79	0.36
OliveiravsReal_AirTemp_North		0.46	0.01		0.46	-0.03
OliveiravsReal_PV		6.68	-0.65		6.68	-0.65
OliveiravsDirint_HeatGain_West		7.15	-0.52		7.07	-0.51
OliveiravsDirint_HeatGain_East		7.16	-0.70		7.01	-0.71

OliveiravsDirint_HeatGain_North		11.65	5.23		11.52	5.22
OliveiravsDirint_RadiantTemp_West		0.37	-0.05		0.30	-0.05
OliveiravsDirint_RadiantTemp_East		0.34	-0.07		0.28	-0.05
OliveiravsDirint_RadiantTemp_North		0.25	0.20		0.20	0.15
OliveiravsDirint_AirTemp_West		0.37	-0.05		0.32	-0.05
OliveiravsDirint_AirTemp_East		0.54	-0.03		0.47	-0.02
OliveiravsDirint_AirTemp_North		0.23	0.12		0.23	0.09
OliveiravsDirint_PV		2.35	-0.69		2.35	-0.69
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	17.18	0.94	case_Glazing2_Blinds3	17.22	0.96
DirintvsReal_HeatGain_East		23.31	5.90		23.34	5.91
DirintvsReal_HeatGain_North		22.98	-0.75		23.06	-0.76
DirintvsReal_RadiantTemp_West		0.64	-0.36		0.64	-0.39
DirintvsReal_RadiantTemp_East		0.80	0.53		0.67	0.49
DirintvsReal_RadiantTemp_North		0.43	-0.11		0.42	-0.12
DirintvsReal_AirTemp_West		0.70	-0.31		0.68	-0.32
DirintvsReal_AirTemp_East		0.90	0.26		0.70	0.27
DirintvsReal_AirTemp_North		0.48	-0.16		0.48	-0.16
DirintvsReal_PV		6.75	0.04		6.75	0.04
OliveiravsReal_HeatGain_West		17.08	0.48		17.12	0.51
OliveiravsReal_HeatGain_East		20.98	5.20		21.00	5.19
OliveiravsReal_HeatGain_North		24.41	3.79		24.49	3.79
OliveiravsReal_RadiantTemp_West		0.65	-0.38		0.66	-0.42
OliveiravsReal_RadiantTemp_East		0.82	0.53		0.71	0.49
OliveiravsReal_RadiantTemp_North		0.43	0.01		0.42	-0.01
OliveiravsReal_AirTemp_West		0.71	-0.34		0.71	-0.37
OliveiravsReal_AirTemp_East		0.92	0.27		0.73	0.28
OliveiravsReal_AirTemp_North		0.46	-0.08		0.46	-0.09
OliveiravsReal_PV		6.68	-0.65		6.68	-0.65
OliveiravsDirint_HeatGain_West		6.79	-0.46		6.78	-0.45
OliveiravsDirint_HeatGain_East		7.00	-0.75		6.84	-0.76
OliveiravsDirint_HeatGain_North		10.06	4.51		10.09	4.52
OliveiravsDirint_RadiantTemp_West		0.22	-0.02		0.21	-0.03
OliveiravsDirint_RadiantTemp_East		0.25	0.00		0.21	0.00
OliveiravsDirint_RadiantTemp_North		0.16	0.12		0.15	0.11
OliveiravsDirint_AirTemp_West		0.29	-0.03		0.28	-0.04
OliveiravsDirint_AirTemp_East		0.65	0.01		0.46	0.01
OliveiravsDirint_AirTemp_North		0.21	0.07		0.21	0.07
OliveiravsDirint_PV		2.35	-0.69		2.35	-0.69

TABLE C.9: Statistical errors of ten variables for two solar radiation models – New York

Location: New York	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	16.97	1.26	case_Glazing2_Blinds1	17.02	1.26
DirintvsReal_HeatGain_East		18.54	0.65		18.53	0.74
DirintvsReal_HeatGain_North		17.80	-0.03		17.68	-0.05
DirintvsReal_RadiantTemp_West		1.07	0.55		0.95	0.54
DirintvsReal_RadiantTemp_East		1.28	0.32		1.03	0.27
DirintvsReal_RadiantTemp_North		0.50	0.33		0.50	0.33
DirintvsReal_AirTemp_West		1.24	0.49		1.05	0.49
DirintvsReal_AirTemp_East		1.92	0.35		1.35	0.30
DirintvsReal_AirTemp_North		0.57	0.29		0.55	0.31
DirintvsReal_PV		6.41	0.35		6.41	0.35
OliveiravsReal_HeatGain_West		15.50	-0.29		15.52	-0.30
OliveiravsReal_HeatGain_East		17.64	-0.73		17.47	-0.67
OliveiravsReal_HeatGain_North		19.26	3.69		19.11	3.66
OliveiravsReal_RadiantTemp_West		0.80	0.16		0.70	0.18
OliveiravsReal_RadiantTemp_East		1.06	0.01		0.85	0.00
OliveiravsReal_RadiantTemp_North		0.52	0.30		0.50	0.30
OliveiravsReal_AirTemp_West		1.00	0.10		0.81	0.13
