



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΧΗΜΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

Πολυκριτηριακή βελτιστοποίηση για την επιλογή
μέτρων εξοικονόμησης ενέργειας σε κτίρια

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΚΩΝΣΤΑΝΤΙΝΟΣ ΖΙΩΖΑΣ

Επιβλέπων Καθηγητής:

Γεώργιος Μαυρωτάς, Αναπληρωτής Καθηγητής ΕΜΠ

ΑΘΗΝΑ 2016

ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία, εκπονήθηκε στο πλαίσιο της ολοκλήρωσης των προπτυχιακών σπουδών μου στην σχολή Χημικών Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου, στο εργαστήριο Βιομηχανικής και Ενεργειακής Οικονομίας. Η θεματική ενότητα της μελέτης εντάσσεται στον τομέα της Επιχειρησιακής Έρευνας και συγκεκριμένα στον τομέα του Πολυκριτηριακού Μαθηματικού Προγραμματισμού.

Σκοπός της εργασίας αυτής είναι η χρήση Πολυκριτηριακού Μαθηματικού Προγραμματισμού για τον ενεργειακό σχεδιασμό κτιρίων. Χρησιμοποιείται μαθηματικό μοντέλο σε γλώσσα GAMS, με σκοπό την ελαχιστοποίηση των αντικειμενικών συναρτήσεων, του συνολικού ετήσιου κόστους, της συνολικής κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα και την επιλογή των αντίστοιχων συστημάτων.

Με την ολοκλήρωση της συγγραφής, θα ήθελα να ευχαριστήσω θερμά τον Αναπληρωτή Καθηγητή κ. Γεώργιο Μαυρωτά για την επίβλεψη της παρούσας εργασίας και την συνεχή καθοδήγηση του καθ' όλη την διάρκεια της μελέτης αυτής, καθώς και τον υποψήφιο διδάκτορα κ. Μάριο Καρμέλλο για την πολύτιμη βοήθειά του, χωρίς την οποία θα ήταν ιδιαίτερα δύσκολη η ολοκλήρωση αυτής της εργασίας.

Επίσης θα ήθελα να ευχαριστήσω τους γονείς μου και τους φίλους μου που μου συμπαραστάθηκαν καθ' όλη την διάρκεια των σπουδών μου. Τέλος θα ήθελα να ευχαριστήσω από καρδιάς τον μαθηματικό μου κ. Σωτήριο Εξαρχάκο, που με την μαθηματική παιδεία που μου προσέφερε ήμουν σε θέση να ολοκληρώσω με επιτυχία τις σπουδές μου.

ΠΕΡΙΛΗΨΗ

Τα περιβαλλοντικά προβλήματα που αντιμετωπίζει ο σύγχρονος κόσμος έχουν οδηγήσει την παγκόσμια επιστημονική κοινότητα να αναζητά συνεχώς νέους τρόπους για την βελτίωση της ενεργειακής απόδοσης των κτιρίων. Τα κτίρια είναι υπεύθυνα για το 40% της παγκόσμιας κατανάλωσης ενέργειας.

Στην παρούσα εργασία γίνεται μια προσπάθεια ανάπτυξης ενός μοντέλου σε γλώσσα GAMS με χρήση πολυκριτηριακού μαθηματικού προγραμματισμού που σκοπό έχει τον αποδοτικό ενεργειακό σχεδιασμό ενός κτιρίου. Σκοπός του μοντέλου είναι η εξαγωγή του πεδίου λύσεων του προβλήματος, έτσι ώστε ο αποφασίζων να είναι σε θέση να επιλέξει τα κατάλληλα συστήματα που θα ικανοποιούν τα κριτήρια του.

Αφού γίνει μια εισαγωγή στον μαθηματικό προγραμματισμό και στην μέθοδο που θα χρησιμοποιηθεί, γίνεται μια εισαγωγή στον ενεργειακό σχεδιασμό των κτιρίων. Μετά περιγράφονται οι μεταβλητές, οι παράμετροι, οι περιορισμοί και οι αντικειμενικές του μοντέλου. Ακολουθεί η παρουσίαση των παραμέτρων του κτιρίου της μελέτης. Έγινε μελέτη 2 περιπτώσεων. Μίας περίπτωσης με σκοπό την ελαχιστοποίηση των αντικειμενικών συναρτήσεων του συνολικού ετήσιου κόστους, της συνολικής κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα, και μια περίπτωσης με σκοπό την ελαχιστοποίηση του συνολικού ετήσιου κόστους και της συνολικής κατανάλωσης πρωτογενούς ενέργειας.

Αφού εκτελεστεί ο κώδικας σε γλώσσα GAMS, χρησιμοποιώντας την μέθοδο των περιορισμών (e-constraint), παρουσιάζονται και σχολιάζονται τα αποτελέσματα μέσω των κατάλληλων διαγραμμάτων. Καταλήγουμε στο συμπέρασμα ότι τα κριτήρια του συνολικού ετήσιου κόστους, της πρωτογενούς κατανάλωσης ενέργειας και των εκπομπών διοξειδίου του άνθρακα είναι αλληλοσυγκρουόμενα. Τέλος, προτείνονται ιδέες για περαιτέρω μελέτη.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ.....	1
ΠΕΡΙΛΗΨΗ	2
Λίστα Εικόνων	5
Λίστα Πινάκων.....	6
ΚΕΦΑΛΑΙΟ 1 : ΕΙΣΑΓΩΓΗ	8
ΚΕΦΑΛΑΙΟ 2: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ.....	10
ΚΕΦΑΛΑΙΟ 3: ΜΑΘΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ.....	11
3.1 Εισαγωγή στον Μαθηματικό Προγραμματισμό.....	11
3.2 Εφαρμογές Γραμμικού Προγραμματισμού	12
3.3 Πολυκριτηριακός Προγραμματισμός	13
3.4 Μέθοδοι του Πολυκριτηριακού Μαθηματικού Προγραμματισμού.....	15
ΚΕΦΑΛΑΙΟ 4: ΕΝΕΡΓΕΙΑΚΟΣ ΣΧΕΔΙΑΣΜΟΣ ΚΤΙΡΙΩΝ	18
4.1 Ενεργειακή κατανάλωση Κτιρίων	18
4.2 Βασικά συστήματα ενός κτιρίου	19
4.2.1 Ενεργειακό Ισοζύγιο – Γενικά για Θερμικές Απώλειες Κτιρίων	19
4.2.2 Θερμικές Απώλειες σε Εξωτερικούς τοίχους –Παράθυρα-Οροφές.....	20
4.2.3 Θερμότητα παραγόμενη από τους ανθρώπους	23
4.2.4 Θερμότητα από τα συστήματα φωτισμού	23
4.2.5 Θερμότητα από τις ηλεκτρικές συσκευές	24
4.3 Ενεργειακά συστήματα κτιρίων	25
4.3.1 Ηλιακοί συλλέκτες.....	25
4.3.2 Λέβητες.....	25
4.3.3 Συσκευές θέρμανσης με ηλεκτρική αντίσταση	25
4.3.4 Κλιματιστικά	26
4.3.5 Θερμοσίφωνες	26
ΚΕΦΑΛΑΙΟ 5 : ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΜΟΝΤΕΛΟΥ.....	27
5.1 Σκοπός του μοντέλου	27

5.2 Περιγραφή των εξισώσεων του μοντέλου	27
5.2.1 Αντικειμενικές Συναρτήσεις	27
5.2.2 Εξισώσεις επιλογής μερών του κτιριακού κελύφους	28
5.2.3 Εξισώσεις επιλογής ενεργειακών συστημάτων	29
5.2.4 Εξισώσεις υπολογισμού ενεργειακών αναγκών	32
5.2.5 Υπολογισμός κατανάλωσης πρωτογενούς ενέργειας.....	37
5.2.6 Υπολογισμός ολικών συντελεστών θερμότητας τοίχων-δαπέδων-οροφών	39
5.2.7 Υπολογισμός εκπομπών διοξειδίου του άνθρακα	39
5.2.8 Υπολογισμός συνολικού ετήσιου κόστους.....	40
ΚΕΦΑΛΑΙΟ 6 : ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ	43
6.1 Περιγραφή του κτιρίου και των συστημάτων	43
6.2 Αποτελέσματα Περίπτωσης 3 αντικειμενικών συναρτήσεων	49
6.3 Αποτελέσματα Περίπτωσης 2 αντικειμενικών συναρτήσεων	58
ΚΕΦΑΛΑΙΟ 7: ΣΥΜΠΕΡΑΣΜΑΤΑ.....	67
Προτάσεις για το μέλλον.....	68
ΒΙΒΛΙΟΓΡΑΦΙΑ	69
ΠΑΡΑΡΤΗΜΑ.....	71
1. Κώδικας GAMS με 3 αντικειμενικές συναρτήσεις	71
2. Κώδικας GAMS με 2 αντικειμενικές συναρτήσεις	111

Λίστα Εικόνων

Εικόνα 1: Ταξινόμηση μεθόδων Πολυκριτηριακής Λήψης Αποφάσεων

Εικόνα 2 Θερμικές απώλειες κτιριακού κελύφους

Εικόνα 3 Θερμικές απώλειες στο εξωτερικό τοίχωμα

Εικόνα 4 Θερμικές απώλειες στα παράθυρα

Εικόνα 5 Κάτοψη του κτιρίου

Εικόνα 6 Διάγραμμα Pareto με τις 3 αντικειμενικές συναρτήσεις

Εικόνα 7 Διάγραμμα Pareto - Συνολικό ετήσιο κόστος - Συνολική κατανάλωση πρωτογενούς ενέργειας

Λίστα Πινάκων

- Πίνακας 1** Χαρακτηριστικά κτιρίου
- Πίνακας 2** Κλιματολογικά χαρακτηριστικά της Αθήνας
- Πίνακας 3** Χαρακτηριστικά διαθέσιμων πορτών
- Πίνακας 4** Χαρακτηριστικά διαθέσιμων παραθύρων
- Πίνακας 5** Χαρακτηριστικά διαθέσιμων τοίχων
- Πίνακας 6** Χαρακτηριστικά διαθέσιμων δαπέδων
- Πίνακας 7** Χαρακτηριστικά διαθέσιμων οροφών
- Πίνακας 8** Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης
- Πίνακας 9** Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων ψύξης
- Πίνακας 10** Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης-ψύξης
- Πίνακας 11** Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης-ZNX
- Πίνακας 12** Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων ZNX
- Πίνακας 13** Χαρακτηριστικά διαθέσιμων ηλιακών θερμοσίφωνων
- Πίνακας 14** Χαρακτηριστικά διαθέσιμων φωτοβολταϊκών
- Πίνακας 15** Χαρακτηριστικά διαθέσιμων λαμπτήρων
- Πίνακας 16** Χαρακτηριστικά διαθέσιμων τηλεοράσεων
- Πίνακας 17** Χαρακτηριστικά διαθέσιμων ηλεκτρικών κουζινών
- Πίνακας 18** Χαρακτηριστικά διαθέσιμων πλυντηρίων
- Πίνακας 19** Χαρακτηριστικά διαθέσιμων ψυγείων
- Πίνακας 20** Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων θέρμανσης
- Πίνακας 21** Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων θέρμανσης-ZNX
- Πίνακας 22** Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων ZNX
- Πίνακας 23** Αποτελέσματα προβλήματος με 3 αντικειμενικές
- Πίνακας 24** Αποτελέσματα προβλήματος με 3 αντικειμενικές
- Πίνακας 25** Αποτελέσματα προβλήματος με 3 αντικειμενικές
- Πίνακας 26** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 27** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 28** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 29** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 30** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 31** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές
- Πίνακας 32** Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

Πίνακας 33 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

Πίνακας 34 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

Πίνακας 35 Payoff Table- Πίνακας Πληρωμών προβλήματος με 3 αντικειμενικές

Πίνακας 36 Αποτελέσματα προβλήματος με 2 αντικειμενικές

Πίνακας 37 Αποτελέσματα προβλήματος με 2 αντικειμενικές

Πίνακας 38 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 39 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 40 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 41 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 42 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 43 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 44 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 45 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Πίνακας 46 Payoff Table- Πίνακας Πληρωμών προβλήματος με 2 αντικειμενικές

ΚΕΦΑΛΑΙΟ 1 : ΕΙΣΑΓΩΓΗ

Η αύξηση της κατανάλωσης πρωτογενούς ενέργειας καθώς και των εκπομπών διοξειδίου του άνθρακα έχουν οδηγήσει την παγκόσμια επιστημονική κοινότητα στην συνεχή αναζήτηση νέων μεθόδων για την μείωση τους. Το πρόβλημα που μελετάται στην παρούσα εργασία είναι ο ενεργειακός σχεδιασμός κτιρίων με χρήση Πολυκριτηριακού Μαθηματικού Προγραμματισμού.

Το μοντέλο σε αυτήν την εργασία βασίζεται σε ένα υπάρχων μοντέλο, αλλά έχει τις εξής τροποποιήσεις και προσθήκες. Πρώτον έχει γραμμικοποιηθεί, έχει εισαχθεί μια ακόμα αντικειμενική συνάρτηση, αυτή των εκπομπής διοξειδίου του άνθρακα, έχει μοντελοποιηθεί σε γλώσσα GAMS και έχει επιλυθεί με την μέθοδο των περιορισμών (e-constraint). Σκοπός του μοντέλου είναι να επιλέξει τα κατάλληλα συστήματα και μέρη του κτιρίου που ελαχιστοποιούν την κατανάλωση πρωτογενούς ενέργειας και των ρύπων, καθώς και το συνολικό ετήσιο κόστος. Το μοντέλο θα υπολογίσει τις τιμές των μεταβλητών απόφασης που ικανοποιούν τους περιορισμούς του προβλήματος και θα δώσει τις κατά Pareto βέλτιστες λύσεις, από τις οποίες ο αποφασίζων μπορεί να επιλέξει αυτή που ικανοποιεί τα κριτήρια του.

Μελετώνται 2 περιπτώσεις του προβλήματος, μία με σκοπό την ελαχιστοποίηση του συνολικού ετήσιου κόστους, της κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα, και μία με σκοπό την ελαχιστοποίηση του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας.