OliveiravsReal_AirTemp_East		1.72	0.03		1.19	0.01
OliveiravsReal_AirTemp_North		0.54	0.25		0.51	0.25
OliveiravsReal_PV		6.06	-0.56		6.06	-0.56
OliveiravsDirint_HeatGain_West		7.90	-1.56		7.92	-1.58
OliveiravsDirint_HeatGain_East		7.10	-1.39		6.81	-1.43
OliveiravsDirint_HeatGain_North		10.01	3.72		9.94	3.71
OliveiravsDirint_RadiantTemp_West		0.63	-0.39		0.56	-0.36
OliveiravsDirint_RadiantTemp_East		0.62	-0.31		0.50	-0.27
OliveiravsDirint_RadiantTemp_North		0.19	-0.02		0.18	-0.04
OliveiravsDirint_AirTemp_West		0.77	-0.39		0.65	-0.37
OliveiravsDirint_AirTemp_East		1.25	-0.32		0.83	-0.29
OliveiravsDirint_AirTemp_North		0.30	-0.04		0.28	-0.06
OliveiravsDirint_PV		3.08	-0.91		3.08	-0.91
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	16.97	1.22	case_Glazing2_Blinds2	16.64	1.20
DirintvsReal_HeatGain_East		18.67	0.73		18.10	0.74
DirintvsReal_HeatGain_North		20.92	0.13		20.37	0.13
DirintvsReal_RadiantTemp_West		1.38	0.58		1.13	0.56
DirintvsReal_RadiantTemp_East		1.68	0.36		1.30	0.32
DirintvsReal_RadiantTemp_North		0.53	0.32		0.51	0.33
DirintvsReal_AirTemp_West		1.42	0.55		1.16	0.52
DirintvsReal_AirTemp_East		2.16	0.44		1.52	0.35
DirintvsReal_AirTemp_North		0.54	0.28		0.54	0.30
DirintvsReal_PV		6.41	0.35		6.41	0.35
OliveiravsReal_HeatGain_West		15.55	-0.38		15.23	-0.37
OliveiravsReal_HeatGain_East		17.60	-0.69		17.04	-0.69
OliveiravsReal_HeatGain_North		22.40	4.34		21.85	4.27
OliveiravsReal_RadiantTemp_West		1.04	0.08		0.83	0.12
OliveiravsReal_RadiantTemp_East		1.39	-0.08		1.07	-0.05
OliveiravsReal_RadiantTemp_North		0.64	0.37		0.57	0.34
OliveiravsReal_AirTemp_West		1.08	0.05		0.88	0.09
OliveiravsReal_AirTemp_East		1.90	0.01		1.35	0.00
OliveiravsReal_AirTemp_North		0.57	0.27		0.54	0.27
OliveiravsReal_PV		6.06	-0.56		6.06	-0.56
OliveiravsDirint_HeatGain_West		7.95	-1.61		7.84	-1.60
OliveiravsDirint_HeatGain_East		6.89	-1.43		6.71	-1.44

OliveiravsDirint_HeatGain_North		11.58	4.21		11.31	4.15
OliveiravsDirint_RadiantTemp_West		0.88	-0.51		0.71	-0.44
OliveiravsDirint_RadiantTemp_East		0.81	-0.44		0.65	-0.37
OliveiravsDirint_RadiantTemp_North		0.25	0.06		0.20	0.01
OliveiravsDirint_AirTemp_West		0.94	-0.51		0.77	-0.43
OliveiravsDirint_AirTemp_East		1.46	-0.44		1.06	-0.35
OliveiravsDirint_AirTemp_North		0.31	0.00		0.27	-0.04
OliveiravsDirint_PV		3.08	-0.91		3.08	-0.91
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	16.93	1.26	case_Glazing2_Blinds3	16.98	1.27
DirintvsReal_HeatGain_East		18.46	0.68		18.48	0.75
DirintvsReal_HeatGain_North		17.69	-0.01		17.60	-0.03
DirintvsReal_RadiantTemp_West		1.07	0.54		0.95	0.53
DirintvsReal_RadiantTemp_East		1.30	0.30		1.02	0.27
DirintvsReal_RadiantTemp_North		0.51	0.33		0.50	0.33
DirintvsReal_AirTemp_West		1.25	0.50		1.05	0.49
DirintvsReal_AirTemp_East		1.94	0.34		1.34	0.30
DirintvsReal_AirTemp_North		0.58	0.30		0.55	0.31
DirintvsReal_PV		6.41	0.35		6.41	0.35
OliveiravsReal_HeatGain_West		15.46	-0.29		15.49	-0.30
OliveiravsReal_HeatGain_East		17.48	-0.69		17.41	-0.68
OliveiravsReal_HeatGain_North		19.13	3.67		19.01	3.65
OliveiravsReal_RadiantTemp_West		0.80	0.17		0.69	0.18
OliveiravsReal_RadiantTemp_East		1.08	0.00		0.84	0.00
OliveiravsReal_RadiantTemp_North		0.52	0.31		0.50	0.30
OliveiravsReal_AirTemp_West		1.00	0.11		0.81	0.13
OliveiravsReal_AirTemp_East		1.73	0.02		1.17	0.02