Στο 2ο κεφάλαιο γίνεται μια βιβλιογραφική ανασκόπηση. Στο 3ο κεφάλαιο γίνεται μια εισαγωγή στον μαθηματικό προγραμματισμό, στις υποκατηγορίες τους, στις μεθόδους του καθώς και στις εφαρμογές του. Στο 4ο κεφάλαιο γίνεται μια εισαγωγή στον ενεργειακό σχεδιασμό κτιρίων, στα μέρη ενός κτιρίων, στις απώλειες θερμότητας και στα διάφορα ενεργειακά συστήματα. Στο 5ο κεφάλαιο παρουσιάζεται το μοντέλο πολυκριτηριακού προγραμματισμού που βασίστηκε η εργασία. Παρουσιάζονται αναλυτικά όλες οι απαραίτητες μεταβλητές εξισώσεις και περιορισμοί και ορίζονται οι αντικειμενικές συναρτήσεις. Στο 6ο κεφάλαιο παρουσιάζονται όλες οι παράμετροι του κτιρίου που μελετάτε, των συστημάτων που διατίθενται για επιλογή, καθώς και τα αποτελέσματα με σχόλια. Στο 7ο κεφάλαιο παρουσιάζονται τα συμπεράσματα της εφαρμογής του μοντέλου για 2 περιπτώσεις, πρώτον της ελαχιστοποίησης των αντικειμενικών συναρτήσεων του συνολικού ετήσιου κόστους, της κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών

διοξειδίου του άνθρακα, και μία με της κατανάλωσης πρωτογενούς ενέργειας και του συνολικού ετήσιου κόστους. Επίσης αναφέρονται ιδέες για περαιτέρω έρευνα. Στο παράρτημα παρατίθενται ο κώδικας που χρησιμοποιήθηκε στις 2 περιπτώσεις σε γλώσσα GAMS.

ΚΕΦΑΛΑΙΟ 2: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Ο υπολογισμός των ενεργειακών φορτίων των κτιρίων και η βελτίωση της ενεργειακής τους απόδοση έχει ερευνηθεί εκτενώς. Αυτή η εργασία βασίζεται στην δημοσίευση των Karmellos et al [1], οι οποίοι έχουν αναπτύξει ένα μοντέλο πολυκριτηριακού μαθηματικού προγραμματισμού που επιλύεται με την μέθοδο των συντελεστών στάθμισης, με σκοπό την επιλογή των κατάλληλων μερών του κτιριακού κελύφους και ενεργειακών συστημάτων που θα ελαχιστοποιήσουν το κόστος και την κατανάλωση πρωτογενούς ενέργειας. Αυτή η δημοσίευση είναι μια εξέλιξη της δημοσίευσης των Diakaki et al [2], οι οποίοι έχουν και την αντικειμενική συνάρτηση των εκπομπών διοξειδίου το άνθρακα. Οι Asadi et al [3] έχουν αναπτύξει ένα μοντέλο πολυκριτηριακού προγραμματισμού χρησιμοποιώντας το κριτήριο Tchebychef για την ανακαίνιση των κτιρίων. Επιλέγει παράθυρα, μονώση για τους τοίχους και τις οροφές και ηλιακό συλλέκτη, και έχει ως αντικειμενικές συναρτήσεις την εξοικονόμηση ενέργειας και το κόστος ανακαίνισης. Οι Kolokotsa et al [4] έχουν ασχοληθεί με κτίρια μηδενικής ενέργειας. Μελετούν επίσης και την περίπτωση κτιρίων θετικής ενέργειας και των συστημάτων αυτοματισμού που είναι υπεύθυνα για τον βέλτιστο έλεγχο των συστημάτων παραγωγής ενέργειας με ανανεώσιμες πηγές. Οι Mahlasi et al [5] έχουν αναπτύξει ένα μοντέλο για την ανακαίνιση κτιρίων, με σκοπό την μεγιστοποίηση της εξοικονόμησης ενέργειας και την ελαχιστοποίηση του χρόνου απόσβεσης, με τη χρήση γενετικού αλγορίθμου. Οι Mavrotas et al [6] έχουν αναπτύξει ένα μοντέλο ενεργειακού σχεδιασμού, με σκοπό την μεγιστοποίηση της ικανοποίησης της ζήτησης σε συνθήκες αβεβαιότητας της ενεργειακής ζήτησης. Η μελέτη περίπτωσης αφορά ένα νοσοκομείο στην Αθήνα και καταλήξανε στο συμπέρασμα ότι η εγκατάσταση συστήματος συμπαραγωγής ενέργειας είναι συμφέρουσα στις περισσότερες περιπτώσεις. Επίσης οι Mavrotas et al [7] έχουν μελετήσει το παραπάνω πρόβλημα με προσομοίωση Monte Carlo, που εξάγει πιο ακριβείς λύσεις δεδομένης της αβεβαιότητας της τιμής των καυσίμων.

ΚΕΦΑΛΑΙΟ 3: ΜΑΘΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

3.1 Εισαγωγή στον Μαθηματικό Προγραμματισμό

Ο Μαθηματικός Προγραμματισμός είναι ο κλάδος των μαθηματικών που ασχολείται με την **μοντελοποίηση** και **βελτιστοποίηση** συστημάτων. Για να είναι εφικτό αυτό πρέπει το σύστημα να είναι μοντελοποιημένο σε κατάλληλες μαθηματικές σχέσεις έτσι ώστε να είναι δυνατή η χρήση των μεθόδων του Μαθηματικού Προγραμματισμού.

Τα βασικά δομικά στοιχεία ενός προβλήματος Μαθηματικού Προγραμματισμού είναι τα εξής [8]:

Μεταβλητές απόφασης: Εκφράζουν τους αγνώστους του προβλήματος και είναι οι μεταβλητές που ο αποφασίζων μπορεί να καθορίσει. Σκοπός της βελτιστοποίησης είναι η εύρεση των τιμών των μεταβλητών απόφασης που βελτιστοποιούνε την αντικειμενική συνάρτηση.

Περιορισμοί: Είναι οι μαθηματικές σχέσεις, ισότητες ή ανισότητες, που καθορίζουνε τις τιμές που επιτρέπεται να πάρουνε οι μεταβλητές απόφασης, ορίζουν δηλαδή το πεδίο ορισμού του προβλήματος.

Αντικειμενική συνάρτηση: Είναι η μαθηματική σχέση των μεταβλητών απόφασης που επιδιώκεται να βελτιστοποιηθεί, δηλαδή να μεγιστοποιηθεί ή να ελαχιστοποιηθεί. Ένα πρόβλημα μπορεί να περιέχει πάνω από μία αντικειμενικές συναρτήσεις.

Παράμετροι: Είναι οι όροι του προβλήματος που δεν καθορίζονται από τον αποφασίζοντα και έχουν σταθερή τιμή σε όλη την διαδικασία επίλυσης. Συνήθως είναι οι συντελεστές των μεταβλητών απόφασης.

Τα προβλήματα Μαθηματικού Προγραμματισμού μπορούν να ταξινομηθούν ανάλογα με το είδος των μεταβλητών απόφασης, το είδος των παραμέτρων, το πλήθος των αντικειμενικών συναρτήσεων και το είδος των μαθηματικών σχέσεων του προβλήματος.

Όσον αφορά το είδος των αντικειμενικών συναρτήσεων και των περιορισμών το πρόβλημα χαρακτηρίζεται **Γραμμικού Προγραμματισμού** (Linear Programming) όταν οι αντικειμενικές συναρτήσεις και οι περιορισμοί είναι γραμμικές ως προς τις μεταβλητές απόφασης. Αν υπάρχει τουλάχιστον ένας μη γραμμικός περιορισμός ή αντικειμενική συνάρτηση τότε το πρόβλημα χαρακτηρίζεται **Μη Γραμμικού Προγραμματισμού** (Non Linear Programming). Τα

προβλήματα Γραμμικού Προγραμματισμού μπορούν να λυθούν σχετικά εύκολα με την μέθοδο Simplex και τις παραλλαγές της. Με αυτή την μέθοδο είναι δυνατή η επίλυση προβλημάτων με μεγάλο αριθμό μεταβλητών απόφασης σε πολύ μικρό χρονικό διάστημα. Τα γραμμικά προβλήματα οδηγούν σε ολικά βέλτιστα, σε αντίθεση με τα μη γραμμικά που οδηγούν σε τοπικά βέλτιστα. Για αυτό το λόγο τα μη γραμμικά προβλήματα όταν κρίνεται απαραίτητο γραμμικοποιούνται με τις κατάλληλες παραδοχές και μεθόδους ώστε να είναι πιο απλή η διαδικασία επίλυσης.

Όσον αφορά το είδος των μεταβλητών απόφασης τα προβλήματα ταξινομούνται σε **Ακέραιου Προγραμματισμού** αν οι μεταβλητές είναι ακέραιες, ενώ αν έχουν και συνεχείς μεταβλητές ταξινομούνται σε **Μικτού Ακέραιου Προγραμματισμού**. Στις περισσότερες περιπτώσεις οι ακέραιες μεταβλητές είναι δυαδικές, δηλαδή παίρνουν τιμές 0 ή 1.

Τέλος, αν υπάρχουν περισσότερες από μία αντικειμενικές συναρτήσεις το πρόβλημα χαρακτηρίζεται **Πολυκριτηριακού Μαθηματικού Προγραμματισμού**.

3.2 Εφαρμογές Γραμμικού Προγραμματισμού

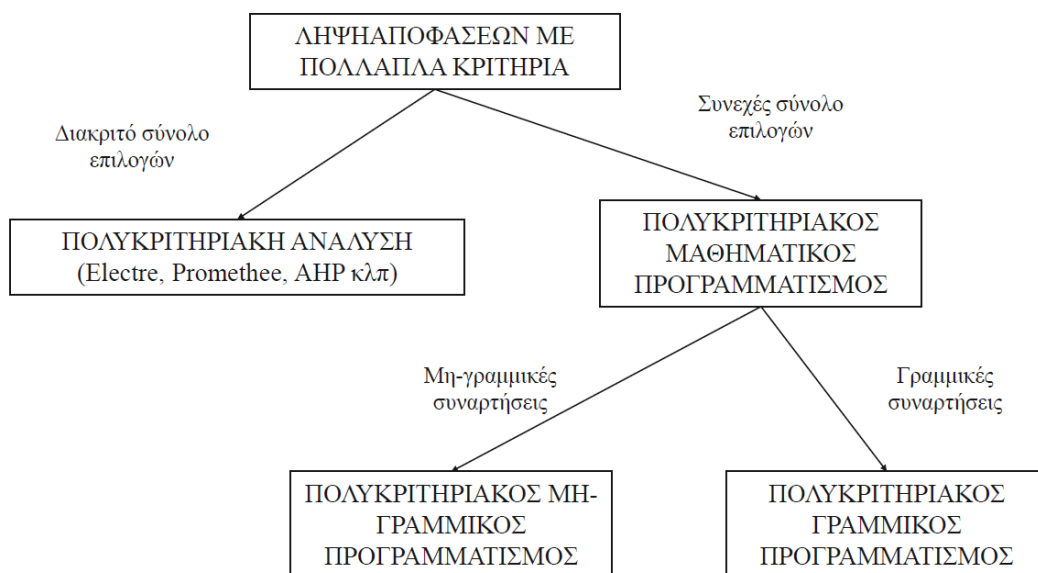
Οι εφαρμογές του Γραμμικού Προγραμματισμού είναι πάρα πολλές και σε πολλούς τομείς. Η πρώτη εφαρμογή του Μαθηματικού Προγραμματισμού ήταν στον Β΄ Παγκόσμιο Πόλεμο, κατά την διάρκεια του οποίου οι Βρετανοί αντιμετώπισαν πρόβλημα με την κατανομή των πόρων τους στα διάφορα στρατόπεδα [9]. Για αυτό και ο όρος Επιχειρησιακή Έρευνα προέρχεται από την στρατιωτική επιχείρηση και όχι από την επιχείρηση-βιομηχανία. Μετά η χρήση του εξαπλώθηκε και σε βιομηχανίες πετρελαίου, καθώς μια βιομηχανία πετρελαίου έχει την επιλογή να αγοράσει πετρέλαιο από διάφορες πηγές με διαφορετικές συστάσεις σε διαφορετικές τιμές και να παρασκευάσει πολλά προϊόντα με διαφορετικές προδιαγραφές. Οι περιορισμοί ενός τέτοιου προβλήματος μπορεί να είναι στην ποσότητα της πρώτης ύλης που μπορεί να παρασκευάσει και την ποσότητα που μπορεί να αγοράσει από μια πηγή. Επίσης με την χρήση του ΓΠ είναι δυνατός ο σχεδιασμός ενός πλάνου της παραγωγής. Δεδομένου ότι οι πωλήσεις μιας βιομηχανίας μεταβάλλονται, η βιομηχανία έχει διάφορες επιλογές όπως να προγραμματίσει υπερωρίες για να ικανοποιήσει την ζήτηση. Σε βιομηχανίες τροφίμων είναι δυνατός ο σχεδιασμός ενός βέλτιστου σχεδίου διανομής των προϊόντων από τις βιομηχανίες στις διάφορες αποθήκες. Οι μεταλλουργικές βιομηχανίες χρησιμοποιούν τον ΓΠ για να επιλέξουν αν συμφέρει η αγορά ή η κατασκευή ενός

προϊόντος.

3.3 Πολυκριτηριακός Προγραμματισμός

Ο Πολυκριτηριακός Μαθηματικός Προγραμματισμός (ΠΜΠ) αποτελεί την γενίκευση του απλού Γραμμικού Προγραμματισμού και χαρακτηρίζεται από την ύπαρξη πάνω από μίας αντικειμενικής συνάρτησης. Αναπτύχθηκε διότι η χρήση πάνω από μίας αντικειμενικής συνάρτησης αποτελεί μια πιο ρεαλιστική προσέγγιση μοντελοποίησης ρεαλιστικών προβλημάτων. Τα προβλήματα ΠΜΠ ανήκουν στην κατηγορία των προβλημάτων χαμηλού βαθμού δόμησης (ill structured problems), είναι δηλαδή προβλήματα όπου η ορθολογική λύση δεν καθορίζεται από το ίδιο το πρόβλημα αλλά αποτελεί αντικείμενο προοδευτικής αναζήτησης με εμπλοκή του αποφασίζοντα στη διαδικασία αυτή [10]. Στο Σχήμα 1 παρουσιάζεται η ταξινόμηση των μεθόδων ΠΜΠ.

Στον ΠΜΠ δεν υπάρχει η έννοια της βέλτιστης λύσης διότι δεν υπάρχει λύση που να βελτιστοποιεί ταυτόχρονα όλες τις αντικειμενικές συναρτήσεις. Στον ΠΜΠ υπάρχει η έννοια των κατά Pareto άριστων λύσεων. Η διαδικασία βελτιστοποίησης ενός συνόλου αντικειμενικών συναρτήσεων ονομάζεται πολυκριτηριακή ή διανυσματική βελτιστοποίηση.