OliveiravsReal_AirTemp_North		0.55	0.25		0.52	0.25
OliveiravsReal_PV		6.06	-0.56		6.06	-0.56
OliveiravsDirint_HeatGain_West		7.90	-1.57		7.92	-1.59
OliveiravsDirint_HeatGain_East		7.07	-1.39		6.78	-1.43
OliveiravsDirint_HeatGain_North		9.98	3.69		9.92	3.68
OliveiravsDirint_RadiantTemp_West		0.63	-0.38		0.55	-0.36
OliveiravsDirint_RadiantTemp_East		0.62	-0.30		0.49	-0.27
OliveiravsDirint_RadiantTemp_North		0.19	-0.02		0.18	-0.04
OliveiravsDirint_AirTemp_West		0.77	-0.40		0.64	-0.37
OliveiravsDirint_AirTemp_East		1.21	-0.32		0.83	-0.29
OliveiravsDirint_AirTemp_North		0.31	-0.04		0.28	-0.06
OliveiravsDirint_PV		3.08	-0.91		3.08	-0.91

TABLE C.10: Statistical errors of ten variables for two solar radiation models – Phoenix

Location: Phoenix	case	rmse	mbe	case	rmse	mbe
DirintvsReal_HeatGain_West	case_Glazing1_Blinds1	21.54	-0.38	case_Glazing2_Blinds1	21.60	-0.29
DirintvsReal_HeatGain_East		28.14	6.45		28.28	6.39
DirintvsReal_HeatGain_North		26.11	-1.54		26.06	-1.52
DirintvsReal_RadiantTemp_West		0.95	-0.63		0.96	-0.69
DirintvsReal_RadiantTemp_East		1.18	0.77		0.99	0.74
DirintvsReal_RadiantTemp_North		0.50	-0.13		0.51	-0.13
DirintvsReal_AirTemp_West		1.05	-0.51		1.00	-0.58
DirintvsReal_AirTemp_East		1.70	0.42		1.16	0.48
DirintvsReal_AirTemp_North		0.55	-0.21		0.56	-0.20
DirintvsReal_PV		8.05	-0.53		8.05	-0.53
OliveiravsReal_HeatGain_West		20.46	-0.99		20.46	-0.90
OliveiravsReal_HeatGain_East		24.92	5.20		25.04	5.15
OliveiravsReal_HeatGain_North		25.66	1.18		25.64	1.21
OliveiravsReal_RadiantTemp_West		1.00	-0.67		1.02	-0.73
OliveiravsReal_RadiantTemp_East		1.07	0.66		0.88	0.62
OliveiravsReal_RadiantTemp_North		0.51	-0.08		0.51	-0.09
OliveiravsReal_AirTemp_West		1.08	-0.56		1.06	-0.63
OliveiravsReal_AirTemp_East		1.62	0.37		1.08	0.39
OliveiravsReal_AirTemp_North		0.55	-0.18		0.55	-0.17
OliveiravsReal_PV		7.74	-0.91		7.74	-0.91
OliveiravsDirint_HeatGain_West		4.36	-0.62		4.33	-0.61
OliveiravsDirint_HeatGain_East		6.03	-1.34		5.62	-1.32
OliveiravsDirint_HeatGain_North		6.89	2.68		6.88	2.69
OliveiravsDirint_RadiantTemp_West		0.18	-0.04		0.17	-0.04
OliveiravsDirint_RadiantTemp_East		0.27	-0.11		0.22	-0.12
OliveiravsDirint_RadiantTemp_North		0.09	0.05		0.08	0.04
OliveiravsDirint_AirTemp_West		0.35	-0.05		0.30	-0.05
OliveiravsDirint_AirTemp_East		1.04	-0.06		0.60	-0.09
OliveiravsDirint_AirTemp_North		0.16	0.04		0.16	0.03
OliveiravsDirint_PV		1.53	-0.37		1.53	-0.37
DirintvsReal_HeatGain_West	case_Glazing1_Blinds2	21.41	-0.44	case_Glazing2_Blinds2	21.04	-0.37
DirintvsReal_HeatGain_East		28.13	6.31		27.48	6.35
DirintvsReal_HeatGain_North		32.57	-1.61		32.03	-1.60
DirintvsReal_RadiantTemp_West		0.95	-0.46		0.92	-0.56
DirintvsReal_RadiantTemp_East		1.58	1.17		1.27	0.98
DirintvsReal_RadiantTemp_North		0.52	-0.10		0.50	-0.10
DirintvsReal_AirTemp_West		0.95	-0.42		0.95	-0.50
DirintvsReal_AirTemp_East		1.53	0.61		1.15	0.60
DirintvsReal_AirTemp_North		0.53	-0.17		0.53	-0.16
DirintvsReal_PV		8.05	-0.53		8.05	-0.53
OliveiravsReal_HeatGain_West		20.28	-1.06		19.93	-0.98
OliveiravsReal_HeatGain_East		25.02	5.14		24.42	5.18
OliveiravsReal_HeatGain_North		31.89	1.63		31.38	1.63
OliveiravsReal_RadiantTemp_West		1.03	-0.54		1.00	-0.63
OliveiravsReal_RadiantTemp_East		1.39	0.97		1.11	0.80
OliveiravsReal_RadiantTemp_North		0.56	-0.01		0.52	-0.04
OliveiravsReal_AirTemp_West		1.03	-0.49		1.04	-0.57
OliveiravsReal_AirTemp_East		1.44	0.48		1.06	0.47
OliveiravsReal_AirTemp_North		0.52	-0.11		0.53	-0.13
OliveiravsReal_PV		7.74	-0.91		7.74	-0.91
OliveiravsDirint_HeatGain_West		4.42	-0.61		4.36	-0.60
OliveiravsDirint_HeatGain_East		5.60	-1.25		5.47	-1.24

OliveiravsDirint_HeatGain_North		8.20	3.19		8.09	3.18
OliveiravsDirint_RadiantTemp_West		0.27	-0.08		0.23	-0.07
OliveiravsDirint_RadiantTemp_East		0.34	-0.21		0.28	-0.18
OliveiravsDirint_RadiantTemp_North		0.13	0.09		0.11	0.06
OliveiravsDirint_AirTemp_West		0.39	-0.07		0.34	-0.07
OliveiravsDirint_AirTemp_East		0.89	-0.12		0.60	-0.13
OliveiravsDirint_AirTemp_North		0.18	0.06		0.17	0.04
OliveiravsDirint_PV		1.53	-0.37		1.53	-0.37
DirintvsReal_HeatGain_West	case_Glazing1_Blinds3	21.52	-0.38	case_Glazing2_Blinds3	21.59	-0.28
DirintvsReal_HeatGain_East		28.07	6.45		28.25	6.37
DirintvsReal_HeatGain_North		26.03	-1.55		26.01	-1.53
DirintvsReal_RadiantTemp_West		0.95	-0.62		0.96	-0.68
DirintvsReal_RadiantTemp_East		1.19	0.78		1.00	0.74
DirintvsReal_RadiantTemp_North		0.51	-0.13		0.51	-0.13
DirintvsReal_AirTemp_West		1.06	-0.52		1.01	-0.59
DirintvsReal_AirTemp_East		1.71	0.43		1.17	0.48
DirintvsReal_AirTemp_North		0.55	-0.21		0.56	-0.20
DirintvsReal_PV		8.05	-0.53		8.05	-0.53
OliveiravsReal_HeatGain_West		20.42	-1.00		20.45	-0.90
OliveiravsReal_HeatGain_East		24.89	5.21		25.01	5.14
OliveiravsReal_HeatGain_North		25.56	1.15		25.59	1.19
OliveiravsReal_RadiantTemp_West		0.99	-0.66		1.01	-0.73
OliveiravsReal_RadiantTemp_East		1.07	0.67		0.89	0.63
OliveiravsReal_RadiantTemp_North		0.51	-0.08		0.51	-0.09
OliveiravsReal_AirTemp_West		1.09	-0.57		1.07	-0.64
OliveiravsReal_AirTemp_East		1.61	0.36		1.08	0.39
OliveiravsReal_AirTemp_North		0.54	-0.17		0.56	-0.18
OliveiravsReal_PV		7.74	-0.91		7.74	-0.91
OliveiravsDirint_HeatGain_West		4.35	-0.62		4.33	-0.61
OliveiravsDirint_HeatGain_East		5.93	-1.33		5.61	-1.32
OliveiravsDirint_HeatGain_North		6.88	2.66		6.90	2.68
OliveiravsDirint_RadiantTemp_West		0.18	-0.04		0.17	-0.04
OliveiravsDirint_RadiantTemp_East		0.28	-0.11		0.21	-0.12
OliveiravsDirint_RadiantTemp_North		0.09	0.05		0.08	0.04
OliveiravsDirint_AirTemp_West		0.34	-0.04		0.30	-0.05
OliveiravsDirint_AirTemp_East		1.01	-0.07		0.58	-0.09
OliveiravsDirint_AirTemp_North		0.15	0.04		0.16	0.03
OliveiravsDirint_PV		1.53	-0.37		1.53	-0.37

# Bibliography

- [1] ASHRAE. Standard method of test for the evaluation of building energy analysis computer programs. *American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.: Atlanta* (2007).
- [2] ASHRAE. ASHRAE Handbook–Fundamentals: Fenestration. *Atlanta GA* (2009).
- [3] ASHRAE. ASHRAE Handbook–Fundamentals: Nonresidential Cooling and Heating Load Calculations. *Atlanta GA* (2009).
- [4] ASHRAE. ASHRAE Handbook–Fundamentals: Psychrometrics. *Atlanta GA* (2009).
- [5] ASHRAE. ASHRAE Handbook–Fundamentals: Ventilation and infiltration. *Atlanta GA* (2009).
- [6] BAHAR, Y. N., PERE, C., LANDRIEU, J., AND NICOLLE, C. A thermal simulation tool for building and its interoperability through the building information modeling (bim) platform. *Buildings* 3, 2 (2013), 380–398.
- [7] BAZJANAC, V. Implementation of semi-automated energy performance simulation: building geometry. In *CIB W78: 26th International Conference* (Istanbul, Turkey, 2009).