Εικόνα 1: Ταξινόμηση μεθόδων Πολυκριτηριακής Λήψης Αποφάσεων

Ας θεωρήσουμε το ακόλουθο πρόβλημα ΠΜΠ με p αντικειμενικές συναρτήσεις, n μεταβλητές απόφασης και m περιορισμούς:

$$\max f_1(x)=z_1$$

$$\max f_2(x)=z_2$$

...

$$\max f_p(x)=z_p$$

s.t

$$x \in S$$

όπου S είναι το εφικτό χωρίο των περιορισμών που καθορίζεται από τους m περιορισμούς, x είναι το διάνυσμα των n μεταβλητών απόφασης και f_1, f_2, \dots, f_p οι p αντικειμενικές συναρτήσεις.

Ορισμός: Μία λύση x' του παραπάνω προβλήματος λέγεται κατά Pareto άριστη λύση (αν και μόνο αν $x' \in S$ και δεν υπάρχει άλλη λύση $x \in S$ τέτοια ώστε $f_i(x) \geq f_i(x')$ για κάθε $i=1,2,\dots,p$ και $f_i(x) > f_i(x')$ για τουλάχιστον ένα i).

Δηλαδή μια λύση Pareto είναι η λύση η οποία αντιστοιχεί σε ένα μη βελτιώσιμο διάνυσμα στο χώρο των αντικειμενικών συναρτήσεων. Με απλά λόγια αυτό σημαίνει ότι δεν μπορούμε να βελτιώσουμε την τιμή μιας αντικειμενικής συνάρτησης χωρίς να χειροτερέψουμε την τιμή τουλάχιστον μίας από τις άλλες. Η επίλυση του προβλήματος είναι ουσιαστικά η εύρεση εκείνης της κατά Pareto λύσης που ικανοποιεί περισσότερο τον αποφασίζοντα.

Τελική ή σχετικά βέλτιστη λύση (final or best compromise solution): Η ικανή εκείνη λύση που τελικά επιλέγει από το σύνολο των ικανών λύσεων ο αποφασίζων, λέγεται τελική ή σχετικά βέλτιστη λύση. Ο όρος «σχετικά» σημαίνει ακριβώς ότι αποτελεί υποκειμενική επιλογή του αποφασίζοντα σε αντίθεση με τη βέλτιστη λύση σε ένα πρόβλημα ΓΠ που είναι αντικειμενικά προσδιοριζόμενη.

Πίνακας τιμών (payoff table): Είναι ο τετραγωνικός πίνακας που αποτελείται από τόσες γραμμές και στήλες όσες και οι αντικειμενικές συναρτήσεις του προβλήματος ΠΜΠ. Σε κάθε στήλη υπάρχει η βέλτιστη τιμή μιας αντικειμενικής συνάρτησης και οι τιμές των υπόλοιπων αντικειμενικών συναρτήσεων που προκύπτουν.

3.4 Μέθοδοι του Πολυκριτηριακού Μαθηματικού Προγραμματισμού

Οι μέθοδοι του ΠΜΠ ταξινομούνται ανάλογα σε ποιο στάδιο επεμβαίνει ο αποφασίζων. Αν δηλαδή επεμβαίνει πριν την επίλυση (μέθοδοι a priori, π.χ. προγραμματισμός στόχων, goal programming), κατά την διάρκεια (αλληλεπιδραστικές μέθοδοι, interactive methods) ή μετά την επίλυση (μέθοδοι παραγωγής, generation methods).

Στην απλή περίπτωση προβλημάτων με μία αντικειμενική συνάρτηση ο αποφασίζων επεμβαίνει μόνο στην μορφοποίηση του προβλήματος. Αφού ορίσει την αντικειμενική συνάρτηση και την κατεύθυνση βελτιστοποίησης, δεν χρειάζεται να επέμβει περαιτέρω. Αυτό συμβαίνει διότι τα προβλήματα με ένα μόνο κριτήριο (αντικειμενική συνάρτηση) είναι ουσιαστικά προβλήματα εύρεσης της βέλτιστης λύσης και όχι προβλήματα λήψης αποφάσεων. Στα πολυκριτηριακά προβλήματα όμως ο αποφασίζων πρέπει να επέμβει στην διαδικασία επίλυσης για να εισάγει τις προτιμήσεις του.

Στην πρώτη περίπτωση μεθόδων a priori, ο αποφασίζων μπορεί να αποδώσει σε κάθε αντικειμενική συνάρτηση έναν συντελεστή βαρύτητας στο διάστημα $[0,1]$ και στη συνέχεια να εισάγει μια τελική αντικειμενική συνάρτηση που περιέχει όλες τις αντικειμενικές συναρτήσεις, συνήθως με γραμμικό άθροισμα αυτών ώστε να μην αλλοιώνεται η γραμμικότητα του προβλήματος. Η άριστη λύση της παραγόμενης αντικειμενικής του γραμμικού προβλήματος είναι και ικανή λύση για το αρχικό πρόβλημα ΠΜΠ όταν όλοι οι συντελεστές βαρύτητας είναι διάφοροι του μηδενός.

Στην δεύτερη περίπτωση καθορίζονται a priori οι τιμές-στόχοι των αντικειμενικών συναρτήσεων. Αυτό λέγεται προγραμματισμός στόχων (Goal Programming). Σε αυτή την μέθοδο ο αποφασίζων θέτει ως στόχους τις τιμές που επιθυμεί να πάρει κάθε μια από τις αντικειμενικές συναρτήσεις και προσπαθεί να ελαχιστοποιήσει την απόκλιση της κάθε αντικειμενικής συνάρτησης από την καθορισμένη τιμή – στόχο.

Το βασικό πλεονέκτημα αυτής της κατηγορίας τεχνικών είναι ότι η διαδικασία επίλυσης είναι απλή και γρήγορη, καθότι υπάρχουν πολλές επιλογές λογισμικού για γραμμικό προγραμματισμό.

Όταν ο αποφασίζων επεμβαίνει κατά την διάρκεια της επίλυσης οι μέθοδοι που χρησιμοποιούνται ονομάζονται αλληλεπιδραστικές. Σε αυτές τις μεθόδους ο αποφασίζων βρίσκεται σε "διάλογο" (αλληλεπίδραση) με το πρόγραμμα και αυτή η αλληλεπίδραση συνεχίζεται μέχρι να ικανοποιηθεί το κριτήριο σύγκλισης που έχει ορισθεί και να βρεθεί η

τελική λύση. Πρόκειται για μία επαναληπτική διαδικασία όπου σε κάθε επανάληψη προκύπτουν κάποιες λύσεις. Ο αποφασίζων επιλέγει μια και με αυτό τον τρόπο εισάγει άμεσα τις προτιμήσεις του κατευθύνοντας την διαδικασία επίλυσης μέχρι την τελική λύση.

Τα βασικά πλεονεκτήματα των αλληλεπιδραστικών μεθόδων είναι τα ακόλουθα:

- 1) Δεν απαιτείται η χρήση εξειδικευμένου λογισμικού καθότι χρησιμοποιούν ρουτίνες γραμμικού προγραμματισμού.
- 2) Μπορούν να χρησιμοποιηθούν για την λύση μεγάλων προβλημάτων
- 3) Η επαναληπτική τους φύση επιτρέπει στον αποφασίζοντα να κατανοήσει βαθύτερα το πρόβλημα.

Τέλος όταν αποφασίζων εισάγει τις προτιμήσεις του μετά την επίλυση του προβλήματος οι μέθοδοι που χρησιμοποιούνται ονομάζονται μέθοδοι παραγωγής (generation methods). Οι μέθοδοι αυτοί παράγουν τις ικανές λύσεις ή ένα υποσύνολο αυτών. Μετά την επίλυση του προβλήματος, ο αποφασίζων εξετάζει όλες τις ικανές λύσεις και επιλέγει την προτιμότερη ως τελική. Οι μέθοδοι αυτές παρουσιάζουν όλες τις δυνατές επιλογές στον αποφασίζοντα πριν επιλέξει την τελική λύση. Οι μέθοδοι παραγωγής απαιτούν μεγάλη υπολογιστική ισχύ, απαιτούν εξειδικευμένο λογισμικό. Σε προβλήματα με πολλές μεταβλητές και περιορισμούς ο αριθμός των ικανών λύσεων είναι υπερβολικά μεγάλος που ο υπολογισμός του καθίσταται υπολογιστικά ανέφικτος, όμως με την εξέλιξη των υπολογιστών όλο και μεγαλύτερα προβλήματα μπορούν να επιλυθούν.

Οι μέθοδοι παραγωγής χωρίζονται σε 2 κατηγορίες : αυτές που παράγουν τις ακραίες ικανές λύσεις και αυτές που παράγουν και μη ακραίες ικανές λύσεις. Η τελική λύση συνήθως είναι από το σύνολο των ακραίων λύσεων, αν και από την προτιμότερη ακραία ικανή λύση και τις γειτονικές της μπορούν να προκύψουν και οι ενδιάμεσες λύσεις τις οποίες μπορεί να εξετάσει ο αποφασίζων.

Η πιο απλή μέθοδος παραγωγής των ακραίων λύσεων είναι μέσω του σταθμισμένου αθροίσματος των αντικειμενικών συναρτήσεων και ονομάζεται μέθοδος των συντελεστών στάθμισης (weighting method). Με την μεταβολή των συντελεστών στάθμισης σαρώνεται όλο το φάσμα των ικανών λύσεων. Απαιτεί έναν απλό επιλύτη γραμμικού προγραμματισμού. Μειονέκτημά της είναι ότι σε προβλήματα με περισσότερες από μία αντικειμενικές συναρτήσεις η εξέταση των συνδυασμών των συντελεστών στάθμισης είναι χρονοβόρα.

Μία άλλη μέθοδος παραγωγής ικανών λύσεων είναι η μέθοδος των περιορισμών (e-constraint method). Στην μέθοδο αυτή επιλέγεται μια αντικειμενική συνάρτηση και οι

υπόλοιπες εισάγονται ως περιορισμοί. Οι περιορισμοί είναι <<μεγαλύτερο ή ίσο>> αν η αντικειμενική έχει κατεύθυνση μεγιστοποίησης και <<μικρότερο ή ίσο>> αν έχει κατεύθυνση ελαχιστοποίησης. Η άριστη λύση του προβλήματος προκύπτει μόνο αν όλοι οι περιορισμοί ικανοποιούνται σαν ισότητες. Αν δεν ικανοποιούνται και προκύπτουν εναλλακτικά άριστα, τότε μπορεί η άριστη λύση που θα βρεθεί να μην ικανοποιεί το πρόβλημα του ΠΜΠ. Σε αυτήν την περίπτωση ο πίνακας πληρωμών (payoff table) του προβλήματος ΠΜΠ μπορεί να βοηθήσει στον καθορισμό της συστηματικής μεταβολής του δεξιού σκέλους των περιορισμών των αντικειμενικών συναρτήσεων. Για την παραγωγή του συνόλου των ικανών λύσεων πρέπει η συστηματική μεταβολή να είναι μικρή για να είναι πιο πυκνό το σύνολο των λύσεων. Αν είναι μεγάλη η μεταβολή η διαδικασία θα είναι συντομότερη αλλά το σύνολο των λύσεων θα είναι πιο αραιό. Για προβλήματα με περισσότερες από 2 αντικειμενικές η μέθοδος αυτή είναι δυσεφάρμοστη.

Μια τροποποιημένη μορφή της μεθόδου των περιορισμών είναι η επαυξημένη μέθοδος των περιορισμών (augmented ϵ -constraint) γνωστή και ως AUGMECON [10]. Η μέθοδος παρουσιάζεται παρακάτω :

$$\begin{aligned} & \max((f_1(x) + eps \times (S_2 / r_2 + S_3 / r_3 + \dots + S_p / r_p)) \\ & st \\ & f_2(x) - S_2 = e_2 \\ & f_3(x) - S_3 = e_3 \\ & \dots \\ & f_p(x) - S_p = e_p \\ & x \in S \\ & S_i \in R^+ \end{aligned}$$

όπου eps είναι ένας μικρός αριθμός.

Η εξέλιξη αυτής της μεθόδου οδήγησε στην AUGMECON2. Η AUGMECON2 έχει μια μικρή τροποποίηση στην αντικειμενική συνάρτηση [11]. Η μέθοδος παρουσιάζεται παρακάτω :

$$\max(f_1(x) + eps \times (S_2 / r_2 + 10^{-1} \times S_3 / r_3 + \dots + 10^{-(p-2)} \times S_p / r_p))$$

ΚΕΦΑΛΑΙΟ 4: ΕΝΕΡΓΕΙΑΚΟΣ ΣΧΕΔΙΑΣΜΟΣ ΚΤΙΡΙΩΝ

4.1 Ενεργειακή κατανάλωση Κτιρίων

Η αύξηση της πρωτογενούς κατανάλωσης ενέργειας και η κλιματική αλλαγή είναι μία από τις μεγαλύτερες προκλήσεις του 21ου αιώνα. Στις περισσότερες χώρες, οι κυβερνήσεις έχουν υιοθετήσει πολιτικές που αποσκοπούν στη μείωση της πρωτογενούς κατανάλωσης ενέργειας με την προώθηση της ενεργειακής απόδοσης. Συγκεκριμένα, ο κτιριακός τομέας αντιπροσωπεύει περίπου το 40% της τελικής κατανάλωσης ενέργειας στην Ευρωπαϊκή Ένωση και περίπου το 40% των της πρωτογενούς κατανάλωσης ενέργειας στον κόσμο [1]. Η μείωση της κατανάλωσης ενέργειας και ιδιαίτερα της πρωτογενούς θα συμβάλει στη μείωση της ενέργειας στην συνολική ενεργειακή αλυσίδα και την αύξηση της αειφορίας των κτιρίων. Η επένδυση στην ενεργειακή απόδοση είναι απαραίτητη καθώς τα οικονομικά και περιβαλλοντικά θα υπερκαλύψουν το αρχικό κόστος επένδυσης. Ο κτιριακός τομέας είναι μεγάλος, τόσο όσον αφορά την κατανάλωση ενέργειας, αλλά όσον αφορά τον αριθμό και το είδος των κτιρίων. Προκειμένου να μειωθεί η κατανάλωση πρωτογενούς ενέργειας στα κτίρια, πρέπει να εφαρμοστούν διάφορα μέτρα για την αύξηση της ενεργειακής τους απόδοσης. Τα μέτρα αυτά μπορούν να χωρισθούν σε κατηγορίες, όπως εκείνα που σχετίζονται με το κτιριακό κέλυφος, τα ενεργειακά συστήματα που παρέχουν θέρμανση, ψύξη και ζεστό νερό, οι ηλεκτρικές συσκευές και τα συστήματα φωτισμού. Επίσης, υπάρχουν ενεργειακά συστήματα που μπορούν να παράγουν ηλεκτρική ενέργεια όπως οι μονάδες συμπαραγωγής ή ανανεώσιμες πηγές ενέργειας (ΑΠΕ), όπως η βιομάζα, η αιολική ενέργεια και η ηλιακή ενέργεια. Τα μέτρα για την βελτίωση της ενεργειακής απόδοσης σε κάθε κατηγορία έχουν διαφορετική συμβολή στη μείωση της πρωτογενούς κατανάλωσης ενέργειας αλλά έχουν ένα αρχικό κόστος επένδυσης, συνήθως υψηλότερο από τα συμβατικά συστήματα. Επιπλέον, η τοποθεσία του κτιρίου παίζει σημαντικό ρόλο, στην κατανάλωση ενέργειας, καθώς σε κάθε τοποθεσία επικρατούν διαφορετικές καιρικές συνθήκες.

4.2 Βασικά συστήματα ενός κτιρίου

Η ενεργειακή κατανάλωση ενός κτιρίου καθορίζεται από τέσσερις βασικούς παράγοντες [12]:

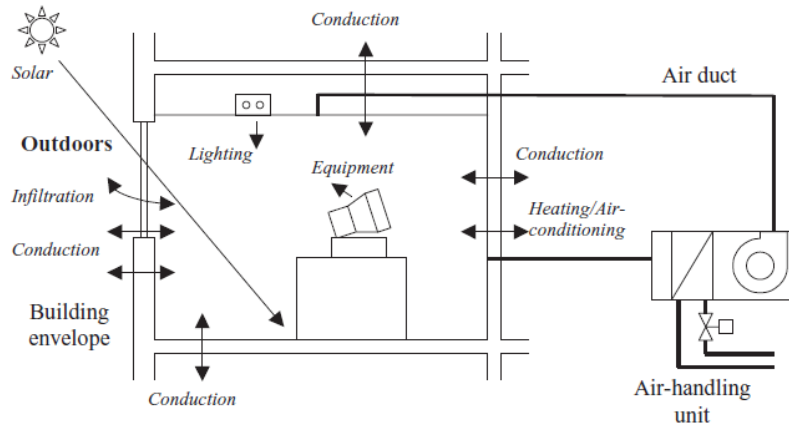
Τα μη ενεργειακά συστήματα του κτιρίου, δηλαδή το κέλυφος και το εσωτερικό του κτιρίου (τοιχοί, δάπεδα, οροφές, χωρίσματα παράθυρα, πόρτες κλπ).

- Τα συστήματα του περιβάλλοντος χώρου του κτιρίου όπως φύτευση, ένταξη υγρών στοιχείων, υπαίθρια σκίαστρα, εκμετάλλευση των καλοκαιρινών δροσερών ρευμάτων αέρα, προστασία από τους ψυχρούς χειμερινούς ανέμους.
- Στα ενεργειακά συστήματα του κτιρίου, δηλαδή τα συστήματα που απαιτούνται για τη θέρμανση, τον κλιματισμό, τον αερισμό, το φωτισμό κλπ.
- Τα ανθρώπινα συστήματα, που περιλαμβάνουν τους ανθρώπους που ζουν στο κτίριο.

4.2.1 Ενεργειακό Ισοζύγιο – Γενικά για Θερμικές Απώλειες Κτιρίων

Σε ένα κτίριο το ενεργειακό ισοζύγιο υπολογίζεται από τα θερμικά κέρδη και τις θερμικές απώλειες. Τα θερμικά κέρδη προέρχονται από τα συστήματα θέρμανσης του κτιρίου, την ηλιακή ακτινοβολία, καθώς και από το εσωτερικές πηγές θερμότητας όπως ο φωτισμός, ο άνθρωπος και οι ηλεκτρονικές συσκευές. Οι θερμικές απώλειες όπως φαίνονται στην Εικόνα 2 προέρχονται από [13]:

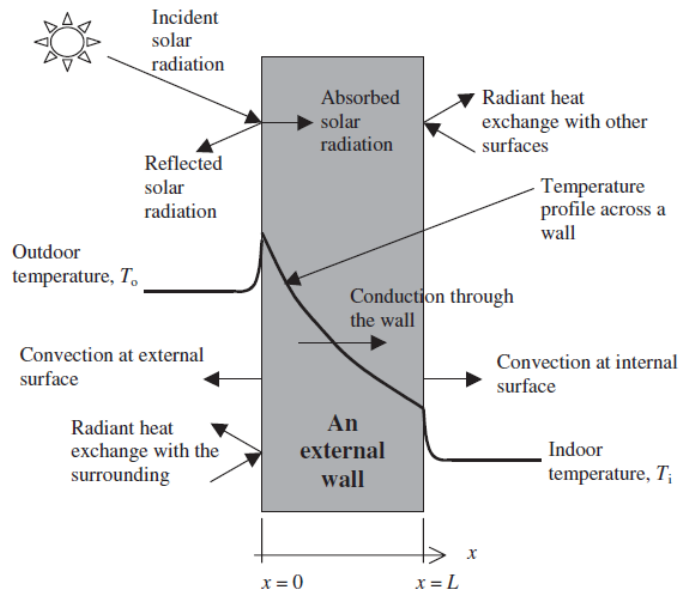
1. Ροή θερμότητας διαμέσου του κτιριακού κελύφους, δηλαδή των δομικών υλικών του κτιρίου, συμπεριλαμβανομένων και του δαπέδου και της οροφής
2. Ροή θερμότητας μέσω ηλιακής ακτινοβολίας από τα παράθυρα του κτιρίου και από τα ανοίγματα του κτιρίου
3. Διείσδυση εξωτερικού αέρα στο κτίριο



Εικόνα 2 Θερμικές απώλειες κτιριακού κελύφους [3]

4.2.2 Θερμικές Απώλειες σε Εξωτερικούς τοίχους –Παράθυρα-Οροφές

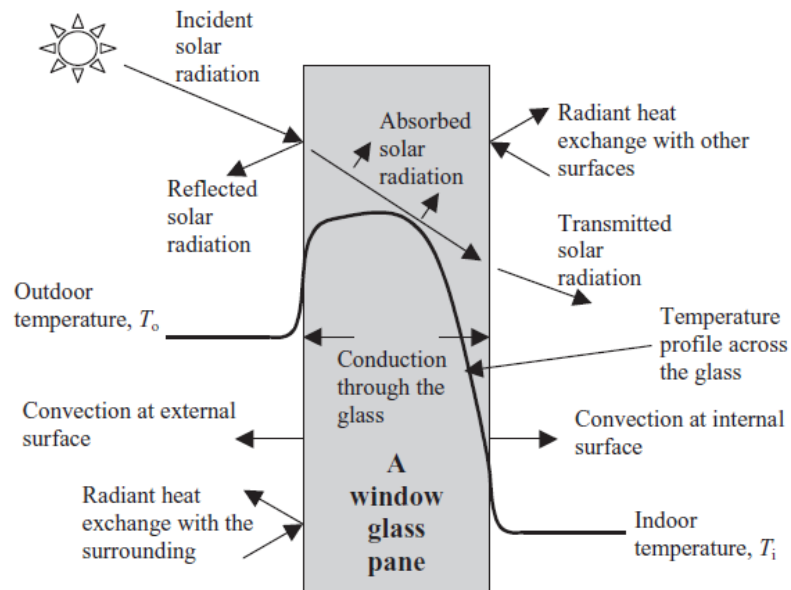
Στον εξωτερικό τοίχο του κτιρίου, πραγματοποιείται μεταφορά θερμότητας με συναγωγή μεταξύ της επιφάνειας και του αέρα που βρίσκεται σε σχετική κίνηση και ταυτόχρονα σε επαφή με αυτήν, καθώς επίσης και μεταφορά θερμότητας με ακτινοβολία από τις επιφάνειες στις οποίες προσπίπτει ηλιακή ακτινοβολία. Οι θερμικές απώλειες φαίνονται στην Εικόνα 3.



Εικόνα 3 Θερμικές απώλειες στο εξωτερικό τοίχωμα[13]

Επίσης έχουμε μεταφορά θερμότητας δια μέσου των παραθύρων. Μέρος της ηλιακής ακτινοβολίας, θα περάσει μέσα από το παράθυρο στον εσωτερικό χώρο του κτιρίου, όπως

φαίνεται στην Εικόνα 4. Καθώς η ηλιακή ακτινοβολία λοιπόν διεισδύει εντός του κτιρίου, μέρος αυτής θα απορροφηθεί από το υλικό του παραθύρου οδηγώντας σε αύξηση της θερμοκρασίας του, δημιουργώντας έτσι μεταφορά θερμότητας και προς τα μέσα και προς τα έξω, πρώτον με αγωγή εντός του τζαμιού και δευτερευόντως με συναγωγή και ακτινοβολία στις επιφάνειες και από τις δύο πλευρές του.



Εικόνα 4 Θερμικές απώλειες στα παράθυρα [13]

Κάθε παράθυρο χαρακτηρίζεται από τον δικό του συντελεστή ηλιακής θερμικής απολαβής (SHGC), ο οποίος ορίζεται από την παρακάτω εξίσωση.

$$SHGC = \frac{Q_{solar\ gain}}{Q_{incident}} : \text{όπου}$$

$Q_{solar\ gain}$: Κέρδος ηλιακής θερμότητας μέσα από το παράθυρο

$Q_{incident}$: Ηλιακή ακτινοβολία που προσπίπτει πάνω στο παράθυρο

Ο συντελεστής αυτός παίρνει τιμές από 0 έως 1, με την τιμή 1 να αντιστοιχεί σε μια τρύπα στον τοίχο, και την τιμή 0 να αντιστοιχεί σε απουσία παραθύρου.

Για να υπολογίσουμε τα πραγματικά ηλιακά κέρδη του κτιρίου από τα παράθυρα χρησιμοποιούμε την παρακάτω σχέση:

$$Q_{solar\ gain} = SHGC * A_{window} * Q_{incident}$$

Awindow: επιφάνεια παραθύρου

Ένας άλλος τρόπος χαρακτηρισμού των παραθύρων με βάση την συμπεριφορά τους στην ηλιακή ακτινοβολία είναι να το συγκρίνουμε με ένα παράθυρο που έχουμε θέσει ως σημείο αναφοράς. Ορίζουμε λοιπόν ένα παράθυρο με SHGC ίσο με 0,87 ως σημείο αναφοράς και μέσω αυτού ορίζουμε τον συντελεστή σκίασης (SC) ως τον λόγο του SHGC του εκάστοτε παραθύρου ως προς τον SHGC του παραθύρου αναφοράς. Δηλαδή ισχύει :

$$SC = \frac{SHGC}{SHGC_{ref}} = \frac{SHGC}{0.87}$$

Για αδιαφανείς επιφάνειες όπως οι τοίχοι και οι οροφές η επίδραση της ηλιακής ακτινοβολίας δεν μπορεί να αγνοηθεί. Η θερμοκρασία αυτών των επιφανειών είναι μεγαλύτερη από την θερμοκρασία του περιβάλλοντος. Για ορθότερο υπολογισμό των θερμικών φορτίων του κτιρίου, στις εξισώσεις μεταφοράς θερμότητας αντικαθιστούμε την εξωτερική θερμοκρασία με την T_{solair} . Η θερμοκρασία αυτή υπολογίζεται από την παρακάτω σχέση [14].

$$T_{solair} = T_{ambient} + \frac{a * Q_{solar}}{h_o} - \frac{\varepsilon * \sigma * (T_{ambient}^4 - T_{surface}^4)}{h_o}$$

$T_{ambient}$: θερμοκρασία περιβάλλοντος

a : συντελεστής ηλιακής απορροφητικότητάς

Q_{solar} : ηλιακή ακτινοβολία που προσπίπτει στην επιφάνεια

ε : συντελεστής εκπομπής

σ : σταθερά Stefan – Boltzmann

$T_{surface}$: θερμοκρασία επιφάνειας τοίχου

Υπό σταθερές συνθήκες, ο ρυθμός μεταφοράς θερμότητας οποιουδήποτε τμήματος του κτιρίου, είτε πρόκειται για τοίχο ή οροφή μπορεί να προσδιοριστεί από την εξίσωση

$$Q = U * A * (T_i - T_o)$$

U : ολικός συντελεστής μεταφοράς θερμότητας

A: επιφάνεια που λαμβάνει χώρα η μεταφορά της θερμότητας

T_i: εσωτερική θερμοκρασία

T_o: εξωτερική θερμοκρασία

Ο συντελεστής U έχει μονάδες W/m²*K, δηλαδή εκφράζει την θερμότητα που ρέει σε υλικό ανά μονάδα επιφάνειας και βαθμού K. Εκφράζει το πόσο αποδοτικό είναι το εκάστοτε μέρος του κτιρίου στην μετάδοση θερμότητας στο εσωτερικό του. Όσο μικρότερος είναι ο συντελεστής, τόσο λιγότερη θερμότητα εισέρχεται στο εσωτερικό του. Τυπικές τιμές του U κυμαίνονται από 0,18 W/m²*K για μονωμένο τοίχο μέχρι 2 W/m²*K για απλά τούβλα. Σε τοίχους που αποτελούνται από πολλά υλικά η συνολική τιμή του U δίνεται από την σχέση:

$$\frac{1}{U_{total}} = \frac{1}{U_1} + \frac{1}{U_2} + \dots + \frac{1}{U_n}$$

U₁, U₂... U_n ο συντελεστής μεταφοράς θερμότητας του κάθε υλικού

4.2.3 Θερμότητα παραγόμενη από τους ανθρώπους

Οι άνθρωποι αποτελούνται από τρισεκατομμύρια μικροσκοπικά κύτταρα, μέσα στα οποία λαμβάνουν χώρα διάφορες λειτουργίες απαραίτητες για την επιβίωση. Το σύνολο αυτών των λειτουργιών ονομάζεται **μεταβολισμός** [14]. Όλες αυτές οι λειτουργίες, ελευθερώνουν θερμότητα από τις καύσεις των διαφόρων τροφών. Ο ρυθμός μεταβολισμού σε κατάσταση ηρεμίας ονομάζεται βασικός μεταβολικός ρυθμός, και είναι ο ρυθμός του μεταβολισμού που απαιτείται για να κρατήσει ένα σώμα που εκτελεί τις βασικές σωματικές λειτουργίες όπως αναπνοή, κυκλοφορία του αίματος χωρίς καμία άλλη εξωτερική δραστηριότητα. Για ένα μέσο άνθρωπο 30 ετών, μάζας 70 κιλών, ύψους 1,73 m, ο βασικός μεταβολικός ρυθμός είναι περίπου 84W. Ο ρυθμός αυτός μπορεί να αυξηθεί μέχρι και 10 φορές, ανάλογα με το επίπεδο της σωματικής άσκησης.