- [8] BAZJANAC, V. Space boundary requirements for modeling of building geometry for energy and other performance simulation. In *CIB W78: 27th International Conference* (Cairo, Egypt, 2010).
- [9] BAZJANAC, V., AND KIVINIEMI, A. Reduction, simplification, translation and interpretation in the exchange of model data. In *CIB W78: 24th International Conference* (Maribor, Slovenia, 2007).
- [10] BAZJANAC, V., AND MAILE, T. Ifc hvac interface to energyplus-a case of expanded interoperability for energy simulation. *Lawrence Berkeley National Laboratory* (2004).
- [11] BEETZ, J., VAN BERLO, L., DE LAAT, R., AND VAN DEN HELM, P. BIMserver. org - an open source IFC model server. In *CIB W78: 27th International Conference* (Cairo, Egypt, 2010).
- [12] BERKELEY, P., HAVES, P., AND KOLDERUP, E. IMPACT OF MODELER DECISIONS ON SIMULATION RESULTS.
- [13] BLANCK, M. Master’s thesis: Design and evaluation of an energy management system based on a model-predictive control approach, 2010.
- [14] BOLAND, J., RIDLEY, B., AND BROWN, B. Models of diffuse solar radiation. *Renewable Energy* 33, 4 (2008), 575–584.
- [15] BUILDINGSMART. Coordinate view version 2.0. "<http://www.buildingsmart-tech.org/specifications/ifc-view-definition/coordination-view-v2.0/summary>", 2013.
- [16] CLARKE, J. *Energy simulation in building design*. Routledge, 2012.
- [17] CRAWLEY, D., HAND, J., KUMMERT, M., AND GRIFFITH, B. Contrasting the capabilities of building energy performance simulation programs. *Building and Environment* 43, 4 (2008), 661–673.

- [18] CRAWLEY, D., LAWRIE, L., WINKELMANN, F., BUHL, W., HUANG, Y., PEDERSEN, C., STRAND, R., LIESEN, R., FISHER, D., WITTE, M., ET AL. EnergyPlus: creating a new-generation building energy simulation program. *Energy and Buildings* 33, 4 (2001), 319–331.
- [19] DE MIGUEL, A., BILBAO, J., AGUIAR, R., KAMBEZIDIS, H., AND NEGRO, E. Diffuse solar irradiation model evaluation in the north mediterranean belt area. *Solar Energy* 70, 2 (2001), 143–153.
- [20] DERVISHI, S., AND MAHDAVI, A. Computing diffuse fraction of global horizontal solar radiation: A model comparison. *Solar Energy* 86, 6 (2012), 1796–1802.
- [21] DOE. EnergyPlus: Guide for Interface Developers, 2013.
- [22] DOE. EnergyPlus: Input-Output Reference, 2014.
- [23] DROSCHE, A., SCHRANZHOFER, H., SANTIAGO, J., CONSTANTIN, A., STREBLOW, R., MULLER, D., EXIZIDOU, P., GIANNAKIS, G., AND ROVAS, D. Pebble deliverable 2.1: Integrated thermal simulation models for the three buildings, 2010.
- [24] DUFFIE, J. A., AND BECKMAN, W. A. *Solar engineering of thermal processes*. John Wiley & Sons, 2013.
- [25] ERBS, D., KLEIN, S., AND DUFFIE, J. Estimation of the diffuse radiation fraction for hourly, daily and monthly-average global radiation. *Solar Energy* 28, 4 (1982), 293–302.
- [26] FRITZSON, P., AND ENGELSON, V. Modelica – a unified object-oriented language for system modeling and simulation. In *ECOOP’98—Object-Oriented Programming*. Springer, 1998, pp. 67–90.
- [27] GEORGESCU, M., AND MEZIĆ, I. Building energy modeling: A systematic approach to zoning and model reduction using koopman mode analysis. *Energy and Buildings* 86 (2015), 794–802.
- [28] GIANNAKIS, G., KATSIGARAKIS, K., KONTES, G., AND ROVAS, D. Co-simulation setup for online model-assisted control design. In *CLIMA 2013: 11th REHVA World Congress* (Prague, Czech Republic, 2013).
- [29] GIANNAKIS, G., KONTES, G., KOSMATOPOULOS, E., AND ROVAS, D. A model-assisted adaptive controller fine-tuning methodology for efficient energy use in buildings. In *Mediterranean Conference on Control & Automation* (June 2011), pp. 49–54.
- [30] GIANNAKIS, G., PICHLER, M., KONTES, G., SCHRANZHOFER, H., AND ROVAS, D. Simulation speedup techniques for computationally demanding tasks. In *BS 2013: 13th Conference of the International Building Performance Simulation Association* (2013).
- [31] HAWLADER, M. Diffuse, global and extra-terrestrial solar radiation for singapore. *International journal of ambient energy* 5, 1 (1984), 31–38.
- [32] HITCHCOCK, R., AND WONG, J. Transforming IFC architectural view bims for energy simulation. In *Proceedings of Building Simulation Conference* (Sydney, Australia, 2011), pp. 1089–1095.
- [33] HITTLE, D. Calculating building heating and cooling loads using the frequency response of multilayered slabs. *Technical Manuscript - United States Army Corps of Engineers, Construction Engineering Research Laboratory* (1981).