4.2.4 Θερμότητα από τα συστήματα φωτισμού

Η ενέργεια που καταναλώνουν τα συστήματα φωτισμού αποτελεί περίπου το 7% της συνολικής ενέργειας στις κατοικίες και το 25% στα εμπορικά κτίρια [14]. Όλα τα σύγχρονα συστήματα φωτισμού χρησιμοποιούν ηλεκτρική ενέργεια. Το ποσό θερμότητας που

εκπέμπει ένα σύστημα φωτισμού εξαρτάται από το είδος του. Οι λάμπες πυρακτώσεως έχουν χαμηλή απόδοση, σε αντίθεση με τις λάμπες φθορισμού που έχουν υψηλή απόδοση αλλά και υψηλό κόστος. Μια κοινή λάμπα παράγει 5 έως 20W ενώ μια λάμπα φθορισμού 55 με 200W. Απορρίπτουνε στο περιβάλλον περίπου το 40% της ενέργειας αυτής.

4.2.5 Θερμότητα από τις ηλεκτρικές συσκευές

Οι περισσότερες ηλεκτρικές συσκευές περιέχουν ένα ηλεκτρικό κινητήρα, έτσι η θερμότητα που εκπέμπει ισοδυναμεί με την ενέργεια που καταναλώνει ο ηλεκτρικός κινητήρας. Η ονομαστική ισχύς του κινητήρα εκφράζει την θερμότητα που αποδίδει σε συνθήκες πλήρους φορτίου. Αλλά ένας κινητήρας λειτουργεί συνήθως σε συνθήκες μερικού φορτίου, περίπου 30% έως 40%. Έτσι ορίζουμε τον συντελεστή φορτίου (loading factor) ως τον λόγο της χρησιμοποιούμενης ισχύς ως προς την μέγιστη δυνατή ισχύ του κινητήρα, με την τιμή 1 να αντιστοιχεί σε συνθήκες πλήρους φορτίου. Ο λόγος της κινητικής ενέργειας που παράγεται ως προς την ηλεκτρική ενέργεια που δίνουμε στον κινητήρα ονομάζεται απόδοση του κινητήρα. Η απόδοση του κινητήρα μειώνεται με την μείωση του συντελεστή φορτίου. Ένας άλλος παράγοντας που επηρεάζει το ποσό της θερμότητας που εκλύεται είναι ο συντελεστής χρήσης, με την τιμή 1 να αντιστοιχεί σε πλήρη χρήση. Η συνολική θερμότητα που εκλύεται από μια ηλεκτρική συσκευή που περιέχει ηλεκτρικό κινητήρα δίνεται από την παρακάτω εξίσωση:

$$Q_{ea} = \frac{W_{motor} * f_{load} * f_{usage}}{\eta}$$

W_{motor} : η ισχύς του κινητήρα

f_{load} : συντελεστής φορτίου

f_{usage} : συντελεστής χρήσης

η : απόδοση κινητήρα

Η παραπάνω εξίσωση μπορεί να χρησιμοποιηθεί για συσκευές όπως το ψυγείο και το πλυντήριο. Συσκευές όπως η τηλεόραση και η ηλεκτρική κουζίνα περιέχουν μόνο ηλεκτρική αντίσταση και η θερμότητα που εκλύουν ισούται με αυτήν που παράγει η συνολική τους αντίσταση. Υπολογίζεται από την σχέση :

$$Q_{tv, cooker} = V * I = I^2 * R$$

V: τάση λειτουργίας

I: ένταση ηλεκτρικού ρεύματος

R: ολική ηλεκτρική αντίσταση της συσκευής

4.3 Ενεργειακά συστήματα κτιρίων

4.3.1 Ηλιακοί συλλέκτες

Οι παθητικοί ηλιακοί συλλέκτες αποτελούνται από μια μαύρη επιφάνεια που απορροφάει ηλιακή ακτινοβολία, ένα διάκενο αέρα και ένα γυάλινο κάλυμμα. Ο αέρας μέσα στο διάκενο θερμαίνεται και με την σειρά του θερμαίνει το νερό που ρέει στους σωλήνες του, παρέχοντας ζεστό νερό χρήσης. Τοποθετούνται συνήθως στις οροφές των κτιρίων.

4.3.2 Λέβητες

Λέβητας είναι ένα δοχείο πίεσης στο οποίο μεταφέρεται θερμότητα από τα καυσαέρια σε ένα ρευστό. Για να επιτευχθεί υψηλή απόδοση, το νερό που περιέχεται στα καυσαέρια πρέπει να συμπυκνωθεί και η λανθάνουσα θερμότητα που θα απελευθερωθεί πρέπει να χρησιμοποιηθεί για την επαναθέρμανση του νερού επιστροφής. Αυτό προϋποθέτει ότι η θερμοκρασία επιστροφής νερού να είναι κάτω από το σημείο δρόσου των καυσαερίων (περίπου 55 ° C για τους λέβητες φυσικού αερίου)[15]. Όσο ψυχρότερο είναι το νερό επιστροφής, τόσο μεγαλύτερη είναι η ποσότητα του νερού που θα συμπυκνωθεί, άρα τόσο υψηλότερη η απόδοση. Οι λέβητες συμπύκνωσης έχουν απόδοση περίπου 85%-95%.

4.3.3 Συσκευές θέρμανσης με ηλεκτρική αντίσταση

Η απόδοση των συσκευών θέρμανσης με ηλεκτρική αντίσταση, είναι 100%, καθώς το σύνολο της ηλεκτρικής ενέργειας που χρησιμοποιείται μετατρέπεται σε θερμότητα. Στο παρών σύστημα ηλεκτρικής παραγωγής, ένα μεγάλο μέρος της ηλεκτρικής ενέργειας παράγεται από καύση άνθρακα με χαμηλή απόδοση, πράγμα που σημαίνει ότι η μετάβαση από ηλεκτρική θέρμανση σε χρήση πετρελαίου ή φυσικού αερίου με ένα λέβητα υψηλής απόδοσης θα οδηγούσε σε σημαντική μείωση στη χρήση πρωτογενούς ενέργειας.

4.3.4 Κλιματιστικά

Ένα κλιματιστικό μπορεί να θεωρηθεί ως μια μικρή αντλία θερμότητας που λειτουργεί στην λειτουργία ψύξης. Χρησιμοποιείται σε κατοικίες και σε μικρά εμπορικά κτίρια. Όπως και στις αντλίες θερμότητας, η απόδοση ενός κλιματιστικού χαρακτηρίζεται από τον βαθμό απόδοσης COP, ο οποίος είναι ο λόγος της θερμότητας που αφαιρεί από ένα κτίριο προς την χρησιμοποιούμενη ενέργεια. Τα κλιματιστικά συμπιέζουν και εκτονώνουν το ψυκτικό τους μέσο. Το ψυκτικό μέσο εξατμίζεται σε χαμηλή πίεση σε έναν εξατμιστήρα, όπου καθίσταται ψυχρότερο από τον περιβάλλοντα αέρα (αντλώντας έτσι θερμότητα από τον αέρα), μετά συμπυκνώνεται υπό πίεση σε ένα συμπυκνωτή, όπου γίνεται θερμότερο από τον περιβάλλοντα αέρα (απελευθερώνοντας έτσι θερμότητα στον αέρα).

4.3.5 Θερμοσίφωνες

Οι ηλεκτρικοί θερμοσίφωνες χρησιμοποιούν την θερμότητα που εκλύει η ηλεκτρική αντίσταση τους για να θερμάνουν το νερό, και έχουν 100% απόδοση. Όμως παρόλο που έχουν υψηλή απόδοση, αυξάνουν πολύ την πρωτογενή κατανάλωση ενέργειας, διότι η ηλεκτρική ενέργεια που χρησιμοποιούν παρέχεται από το δίκτυο παροχής ηλεκτρικής ενέργειας, που έχει χαμηλό βαθμό απόδοσης, της τάξης του 30% με 35%. Οι θερμοσίφωνες που χρησιμοποιούν φυσικό αέριο για την θέρμανση του νερού, έχουν απόδοση 76% με 85% [15]. Αν το νερό αποθηκευτεί σε μια δεξαμενή για να χρησιμοποιηθεί αργότερα, η θερμότητα θα ρέει μέσω των τοιχωμάτων της δεξαμενής και θα χαθεί. Οι απώλειες θερμότητας της δεξαμενής εξαρτώνται από τον λόγο της επιφάνειας της δεξαμενής ως προς τον όγκο της, στην θερμοκρασιακή διαφορά μεταξύ του εσωτερικού της δεξαμενής και του περιβάλλοντα χώρου, καθώς και από την αντίσταση της μόνωσης.

ΚΕΦΑΛΑΙΟ 5 : ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΜΟΝΤΕΛΟΥ

5.1 Σκοπός του μοντέλου

Στο κεφάλαιο αυτό παρουσιάζεται το μοντέλο στο οποίο βασίστηκε η παρούσα εργασία. Το μοντέλο περιέχει τις απαραίτητες εξισώσεις, περιορισμούς, παραμέτρους και παραδοχές που χρειάζονται για να υπολογιστούν οι ενεργειακές ανάγκες ενός νέου κτιρίου και να επιλεγθούν τα κατάλληλα συστήματα, αφού το μοντέλο δώσει το σύνολο των λύσεων του προβλήματος. Βασίζεται στο μοντέλο της επιστημονικής δημοσίευσης των M.Karmellos et al [1]"A multi-objective approach for optimal prioritization of energy efficiency measures in buildings " αλλά έχει τις εξής τροποποιήσεις:

1. Έχουν γραμμικοποιηθεί οι εξισώσεις για τον υπολογισμό των ενεργειακών αναγκών (το τελικό πρόβλημα είναι γραμμικό).
2. Έχει προστεθεί η αντικειμενική συνάρτηση των συνολικών εκπομπών διοξειδίου του άνθρακα.
3. Επιλύεται με την μέθοδο AUGMECON2 αντί της μεθόδου των συντελεστών στάθμισης.
4. Έχει μοντελοποιηθεί σε πλατφόρμα GAMS (η προηγούμενη μοντελοποίηση ήταν σε MATLAB).

5.2 Περιγραφή των εξισώσεων του μοντέλου

5.2.1 Αντικειμενικές Συναρτήσεις

Το πρόβλημα που μελετάται έχει 3 αντικειμενικές συναρτήσεις :

1. Συνολική πρωτογενής κατανάλωση ενέργειας
2. Συνολικές εκπομπές διοξειδίου του άνθρακα
3. Συνολικό ετήσιο κόστος

Παρακάτω περιγράφονται όλες τις εξισώσεις αναλυτικά, ώστε να ορισθούν με λεπτομέρειες οι 3 αντικειμενικές συναρτήσεις του προβλήματος.

5.2.2 Εξισώσεις επιλογής μερών του κτιριακού κελύφους

Για την επιλογή συστημάτων χρησιμοποιήθηκαν δυαδικές μεταβλητές. Όταν επιλέγεται ένα σύστημα η αντίστοιχη δυαδική μεταβλητή παίρνει την τιμή 1, αλλιώς την τιμή 0. Για να πάρει τιμή αυτή η δυαδική μεταβλητή πρέπει να ορισθεί η αντίστοιχη εξίσωση. Οι εξισώσεις για την επιλογή των μερών του κτιριακού κελύφους παρατίθενται παρακάτω :

- **Πόρτες**

Για την επιλογή πόρτας χρησιμοποιείται η εξίσωση :

$$\sum_{d=1}^D x_d^{DOOR} = 1 \quad (1), \text{ όπου}$$

x_d^{DOOR} : η δυαδική μεταβλητή για την επιλογή της πόρτας d από ένα σύνολο D διαθέσιμων επιλογών

- **Παράθυρα**

Για την επιλογή παραθύρων χρησιμοποιείται η εξίσωση:

$$\sum_{z=1}^Z x_z^{WIN} = 1 \quad (2), \text{ όπου}$$

x_z^{WIN} : η δυαδική μεταβλητή για την επιλογή του παραθύρου z από ένα σύνολο Z διαθέσιμων επιλογών

- **Τοίχοι**

Για την επιλογή τοίχων χρησιμοποιείται η εξίσωση:

$$\sum_{w=1}^W x_w^{WALL} = 1 \quad (3), \text{ όπου}$$

x_w^{WALL} : η δυαδική μεταβλητή για την επιλογή του τοίχου w από ένα σύνολο W διαθέσιμων επιλογών

- **Οροφές**

Για την επιλογή της οροφής χρησιμοποιείται η εξίσωση:

$$\sum_{r=1}^R x_r^{CEIL} = 1 \quad (4), \text{ όπου}$$

x_r^{CEIL} : η δυαδική μεταβλητή για την επιλογή της οροφής r από ένα σύνολο R διαθέσιμων επιλογών

- **Δάπεδα**

Για την επιλογή του δαπέδου χρησιμοποιείται η εξίσωση :

$$\sum_{h=1}^H x_h^{FLO} = 1 \quad (5), \text{ όπου}$$

x_r^{CEIL} : η δυαδική μεταβλητή για την επιλογή του δαπέδου r από ένα σύνολο R διαθέσιμων επιλογών

5.2.3 Εξισώσεις επιλογής ενεργειακών συστημάτων

- **Συστήματα θέρμανσης**

Για την επιλογή του συστήματος θέρμανσης χρησιμοποιείται η εξίσωση:

$$\sum_{ehsi=1}^{EHSI} x_{ehsi}^{EHS} + \sum_{nehsi=1}^{NEHSI} x_{nehsi}^{NEHS} + \sum_{ehcsi=1}^{EHCSI} x_{ehcsi}^{EHCS} + \sum_{ehwsi=1}^{EHWSI} x_{ehwsi}^{EHWS} + \sum_{nehwsi=1}^{NEHWSI} x_{nehwsi}^{NEHWS} = 1 \quad (6), \text{ όπου}$$

x_{ehsi}^{EHS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης ehsi από ένα σύνολο EHS διαθέσιμων επιλογών

x_{nehsi}^{NEHS} : η δυαδική μεταβλητή για την επιλογή του μη ηλεκτρικού συστήματος θέρμανσης nehsi από ένα σύνολο NEHS διαθέσιμων επιλογών

x_{ehcsi}^{EHCS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης-ψύξης ehcsi από ένα σύνολο EHCS διαθέσιμων επιλογών

x_{ehwsi}^{EHWS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης χώρου και νερού ehwsi από ένα σύνολο EHWS διαθέσιμων επιλογών

x_{nehwsi}^{NEHWS} : η δυαδική μεταβλητή για την επιλογή του μη ηλεκτρικού συστήματος θέρμανσης χώρου και νερού nehwsi από ένα σύνολο NEHWS διαθέσιμων επιλογών

- **Συστήματα ψύξης**

Για την επιλογή του συστήματος ψύξης χρησιμοποιείται η εξίσωση:

$$\sum_{ecsi=1}^{ECSI} x_{ecsi}^{ECS} + \sum_{ehcsi=1}^{EHCSI} x_{ehcsi}^{EHCS} = 1 \quad (7), \text{ όπου}$$

x_{ecsi}^{ECS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος ψύξης $ecsi$ από ένα σύνολο ECS διαθέσιμων επιλογών

x_{ehcsi}^{EHCS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης-ψύξης $ehcsi$ από ένα σύνολο EHCS διαθέσιμων επιλογών

- **Συστήματα θέρμανσης νερού**

Για την επιλογή του συστήματος θέρμανσης νερού χρησιμοποιείται η εξίσωση:

$$\sum_{ewsi=1}^{EWSI} x_{ewsi}^{EWS} + \sum_{newsi=1}^{NEWSI} x_{newsi}^{NEWS} + \sum_{ehwsi=1}^{EHWI} x_{ehwsi}^{EHS} + \sum_{nehwsi=1}^{NEHWI} x_{nehwsi}^{NEHS} = 1 \quad (8), \text{ όπου}$$

x_{ewsi}^{EWS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης νερού $ewsi$ από ένα σύνολο EWS διαθέσιμων επιλογών

x_{newsi}^{NEWS} : η δυαδική μεταβλητή για την επιλογή του μη ηλεκτρικού συστήματος θέρμανσης νερού $newsi$ από ένα σύνολο NEWS διαθέσιμων επιλογών

x_{ehwsi}^{EHS} : η δυαδική μεταβλητή για την επιλογή του ηλεκτρικού συστήματος θέρμανσης χώρου και νερού $ehwsi$ από ένα σύνολο EHS διαθέσιμων επιλογών

x_{nehwsi}^{NEHS} : η δυαδική μεταβλητή για την επιλογή του μη ηλεκτρικού συστήματος θέρμανσης χώρου και νερού $nehwsi$ από ένα σύνολο NEHS διαθέσιμων επιλογών

- **Ηλιακός θερμοσίφωνα**

Για την επιλογή του ηλιακού θερμοσίφωνα χρησιμοποιείται η εξίσωση:

$$\sum_{slci=1}^{SLCI} x_{slci}^{SLC} \leq 1 \quad (9), \text{ όπου}$$

x_{slci}^{SLC} : η δυαδική μεταβλητή για την επιλογή του ηλιακού θερμοσίφωνα $slci$ από ένα σύνολο SLCI διαθέσιμων επιλογών

Η επιλογή του ηλιακού θερμοσίφωνα είναι προαιρετική, για αυτό στην ανίσωση έχουμε <<μικρότερο ή ίσο>> αντί για απλή ισότητα.

- **Φωτοβολταϊκά**

Για την επιλογή του φωτοβολταϊκού χρησιμοποιείται η εξίσωση:

$$\sum_{resi=1}^{RESI} x_{resi}^{RES} \leq 1 \text{ (10), όπου}$$

x_{resi}^{RES} : η δυαδική μεταβλητή για την επιλογή του φωτοβολταϊκού $resi$ από ένα σύνολο SLCI διαθέσιμων επιλογών

Η επιλογή του φωτοβολταϊκού είναι προαιρετική, για αυτό στην ανίσωση έχουμε <<μικρότερο ή ίσο>> αντί για απλή ισότητα.

- **Συστήματα φωτισμού**

Για την επιλογή του συστήματος φωτισμού χρησιμοποιείται η εξίσωση :

$$\sum_{li=1}^{LI} x_{li}^{LAMP} = 1 \text{ (11), όπου}$$

x_{li}^{LAMP} : η δυαδική μεταβλητή για την επιλογή του συστήματος φωτισμού li από ένα σύνολο LI διαθέσιμων επιλογών

- **Τηλεόραση**

Για την επιλογή τηλεόρασης χρησιμοποιείται η εξίσωση :

$$\sum_{tvi=1}^{TVS} x_{tvi}^{TV} = 1 \text{ (12), όπου}$$

x_{tvi}^{TV} : η δυαδική μεταβλητή για την επιλογή της τηλεόρασης tvi από ένα σύνολο TVS διαθέσιμων επιλογών

Πλυντήριο

Για την επιλογή πλυντηρίου χρησιμοποιείται η εξίσωση:

$$\sum_{wash=1}^{WASH} x_{wash}^{WASHING} = 1 \text{ (13), όπου}$$

$x_{wash}^{WASHING}$: η δυαδική μεταβλητή για την επιλογή του πλυντηρίου $wash$ από ένα σύνολο WASH διαθέσιμων επιλογών

- **Ηλεκτρική κουζίνα**

Για την επιλογή ηλεκτρικής κουζίνας χρησιμοποιείται η εξίσωση:

$$\sum_{cooki=1}^{COOK} x_{cooki}^{COOKING} = 1 \text{ (14), όπου}$$

$x_{cooki}^{COOKING}$: η δυαδική μεταβλητή για την επιλογή της ηλεκτρικής κουζίνας cooki από ένα σύνολο COOK διαθέσιμων επιλογών

Ψυγείο

Για την επιλογή ψυγείου χρησιμοποιείται η εξίσωση:

$$\sum_{refi=1}^{REFRIG} x_{refi}^{REF} = 1 \quad (15), \text{ όπου}$$

x_{refi}^{REF} : η δυαδική μεταβλητή για την επιλογή του ψυγείου refi από ένα σύνολο REF διαθέσιμων επιλογών

5.2.4 Εξισώσεις υπολογισμού ενεργειακών αναγκών

Αρχικά υπολογίζονται οι μηνιαίες ενεργειακές ανάγκες του κτιρίου.

- Μηνιαίες ανάγκες θέρμανσης

Οι ανάγκες θέρμανσης δίνονται από την παρακάτω εξίσωση :

$$Q_m^{HD} = HS_m * F_{conv} * (Q_{T,m} + Q_{VEN,m} - Q_{INHG,m} - Q_{SL,m}) \quad (16), \text{ αν } Q_m^{HD} > 0, \text{ αλλιώς}$$

$$Q_m^{HD} = 0$$

$$Q_{T,m} = BLC * (T_{IH} - T_{o,m}) * t_m \quad (17) : \text{ απώλειες ενέργειας μέσω των τοιχωμάτων (kWh)}$$

$$Q_{VEN,m} = d_{air} * CP_{air} * ACH * V * (T_{IH} - T_{o,m}) * t_m * /3600 \quad (18) : \text{ απώλειες λόγω εξαερισμού (kWh)}$$

$$Q_{INHG,m} = (n_{people} * Q_{people} + Q_{ea,m}) * t_m \quad (19) : \text{ θερμότητα που παράγεται από τους ανθρώπους και τις ηλεκτρικές συσκευές (kWh)}$$

$$Q_{SL,m} = A_{windows} * F_F * F_S * F_C * I_{SL,m} * t_d * x_z^{WIN} \quad (20) : \text{ θερμικό ηλιακό κέρδος (kWh)}$$

$$BLC = \sum_{com=1}^{COM} A_{com} * U_{com} * b_{com} \quad (21)$$

Ο όρος BLC υπολογίζεται για com=πόρτες, παράθυρα, δάπεδα, οροφές, τοίχους

A_{com} : επιφάνεια στοιχείου του κτιρίου (πόρτες, παράθυρα, δάπεδα, οροφές, τοίχους) (m²)

U_{com} : συντελεστής μεταφοράς θερμότητας στοιχείου του κτιρίου (πόρτες, παράθυρα, δάπεδα, οροφές, τοίχους) ($W/m^2 \cdot K$)

b_{com} : δυαδική παράμετρος που παίρνει την τιμή 1 αν το εκάστοτε στοιχείου του κτιρίου δέχεται ροή θερμότητας και 0 αν δεν δέχεται

HS_m : δυαδική παράμετρος που παίρνει την τιμή 1 όταν απαιτείται θέρμανση και 0 όταν δεν απαιτείται

F_{conv} : συντελεστής μετατροπής των kWh σε MJ (3,6 MJ/kWh)

BLC : συντελεστής φορτίου του κτιρίου

F_F : λόγος επιφάνειας τζαμιού προς συνολική την επιφάνεια παραθύρου

F_S : συντελεστής διόρθωσης σκίασης παραθύρου

F_C : συντελεστής διόρθωσης σκίασης παραθύρου για μετακινούμενα αντικείμενα

$A_{windows}$: επιφάνεια παραθύρου

$I_{SL,m}$: ηλιακή ακτινοβολία προς προσπίπτει στο παράθυρο

t_d : διάρκεια του μήνα m σε μέρες

t_m : διάρκεια του μήνα m σε ώρες

T_{IH} : εσωτερική θερμοκρασία σχεδιασμού για την περίοδο θέρμανσης ($^{\circ}C$) (

$T_{0,m}$: μέση εξωτερική θερμοκρασία του μήνα m ($^{\circ}C$)

CP_{air} : θερμοχωρητικότητα του αέρα ($kJ/mol \cdot K$)

d_{air} : πυκνότητα αέρα (kg/m^3)

ACH : ρυθμός ανανέωσης του εσωτερικού αέρα του κτιρίου

V : όγκος του αέρα του κτιρίου (m^3)

Q_{people} : θερμότητα του εκπέμπει ένας άνθρωπος

n_{people} : πλήθος ανθρώπων στο εσωτερικό του κτιρίου

Q_{ea} : θερμότητα του εκπέμπουν οι ηλεκτρικές συσκευές

- Μηνιαίες ανάγκες ψύξης

Οι ανάγκες ψύξης δίνονται από την παρακάτω εξίσωση :

$$Q_m^{CD} = CS_m * F_{conv} * (Q_{INH.G.m} + Q_{SL,m} - Q_{T,m} - Q_{VEN,m}) \quad (22), \text{ αν } Q_m^{CD} > 0, \text{ αλλιώς } Q_m^{CD} = 0$$

$$Q_{T,m} = BLC * (T_{IC} - T_{0,m}) * t_m \quad (23)$$

$$Q_{VEN,m} = d_{air} * CP_{air} * ACH * V * (T_{IC} - T_{0,m}) * t_m / 3600 + d_{air} * h_{fg} * ACH * V * (w_1 - w_{0,m}) * t_m / 3600 \quad (24)$$

CS_m : δυαδική παράμετρος που παίρνει την τιμή 1 όταν απαιτείται θέρμανση και 0 όταν δεν απαιτείται

T_{IC} : εσωτερική θερμοκρασία σχεδιασμού για την περίοδο ψύξης ($^{\circ}C$)

α : απορροφητικότητα του υλικού στο εξωτερικό του τοίχου

h_{fg} : λανθάνουσα θερμότητα εξάτμισης

w_i : εσωτερική θερμοκρασία ειδική υγρασία

$w_{0,m}$: μέση εξωτερική υγρασία

q_{sol} ηλιακή ακτινοβολία που προσπίπτει στην επιφάνεια του τοίχου

- Μηνιαίες ανάγκες θέρμανσης νερού

Οι ανάγκες θέρμανσης του νερού δίνονται από την παρακάτω εξίσωση :

$$Q_m^{DHW} = WS_m * F_{conv} * (Q_{dhwu,m} - Q_{dSLC,m}) \quad (25), \text{ αν } Q_{dhwu,m} > Q_{dSLC,m} \text{ αλλιώς } Q_{DHW,m} = 0$$

WS_m : δυαδική παράμετρος που παίρνει την τιμή 1 όταν απαιτείται η θέρμανση του νερού και την τιμή 0 όταν δεν απαιτείται

$Q_{dhwu,m}$: θερμότητα που απαιτείται για την θέρμανση του νερού χρήσης τον μήνα m

$Q_{dSLC,m}$: ενέργεια που παρέχεται από τον ηλιακό θερμοσίφωνα για την θέρμανση του νερού χρήσης τον μήνα m

$$Q_{dhwu,m} = \dot{m} * d_{water} * CP_{water} * (T_{DHW} - T_{DCW,m}) * t_m \quad (26)$$

m : ροή ζεστού νερού χρήσης (m^3/s)

d_{water} : πυκνότητα νερού

CP_{water} : θερμοχωρητικότητα του νερού

T_{DHW} : επιθυμητή θερμοκρασία του ζεστού νερού χρήσης ($^{\circ}C$)

$T_{DCW,m}$: θερμοκρασία του νερού παροχής τον μήνα m (°C)

$$Q_{dSLC,m} = F_{conv} * A_{SLC} * F_{SLC} * I_{SLC,m} * t_d * \sum_{slci=1}^{SLCI} x_{slci}^{SLC} * n_{slci} \quad (27)$$

A_{SLC} : επιφάνεια ηλιακού θερμοσίφωνα

F_{SLC} : συντελεστής διόρθωσης σκίασης του θερμοσίφωνα

$I_{SLC,m}$: ηλιακή ακτινοβολία που προσπίπτει στον ηλιακό θερμοσίφωνα

n_{slci} : απόδοση του ηλιακού θερμοσίφωνα

- Μηνιαίες ανάγκες ηλεκτρικής ενέργειας για φωτισμό

Οι ανάγκες ηλεκτρικής ενέργειας για φωτισμό δίνονται από την παρακάτω εξίσωση :

$$Q_m^{LD} = F_{conv} * t_d * \sum_{l=1}^L P_l * f_{use} * \sum_{li=1}^{LI} x_{li}^{LAMP} \quad (28)$$

P_l : ισχύς λαμπτήρα

f_{use} : ημερήσια χρήση λαμπτήρα

L : αριθμός λαμπτήρων

- Μηνιαία παραγωγή ηλεκτρικής ενέργειας από φωτοβολταϊκά

Η μηνιαία παραγωγή ηλεκτρικής ενέργειας δίνεται από την παρακάτω εξίσωση :

$$Q_m^{PV} = A_{pv} * n_{pv} * PR_{pv} * F_{pv} * I_{SL,m} \quad (29)$$

A_{pv} : επιφάνεια φωτοβολταϊκού

n_{pv} : απόδοση του φωτοβολταϊκού

PR_{pv} : απόδοση που εκφράζει τις απώλειες του συστήματος

F_{pv} : συντελεστής σκίασης

$I_{SL,m}$: ηλιακή ακτινοβολία προσπίπτουσα στο φωτοβολταϊκό τον μήνα m

Μηνιαίες ανάγκες ηλεκτρικής ενέργειας για τις ηλεκτρικές συσκευές

Οι ανάγκες ηλεκτρικής ενέργειας για τις ηλεκτρικές συσκευές δίνονται από την παρακάτω

εξίσωση :

$$Q_m^{EA} = F_{conv} * t_d * \sum_{ea=1}^{EA} (x_{ea}^{EAS} * P_{ea} * f_{use_ea} * f_{load}) \quad (30)$$

P_{ea} : ισχύς ηλεκτρικής συσκευής

f_{use_ea} : ημερήσια χρήση συσκευής

f_{load} : συντελεστής φορτίου συσκευής

Μετά υπολογίζονται οι ετήσιες ενεργειακές ανάγκες.