- [34] HOLLANDS, K. A derivation of the diffuse fraction’s dependence on the clearness index. *Solar energy* 35, 2 (1985), 131–136.
- [35] HOLLANDS, K., AND CRHA, S. An improved model for diffuse radiation: correction for atmospheric back-scattering. *Solar energy* 38, 4 (1987), 233–236.
- [36] HONG, T., BUHL, F., AND HAVES, P. Energyplus run time analysis, 2009.
- [37] HÖTTGES, K., SANTIAGO, J., GARCÍA, M., KONTES, G., ROVAS, D., VALMASEDA, C., HIDALGO, O., AND MARTÍN, J. BaaS Project Deliverable 4.2: Development of simulation models for the pilot buildings, 2014.

- [38] ISO13790. ISO 13790: Energy performance of buildings – Calculation of energy use for space heating and cooling. *European Committee for Standardization (CEN), Brussels* (2008).
- [39] ISO16739. ISO 16739: Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries. *European Committee for Standardization (CEN), Brussels* (2013).
- [40] JACKSON, D. Boundary representation modelling with local tolerances. In *ACM: 3rd Symposium on Solid modeling and applications* (1995), ACM, pp. 247–254.
- [41] JACOVIDES, C., TYMVIOS, F., ASSIMAKOPOULOS, V., AND KALTSOUNIDES, N. Comparative study of various correlations in estimating hourly diffuse fraction of global solar radiation. *Renewable Energy* 31, 15 (2006), 2492–2504.
- [42] JOKELA, M., KEINÄNEN, A., LAHTELA, H., LASSILA, K., AND GRANLUND, O. Integrated building simulation tool RIUSKA. In *BS 1997: 5th Conference of the International Building Performance Simulation Association* (Prague, Czech Republic, 1997).
- [43] KARATASOU, S., SANTAMOURIS, M., AND GEROS, V. Modeling and predicting building’s energy use with artificial neural networks: Methods and results. *Energy and Buildings* 38, 8 (2006), 949–958.
- [44] KAROLA, A., LAHTELA, H., HÄNNINEN, R., HITCHCOCK, R., CHEN, Q., DAJKA, S., AND HAGSTRÖM, K. BSPPro com-server interoperability between software tools using industrial foundation classes. *Energy and Buildings* 34, 9 (2002), 901–907.
- [45] KASTEN, F. A simple parameterization of the pyrheliometric formula for determining the linke turbidity factor. *Meteorologische Rundschau* 33 (1980), 124–127.
- [46] KASTEN, F., AND CZEPLAK, G. Solar and terrestrial radiation dependent on the amount and type of cloud. *Solar energy* 24, 2 (1980), 177–189.
- [47] KASTEN, F., AND YOUNG, A. T. Revised optical air mass tables and approximation formula. *Appl. Opt* 28, 22 (1989), 4735–4738.
- [48] KHEMLANI, L. Solibri model checker. *CADENCE-AUSTIN-* (2002), 32–34.
- [49] KLEIN, S., BECKMAN, W., AND DUFFIE, J. TRNSYS - A TRANSIENT SIMULATION PROGRAM. *ASHRAE Transactions* 82, 1 (1976), 623–633.
- [50] KOLOKOTSA, D., POULIEZOS, A., STAVRAKAKIS, G., AND LAZOS, C. Predictive control techniques for energy and indoor environmental quality management in buildings. *Building and Environment* 44, 9 (2009), 1850–1863.
- [51] KONTES, G., GIANNAKIS, G., KATSIGARAKIS, K., KOSMATOPOULOS, E., ROVAS, D., CONSTANTIN, A., FUETTERER, J., PICHLER, M., SCHRANZHOFER, H., AND RODRIGUEZ, J. PEBBLE Deliverable 6.1: Demonstration Report, 2012.
- [52] KONTES, G., GIANNAKIS, G., KOSMATOPOULOS, E. B., AND ROVAS, D. Adaptive-fine tuning of building energy management systems using co-simulation. In *IEEE International Conference on Control Applications (CCA)* (Zagreb, Croatia, 2012), pp. 1664–1669.
- [53] KONTES, G., VALMASEDA, C., GIANNAKIS, G., KATSIGARAKIS, K., AND ROVAS, D. Intelligent BEMS design using detailed thermal simulation models and surrogate-based stochastic optimization. *Journal of Process Control* (2014).
- [54] LAM, J. C., AND LI, D. H. Correlation between global solar radiation and its direct and diffuse components. *Building and environment* 31, 6 (1996), 527–535.
- [55] LEE, K., AND BRAUN, J. Model-based demand-limiting control of building thermal mass. *Building and Environment* 43, 10 (2008), 1633–1646.

- [56] LILIS, G., GIANNAKIS, G., KONTES, G., AND ROVAS, D. Semi-automatic thermal simulation model generation from IFC data. In *European Conference on Product and Process Modelling (ECPPM)* (Vienna, Austria, 2014).
- [57] LILIS, G., GIANNAKIS, G., AND ROVAS, D. Detection and semi-automatic correction of geometric inaccuracies in ifc file. In *preparation* (2014).