- Ετήσια απαιτούμενη ενέργεια για θέρμανση

Η ετήσια απαιτούμενη ενέργεια για θέρμανση δίνεται από την παρακάτω εξίσωση :

$$Q^{HD} = \sum_{m=1}^{12} Q_m^{HD} \quad (31)$$

- Ετήσια απαιτούμενη ενέργεια για ψύξη

Η ετήσια απαιτούμενη ενέργεια για ψύξη δίνεται από την παρακάτω εξίσωση :

$$Q^{CD} = \sum_{m=1}^{12} Q_m^{CD} \quad (32)$$

- Ετήσια απαιτούμενη ενέργεια για την θέρμανση του νερού

Η ετήσια απαιτούμενη ενέργεια για την θέρμανση δίνεται από την παρακάτω εξίσωση :

$$Q^{WD} = \sum_{m=1}^{12} Q_m^{DHW} \quad (33)$$

- Ετήσια απαιτούμενη ενέργεια για φωτισμό

Η ετήσια απαιτούμενη ενέργεια για φωτισμό δίνεται από την παρακάτω εξίσωση :

$$Q^{LD} = \sum_{m=1}^{12} Q_m^{LD} \quad (34)$$

- Ετήσια παραγωγή ηλεκτρικής ενέργειας για τις ηλεκτρικές συσκευές

Η ετήσια παραγωγή ηλεκτρικής ενέργειας για τις ηλεκτρικές συσκευές δίνεται από την παρακάτω εξίσωση :

$$Q^{EA} = \sum_{m=1}^{12} Q_m^{EA} \quad (35)$$

- Ετήσια παραγωγή ηλεκτρικής ενέργειας από φωτοβολταϊκά

Η ετήσια παραγωγή ηλεκτρικής ενέργειας από φωτοβολταϊκά δίνεται από την παρακάτω εξίσωση :

$$Q^{PV} = \sum_{m=1}^{12} Q_m^{PV} \quad (36)$$

5.2.5 Υπολογισμός κατανάλωσης πρωτογενούς ενέργειας

Για να υπολογιστεί η πρωτογενής κατανάλωση ενέργειας ενός συστήματος πρέπει πρώτα να υπολογιστεί η συνολικά καταναλισκόμενη ενέργεια. Για να βρεθεί η συνολικά καταναλισκόμενη ενέργεια πολλαπλασιάζεται ο ολικός συντελεστή απόδοσης του εκάστοτε συστήματος με την αντίστοιχη ζήτηση ενέργειας.

Παρακάτω γίνονται οι υπολογισμοί για την περίπτωση του συστήματος θέρμανσης με ηλεκτρική ενέργεια.

Ο ολικός συντελεστής απόδοσης του συστήματος θέρμανσης με ηλεκτρική ενέργεια δίνεται από την παρακάτω εξίσωση :

$$SEH_{el} = \sum_{ehsi=1}^{EHSI} \frac{x_{ehsi}^{EHS}}{n_{ehsi}^{EHS}} + \sum_{ehcsi=1}^{EHCSI} \frac{x_{ehcsi}^{EHCS}}{n_{ehcsi}^{EHCS}} + \sum_{ehwsi=1}^{EHWSI} \frac{x_{ehwsi}^{EHWS}}{n_{ehwsi}^{EHWS}} \quad (37)$$

n_{ehsi}^{EHS} : συντελεστής απόδοσης του συστήματος θέρμανσης με ηλεκτρική ενέργεια ehsi

Με παρόμοιο τρόπο υπολογίζονται και οι ολικοί συντελεστές απόδοσης των συστημάτων θέρμανσης με καύσιμο, ψύξης, θέρμανσης του νερού με ηλεκτρική ενέργεια και με καύσιμο.

Η συνολικά καταναλισκόμενη ενέργεια δίνεται από την παρακάτω εξίσωση :

$$Q_{el}^H = Q^{HD} * SEH_{el} \quad (38)$$

Παρατηρείται ότι η παραπάνω εξίσωση είναι μη γραμμική. Στο παρών μοντέλο επιλέχθηκε να γραμμικοποιηθεί η παραπάνω εξίσωση σύμφωνα με την ακόλουθη μέθοδο:

Ορίζονται οι βοηθητικές μεταβλητές y_1, y_2, y_3 και ένας μεγάλο θετικός αριθμός $mpos$.

Με την βοήθεια των παραπάνω υπολογίζεται η συνολικά καταναλισκόμενη ενέργεια του συστήματος 1 με τις παρακάτω ανισώσεις:

$$y_1 \leq mpos * x_{EHS1}^{EHS} \quad (39)$$

$$y_1 \leq Q^{HD} \quad (40)$$

$$y_1 \geq Q^{HD} - mpos * (1 - x_{EHS1}^{EHS}) \quad (41)$$

$$y_1 \geq 0 \quad (42)$$

EHS1: το πρώτο σύστημα θέρμανσης με ηλεκτρική ενέργεια

Οι μεταβλητές y_2, y_3 ορίζονται με παρόμοιο τρόπο για EHS2 και EHS3.

Με την παραπάνω μέθοδο αν επιλεγεί παραδείγματος χάριν το σύστημα 1, η μεταβλητή απόφασης θα πάρει την τιμή 1. Η ανίσωση (41) θα γίνει:

$$y_1 \geq Q^{HD}$$

Σε συνδυασμό με την ανίσωση (40) η μεταβλητή y_1 θα πάρει την τιμή Q^{HD} .

Σε περίπτωση που δεν επιλεγεί το σύστημα 1, η μεταβλητή απόφασης θα πάρει την τιμή 0. Έτσι η ανίσωση (39) θα γίνει:

$$y_1 \leq 0$$

Σε συνδυασμό με την ανίσωση (42) η μεταβλητή y_1 θα πάρει την τιμή 0.

Και στις 2 περιπτώσεις η μεταβλητή y_1 "αναγκάζεται" να πάρει την κατάλληλη τιμή, διότι μόνο η ισότητα ικανοποιεί όλες τις ανισώσεις.

Έτσι υπολογίζεται η συνολικά καταναλισκόμενη ενέργεια αθροίζοντας όλες τις περιπτώσεις

$$Q_{el}^H = \frac{y_1}{n_{EHS1}^{EHS}} + \frac{y_2}{n_{EHS2}^{EHS}} + \frac{y_3}{n_{EHS3}^{EHS}} \quad (43)$$

Επειδή μόνο ένα σύστημα θα επιλεγεί, μόνο μία εκ των μεταβλητών θα πάρει τιμή διάφορη του μηδενός.

Με την ίδια μέθοδο υπολογίζεται και η συνολική κατανάλωση ενέργειας για θέρμανση με καύσιμο, για ψύξη, για θέρμανση του νερού με ηλεκτρική ενέργεια και με καύσιμο.

Η συνολική απαιτούμενη ηλεκτρική ενέργεια υπολογίζεται από την παρακάτω εξίσωση :

$$Q_{el}^D = Q_{el}^H + Q_{el}^C + Q_{el}^W + Q_{el}^L + Q_{el}^{EA} \quad (44)$$

Αφαιρώντας την ενέργεια που παράγουν τα φωτοβολταϊκά βρίσκεται η ηλεκτρική ενέργεια που παρέχει το δίκτυο.

$$Q_{el}^{grid} = Q_{el}^D - Q^{PV} \quad (45)$$

Αθροίζοντας τις καταναλώσεις των μη ηλεκτρικών συστημάτων βρίσκεται η συνολική κατανάλωση ενέργειας των μη ηλεκτρικών συστημάτων.

$$Q_{nel} = Q_{nel}^H + Q_{nel}^W \quad (46)$$

Q_{nel}^H : ενεργειακή κατανάλωση μη ηλεκτρικών συστημάτων θέρμανσης

Q_{nel}^W : ενεργειακή κατανάλωση μη ηλεκτρικών συστημάτων θέρμανσης νερού

Τώρα μπορεί να ορισθεί η πρώτη αντικειμενική συνάρτηση, αυτή της κατανάλωσης πρωτογενούς ενέργειας. Η συνολική κατανάλωση πρωτογενούς ενέργειας ορίζεται από την παρακάτω εξίσωση :

$$Q_t = \frac{Q_{el}^{grid}}{n_{grid}} + Q_{nel} \quad (47)$$

5.2.6 Υπολογισμός ολικών συντελεστών θερμότητας τοίχων-δαπέδων-οροφών

Διατίθενται 3 επιλογές για τοίχους, δάπεδα και οροφές και 3 υλικά για την μόνωση αυτών. Ο υπολογισμός του ολικού συντελεστή μεταφοράς θερμότητας κάθε στοιχείου γίνεται με βάση την εξίσωση (49):

$$U = \left(\frac{1}{h_i} + \frac{1}{h_0} + \frac{L}{k} \right)^{-1} \quad (48)$$

Οι υπολογισμοί των ολικών συντελεστών έγιναν σε ένα φύλλο excel και εισήχθησαν σαν παράμετροι με πίνακα 12 στοιχείων. Τα στοιχεία 1-3 αντιστοιχούν στις 3 διαθέσιμες επιλογές χωρίς μόνωση, τα στοιχεία 4-6 αντιστοιχούν στις επιλογές 1,2 και 3 με υλικό μόνωσης 1, τα στοιχεία 7-9 στις επιλογές 1,2, και 3 με υλικό μόνωσης 2 και τέλος τα στοιχεία 10-12 αντιστοιχούν στις επιλογές 1,2 και 3 με υλικό μόνωσης 3.

5.2.7 Υπολογισμός εκπομπών διοξειδίου του άνθρακα

Για τον υπολογισμό των εκπομπών διοξειδίου του άνθρακα του δικτύου χρησιμοποιείται η παρακάτω εξίσωση :

$$CO_2_{-grid} = Q_{el}^{grid} * grid_{-CO_2} \quad (49)$$

$grid_{CO_2}$: συντελεστής εκπομπής CO₂ του δικτύου (0.295 kgCO₂/MJ)

Για τον υπολογισμό των εκπομπών διοξειδίου από μη ηλεκτρικά συστήματα χρησιμοποιείται η παρακάτω εξίσωση:

$$CO_2_non_electric_i = \frac{y_i * emission_factor_{fuel} * d_{fuel}}{efficiency_i * lhv_{fuel}} \quad (50)$$

y_i : η βοηθητική μεταβλητή που παίρνει την τιμή της ενεργειακής κατανάλωσης του μη ηλεκτρικού συστήματος i ($i=NEHS, NEHWS, NEWS$)

$emission_factor_{fuel}$: συντελεστής εκπομπής του καυσίμου $fuel$ ($fuel=natural\ gas, oil$)

d_{fuel} : πυκνότητα του καυσίμου $fuel$

$efficiency_i$: βαθμός απόδοσης του μη ηλεκτρικού συστήματος i

lhv_{fuel} : κατώτερη θερμογόνο δύναμη του καυσίμου $fuel$

Αθροίζοντας τις εκπομπές όλων των μη ηλεκτρικών συστημάτων, υπολογίζονται οι συνολικές εκπομπές διοξειδίου των μη ηλεκτρικών συστημάτων :

$$CO_2_nel = \sum_{i=1}^N \frac{y_i * emission_factor_{fuel} * d_{fuel}}{efficiency_i * lhv_{fuel}} \quad (51)$$

Με την χρήση των εξισώσεων (49) και (51), ορίζεται η δεύτερη αντικειμενική συνάρτηση, αυτή των συνολικών εκπομπών διοξειδίου ως εξής :

$$Total_CO_2 = CO_2_nel + CO_2_grid \quad (52)$$

5.2.8 Υπολογισμός συνολικού ετήσιου κόστους

Για τον υπολογισμό του συνολικού ετήσιου κόστους είναι απαραίτητος ο συντελεστής ανάκτησης κεφαλαίου. Ο συντελεστής ανάκτησης κεφαλαίου ορίζεται από την παρακάτω εξίσωση:

$$ann_factor = \frac{rate * (1 + rate)^{lifetime}}{(1 + rate)^{lifetime} - 1} \quad (53)$$

$rate$: επιτόκιο προεξόφλησης

$lifetime$: χρόνος ζωής επένδυσης

Τα μέρη του κτιριακού κελύφους θεωρείται ότι έχουν 50 χρόνια χρόνο ζωής, τα παράθυρα 30 χρόνια, τα ενεργειακά συστήματα 20 χρόνια, οι ηλεκτρικές συσκευές 10 χρόνια και τα συστήματα φωτισμού 4 χρόνια. Θεωρείται επιτόκιο προεξόφλησης 6%.