- [58] LILIS, G. N., SKLIVANIOTIS, K. F., GIANNAKIS, G. I., AND ROVAS, D. V. Src: A systemic approach to building thermal simulation. In *BS 2013: 13th Conference of the International Building Performance Simulation Association* (2013), pp. 3192–3199.
- [59] LOUCHE, A., NOTTON, G., POGGI, P., AND SIMONNOT, G. Correlations for direct normal and global horizontal irradiation on a french mediterranean site. *Solar Energy* 46, 4 (1991), 261–266.
- [60] MACAGNAN, M., LORENZO, E., AND JIMENEZ, C. Solar radiation in madrid. *International journal of solar energy* 16, 1 (1994), 1–14.
- [61] MAILE, T. *Comparing measured and simulated building energy performance data*. PhD thesis, Stanford University, 2010.
- [62] MAILE, T., FISCHER, M., AND BAZJANAC, V. Building energy performance simulation tools-a life-cycle and interoperable perspective. *Center for Integrated Facility Engineering (CIFE) Working Paper 107* (2007), 1–49.
- [63] MAILE, T., O'DONNELL, J., BAZJANAC, V., AND ROSE, C. BIM-Geometry Modelling Guidelines for Building Energy Performance Simulation. In *BS 2013: 13th Conference of the International Building Performance Simulation Association* (2013).
- [64] MAIMON, O., AND ROKACH, L. *Data mining and knowledge discovery handbook*, vol. 2. Springer, 2005.
- [65] MALDONADO, E. *Implementing the Energy Performance of Building Directive (EPBD): Featuring Country Reports 2012*. ADENE, Agência Para a Energia, 2013.
- [66] MAXWELL, E. L. A quasi-physical model for converting hourly global horizontal to direct normal insolation, 1987.
- [67] MEZIĆ, I. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics* 41, 1-3 (2005), 309–325.
- [68] MODELISAR. Functional Mock-up Interface for Co-Simulation. *Accessed March* (2014).
- [69] MOSS, K. *Heat and mass transfer in buildings*. Taylor & Francis Group, 2007.
- [70] NGUYEN, A.-T., REITER, S., AND RIGO, P. A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy* 113 (2014), 1043–1058.
- [71] NOTTON, G., CRISTOFARI, C., MUSELLI, M., AND POGGI, P. Calculation on an hourly basis of solar diffuse irradiances from global data for horizontal surfaces in ajaccio. *Energy conversion and management* 45, 18 (2004), 2849–2866.
- [72] NOUIDUI, T., LORENZETTI, D., AND WETTER, M. EnergyPlusToFMU, 2013.
- [73] O'DONNELL, J., MAILE, T., ROSE, C., MRAZOVIĆ, N., MORRISSEY, E., REGNIER, C., PARRISH, K., AND V., B. Transforming BIM to BEM: Generation of building geometry for the NASA ames sustainability base BIM. Tech. rep., Simulation Research Group, 2013.
- [74] O'DONNELL, J., SEE, R., ROSE, C., MAILE, T., BAZJANAC, V., AND P., H. SimModel: A domain data model for whole building energy simulation. In *Proceedings of Building Simulation Conference* (Sydney, Australia, 2011), pp. 382–389.

- [75] OLDEWURTEL, F., PARISIO, A., JONES, C., GYALISTRAS, D., GWERDER, M., STAUCH, V., LEHMANN, B., AND MORARI, M. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings* 45 (2012), 15–27.
- [76] OLIVEIRA, A. P., ESCOBEDO, J. F., MACHADO, A. J., AND SOARES, J. Correlation models of diffuse solar-radiation applied to the city of sao paulo, brazil. *Applied Energy* 71, 1 (2002), 59–73.
- [77] ORGILL, J., AND HOLLANDS, K. Correlation equation for hourly diffuse radiation on a horizontal surface. *Solar energy* 19, 4 (1977), 357–359.
- [78] PEREZ, R., INEICHEN, P., MAXWELL, E., SEALS, R., AND ZELENKA, A. Dynamic models for hourly global-to-direct irradiance conversion. In *1991 Solar world congress: Proceedings of the biennial congress of the international solar energy society* (1991), vol. 1, pp. 951–956.
- [79] PICHLER, M., DROESCHER, A., SCHRANZHOFER, H., KONTES, G., GIANNAKIS, G., KOSMATOPOULOS, E., AND ROVAS, D. Simulation-assisted building energy performance improvement using sensible control decisions. 3rd ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings (Accepted).
- [80] PICHLER, M., DROESCHER, A., SCHRANZHOFER, H., CONSTANTIN, A., HUBER, M., GIANNAKIS, G., EXIZIDOU, N., AND ROVAS, D. PEBBLE Deliverable 2.3: Simulation Model Improvements, 2011.
- [81] PRATT, K., JONES, N., SCHUMANN, L., BOSWORTH, D., AND HEUMANN, A. Automated translation of architectural models for energy simulation. In *Symposium on Simulation for Architecture and Urban Design* (Orlando, USA, 2012), p. 6.
- [82] PRÍVARA, S., CIGLER, J., VÁNA, Z., OLDEWURTEL, F., SAGERSCHNIG, C., AND ŽÁČEKOVÁ, E. Building modeling as a crucial part for building predictive control. *Energy and Buildings* 56 (2013), 8–22.
- [83] PRÍVARA, S., ŠIROKÝ, J., FERKL, L., AND CIGLER, J. Model predictive control of a building heating system: The first experience. *Energy and Buildings* 43, 2 (2011), 564–572.
- [84] PRÍVARA, S., VÁNA, Z., GYALISTRAS, D., CIGLER, J., SAGERSCHNIG, C., MORARI, M., AND FERKL, L. Modeling and identification of a large multi-zone office building. In *Control Applications (CCA), 2011 IEEE International Conference on* (2011), IEEE, pp. 55–60.
- [85] PRÍVARA, S., VÁNA, Z., ŽÁČEKOVÁ, E., AND CIGLER, J. Building modeling: Selection of the most appropriate model for predictive control. *Energy and Buildings* 55 (2012), 341–350.
- [86] REINDL, D., BECKMAN, W., AND DUFFIE, J. Diffuse fraction correlations. *Solar energy* 45, 1 (1990), 1–7.
- [87] RENO, M. J., HANSEN, C. W., AND STEIN, J. S. Global horizontal irradiance clear sky models: Implementation and analysis. *SANDIA report SAND2012-2389* (2012).
- [88] RIDLEY, B., BOLAND, J., AND LAURET, P. Modelling of diffuse solar fraction with multiple predictors. *Renewable Energy* 35, 2 (2010), 478–483.
- [89] ROBERT, S., MAZZA, D., HILAIRE, B., SETTE, P., AND VINOT, B. An approach to enhancing the connection between bim models and building energy simulation–hvac systems in the loop. *eWork and eBusiness in Architecture, Engineering and Construction: ECPPM 2014* (2014), 149.
- [90] ROSE, C., AND BAZJANAC, V. An algorithm to generate space boundaries for building energy simulation. *Engineering with Computers* (2013), 1–10.
- [91] SEEM, J. E. Modeling of heat transfer in buildings. Tech. rep., Wisconsin Univ., Madison (USA), 1987.
- [92] SKARTVEIT, A., AND OLSETH, J. A. A model for the diffuse fraction of hourly global radiation. *Solar Energy* 38, 4 (1987), 271–274.
- [93] SKARTVEIT, A., OLSETH, J. A., AND TUFT, M. E. An hourly diffuse fraction model with correction for variability and surface albedo. *Solar Energy* 63, 3 (1998), 173–183.

- [94] STURZENEGGER, D., GYALISTRAS, D., MORARI, M., AND SMITH, R. S. Semi-Automated Modular Modeling of Buildings for Model Predictive Control. In *Buildsys'12* (2012), pp. 99–106.
- [95] STURZENEGGER, D., GYALISTRAS, D., SEMERARO, V., MORARI, M., AND SMITH, R. S. Brcm matlab toolbox: Model generation for model predictive building control. In *American Control Conference (ACC), 2014* (2014), IEEE, pp. 1063–1069.
- [96] SUSUKI, Y., AND MEZIC, I. Nonlinear koopman modes of coupled swing dynamics and coherency identification. In *Power and Energy Society General Meeting, 2010 IEEE* (2010), IEEE, pp. 1–8.
- [97] THIBAUT, W., AND NAYLOR, B. Set operations on polyhedra using binary space partitioning trees. In *ACM 1987: 14th annual conference on Computer graphics and interactive techniques* (1987), vol. 21, pp. 153–162.
- [98] THOMAS, D., MILLER, C., KÄMPF, J., AND SCHLUETER, A. Multiscale co-simulation of energyplus and citysim models derived from a building information model. In *Bausim 2014: Fifth German-Austrian IBPSA Conference* (2014).
- [99] TORRES, J., DE BLAS, M., GARCÍA, A., AND DE FRANCISCO, A. Comparative study of various models in estimating hourly diffuse solar irradiance. *Renewable Energy* 35, 6 (2010), 1325–1332.
- [100] VANA, Z., KUBECEK, J., AND FERKL, L. Notes on finding black-box model of a large building. In *Proceedings of the IEEE International Conference on Control Applications (CCA)* (2010), pp. 1017–1022.
- [101] VATTI, B. A generic solution to polygon clipping. *Communications of the ACM* 35, 7 (1992), 56–63.
- [102] VIGNOLA, F., AND MCDANIELS, D. Correlations between diffuse and global insolation for the pacific north-west. *Solar energy* 32, 2 (1984), 161–168.
- [103] WANG, S., AND CHEN, Y. Transient heat flow calculation for multilayer constructions using a frequency-domain regression method. *Building and environment* 38, 1 (2003), 45–61.
- [104] WEISE, M., LIEBICH, T., SEE, R., BAZJANAC, V., LAINE, T., AND WELLE, B. Implementation guide: Space boundaries for energy analysis. *General Services Administration (GSA) and Open Geospatial Consortium (OGC)* (2011).
- [105] WETTER, M. Co-simulation of building energy and control systems with the building controls virtual test bed. *Journal of Building Performance Simulation* 4, 3 (2011), 185–203.