Υπολογίζονται τα κόστη αγοράς των μερών του κτιριακού κελύφους με βάση την εξίσωση (54).

$$COST_{com} = \sum_{n=1}^N A_n^{com} * \sum_{n=1}^N x_n^{com} * C_n^{com} \quad (54), \text{ όπου}$$

com=DOOR,WIN,WALL,CEILING,FLOOR

A_n^{com} : επιφάνεια του εκάστοτε στοιχείου

C_n^{com} : κόστος ανά τετραγωνικό μέτρο του εκάστοτε στοιχείου

x_n^{com} : μεταβλητή απόφασης του εκάστοτε στοιχείου (τιμή 1 αν επιλεγεί το στοιχείο ή τιμή 0 αν δεν επιλεγεί)

Υπολογίζονται τα κόστη αγοράς των συστημάτων με βάση την εξίσωση (55).

$$COST_{system} = \sum_{n=1}^N x_n^{system} * C_n^{system} \quad (55), \text{ όπου}$$

system=heating, cooling, dhw, heating-cooling, heating-dhw, solar, photovoltaic, lighting ή electrical appliance

C_n^{system} : κόστος του εκάστοτε συστήματος

x_n^{system} : μεταβλητή απόφασης του εκάστοτε συστήματος (τιμή 1 αν επιλεγεί το σύστημα ή τιμή 0 αν δεν επιλεγεί)

Το συνολικό κόστος επένδυσης υπολογίζεται αθροίζοντας τα επιμέρους κόστη.

$$\begin{aligned} INV_COST = & COST_{WIN} + COST_{DOOR} + COST_{FLOOR} + COST_{CEILING} + COST_{WALL} + \\ & COST_{LIGHT} + COST_{EHS} + COST_{EHCS} + COST_{EHWS} + COST_{NEHS} + \\ & COST_{NEHWS} + COST_{ECS} + COST_{EWS} + COST_{NEWS} + COST_{SOLAR} + COST_{PHOTO} + \\ & COST_{TV} + COST_{REFRIG} + COST_{COOKER} + COST_{WASHING_{ea}} \end{aligned} \quad (56)$$

Το κόστος κεφαλαίου κάθε χρόνο υπολογίζεται πολλαπλασιάζοντας τα επιμέρους κόστη με τον αντίστοιχο συντελεστή ανάκτησης κεφαλαίου.

$$\begin{aligned} CAP_COST = & (COST_{DOOR} + COST_{FLOOR} + COST_{CEILING} + COST_{WALL}) * af_{structure} + \\ & COST_{LIGHT} * af_{light} + (COST_{EHS} + COST_{EHCS} + COST_{EHWS} + COST_{NEHS} + \\ & COST_{NEHWS} + COST_{ECS} + COST_{EWS} + COST_{NEWS} + COST_{SOLAR} + COST_{PHOTO}) * af_{systems} + \\ & (COST_{TV} + COST_{REFRIG} + COST_{COOKER} + COST_{WASHING}) * af_{ea} + COST_{WIN} * af_{windows} \end{aligned} \quad (57)$$

Το λειτουργικό κόστος των ηλεκτρικών συστημάτων δίνεται από την παρακάτω εξίσωση :

$$OPER_COST_{el} = \left(\frac{1}{F_{conv}}\right) * cost_elec * Q_{el}^{grid} \quad (58)$$

$cost_elec$: κόστος αγοράς ηλεκτρικής ενέργειας από το δίκτυο

Το λειτουργικό κόστος των μη ηλεκτρικών συστημάτων δίνεται από την παρακάτω εξίσωση:

$$OPER_COST_{nel} = \sum_{i=1}^N \frac{y_i * cost_{fuel}}{efficiency_i * l_{hv_{fuel}}} \quad (59)$$

y_i : η βοηθητική μεταβλητή που παίρνει την τιμή της ενεργειακής κατανάλωσης του μη ηλεκτρικού συστήματος i ($i=NEHWS, NEWS, \dots$)

$efficiency_i$: βαθμός απόδοσης του μη ηλεκτρικού συστήματος i

$l_{hv_{fuel}}$: κατώτερη θερμογόνο δύναμη του καυσίμου $fuel$

$cost_{fuel}$: κόστος του καυσίμου $fuel$ ανά m^3

Το συνολικό ετήσιο λειτουργικό κόστος δίνεται από την εξίσωση (61) :

$$OPER_COST = OPER_COST_{el} + OPER_COST_{nel} \quad (61)$$

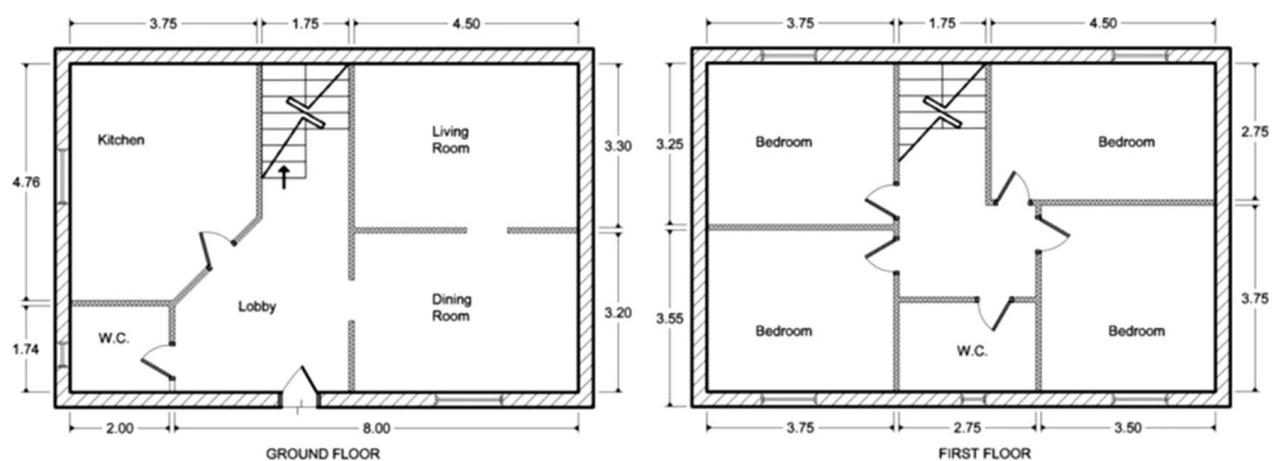
Τέλος, ορίζεται η τρίτη αντικειμενική συνάρτηση, το συνολικό ετήσιο κόστος. Υπολογίζεται αθροίζοντας τα 2 παραπάνω κόστη :

$$Total_Annual_Cost = CAP_COST + OPER_COST \quad (61)$$

ΚΕΦΑΛΑΙΟ 6 : ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ

6.1 Περιγραφή του κτιρίου και των συστημάτων

Το κτίριο της παρούσας μελέτης θα κτιστεί στην ευρύτερη περιοχή της Αθήνας. Είναι ένα κτίριο 2 ορόφων με εσωτερική σκάλα, που διαθέτει 4 δωμάτια σε κάθε όροφο. Στην Εικόνα 5 φαίνεται η κάτοψη των 2 ορόφων.



Εικόνα 5 Κάτοψη του κτιρίου[1]

Στον Πίνακα 1 δίνονται τα χαρακτηριστικά του κτιρίου :

Επιφάνεια εξωτερικών τοίχων (m ²)	194
Επιφάνεια εσωτερικών τοίχων (m ²)	99
Επιφάνεια ισογείου (m ²)	65
Επιφάνεια στέγης (m ²)	75
Επιφάνεια οροφής πρώτου ορόφου (m ²)	65
Επιφάνεια παραθύρων (m ²)	13
Επιφάνεια πόρτας (m ²)	3
Εσωτερικός όγκος (m ³)	344

Πίνακας 1 Χαρακτηριστικά κτιρίου

Στο Πίνακα 2 δίνονται τα κλιματολογικά χαρακτηριστικά της Αθήνας :

Μήνας	Εξωτερική Θερμοκρασία (Celsius)	Ημερήσια Ηλιακή Ακτινοβολία (kWh/m ² /day)	Σχετική Υγρασία (%)	Θερμοκρασία Νερού (Celsius)
1	7,4	1,39	69,5	11,3
2	7,8	1,91	64,4	10,9
3	10,8	2,78	56,7	11,8
4	15,8	3,85	47,4	14,3
5	21,5	5,01	39,9	17,7
6	26,4	5,27	34,5	21,6
7	28,6	4,93	33,9	24,7
8	28	4,62	36,5	25,4
9	24,2	3,93	41,6	24,2
10	18,9	2,49	51,5	21,1
11	13,1	1,54	63,7	16,9
12	8,7	1,22	71,2	13,5

Πίνακας 2 Κλιματολογικά χαρακτηριστικά της Αθήνας

Στο κτίριο θεωρείται ότι θα κατοικήσουν 4 άνθρωποι, που κατά μέσο όρο εκπέμπουν θερμότητα ίση με 115 W έκαστος. Ο αέρας θα πρέπει να ανανεώνεται 1,5 φορές ανά ώρα. Η ζήτηση σε ζεστό νερό χρήσης θερμοκρασίας 60 °C θεωρείται ίση με 60 λίτρα ανά μέρα. Η εσωτερική θερμοκρασία σχεδιασμού για την περίοδο ψύξης θεωρείται ίση με 26 °C και 18 °C για την περίοδο θέρμανσης. Θεωρήθηκε ότι απαιτείται θέρμανση, ψύξη και ζεστό νερό χρήσης όλους τους μήνες του χρόνου.

Το μοντέλο καλείται να επιλέξει πόρτα, παράθυρα, τοίχους, δάπεδο και οροφή από ένα σύνολο διαθέσιμων επιλογών. Είναι απαραίτητη η επιλογή μόνο ενός. Τα χαρακτηριστικά αυτών παρατίθενται στους Πίνακες 3, έως 7.

	U value (W/m ² *K)	Κόστος (Euro/m ²)
Door1	3.00	226
Door2	1.00	510
Door3	2.20	244

Πίνακας 3 Χαρακτηριστικά διαθέσιμων πορτών

	U value (W/m ² *K)	SHGC	Κόστος (Euro/m ²)
Window1	1,6	0,44	209
Window2	0,9	0,64	258
Window3	1,4	0,44	161

Πίνακας 4 Χαρακτηριστικά διαθέσιμων παραθύρων

	U value (W/m ² *K)	Κόστος (Euro/m ²)
Wall1	1,4731	90,25
Wall2	1,4203	140,5
Wall3	1,5295	100
Wall4	0,3388	99,09
Wall5	0,3359	149,34
Wall6	0,3417	108,84
Wall7	0,5587	97,21
Wall8	0,5509	147,46
Wall9	0,5666	106,96
Wall10	0,4926	95,2
Wall11	0,4865	145,45
Wall12	0,4987	104,95

Πίνακας 5 Χαρακτηριστικά διαθέσιμων τοίχων

	U value (W/m ² *K)	Κόστος (Euro/m ²)
Floor1	2,694	26,25
Floor2	2,6676	28,25
Floor3	2,7005	27,25
Floor4	0,3782	35,09
Floor5	0,3777	37,09
Floor6	0,3784	36,09
Floor7	0,6746	33,21
Floor8	0,673	35,21
Floor9	0,675	34,21
Floor10	0,5805	31,2
Floor11	0,5793	33,2
Floor12	0,5808	32,2

Πίνακας 6 Χαρακτηριστικά διαθέσιμων δαπέδων

	U value (W/m ² *K)	Κόστος (Euro/m ²)
Ceiling1	2,6901	20,95
Ceiling2	2,7079	20,5
Ceiling3	2,4296	44,25
Ceiling4	0,3781	29,79
Ceiling5	0,3785	29,34
Ceiling6	0,3725	53,09
Ceiling7	0,6744	27,91
Ceiling8	0,6755	27,46
Ceiling9	0,6567	51,21
Ceiling10	0,5804	25,9
Ceiling11	0,5812	25,45
Ceiling12	0,5672	49,2

Πίνακας 7 Χαρακτηριστικά διαθέσιμων οροφών

Το μοντέλο καλείται να επιλέξει ενεργειακά συστήματα (θέρμανσης, ψύξης, ZNX, κλπ) από ένα σύνολο διαθέσιμων επιλογών. Είναι απαραίτητη η επιλογή μόνο ενός συστήματος, εκτός από την περίπτωση του ηλιακού θερμοσίφωνα και του φωτοβολταϊκού που είναι προαιρετική η επιλογή. Τα χαρακτηριστικά αυτών παρατίθενται στους πίνακες 8, έως 22:

	Απόδοση(%)	Κόστος (Euro)
EHS1	100	1290
EHS2	98	1226
EHS3	95	1161

Πίνακας 8 Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης

	Απόδοση(%)	Κόστος (Euro)
ECS1	250	387
ECS2	300	516
ECS3	350	645

Πίνακας 9 Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων ψύξης

	Απόδοση(%)	Κόστος (Euro)
EHCS1	300	645
EHCS2	350	774
EHCS3	400	903

Πίνακας 10 Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης-ψύξης

	Απόδοση(%)	Κόστος (Euro)
EHWS1	100	1548
EHWS2	98	1277
EHWS3	400	3057

Πίνακας 11 Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων θέρμανσης-ZNX

	Απόδοση(%)	Κόστος (Euro)
EWS1	100	1169
EWS2	98	1096
EWS3	95	1032

Πίνακας 12 Χαρακτηριστικά διαθέσιμων ηλεκτρικών συστημάτων ZNX

	Απόδοση (%)	Επιφάνεια (m2)	Κόστος (Euro)
solar1	82,7	2,3	7224
solar2	64,2	2	5160
solar3	79,1	2,35	6966

Πίνακας 13 Χαρακτηριστικά διαθέσιμων ηλιακών θερμοσίφωνων

	Συνολική ισχύς	Απόδοση (%)	Επιφάνεια (m2)	PR (%)	Κόστος (Euro)
photov1	1000	15,65,	6,4	75	1548
photov2	1040	16,16	6,4	75	1585
photov3	1200	14,8	8,13	75	1817

Πίνακας 14 Χαρακτηριστικά διαθέσιμων φωτοβολταϊκών

	Αριθμός λαμπτήρων	Ημερήσια χρήση (h)	Ισχύς (W)	Κόστος (Euro)
lamp1	8	13	20	13
lamp2			24	6,5
lamp3			18	78

Πίνακας 15 Χαρακτηριστικά διαθέσιμων λαμπτήρων

	Ισχύς (W)	Ημερήσια χρήση (h)	Κόστος (Euro)
TV1	120	4	437
TV2	52	4	551
TV3	58	4	516

Πίνακας 16 Χαρακτηριστικά διαθέσιμων τηλεοράσεων

	Ισχύς (W)	Ημερήσια χρήση (h)	Κόστος (Euro)
cooker1	9530	1.50	645
cooker2	10475	1.50	516
cooker3	10000	1.50	580

Πίνακας 17 Χαρακτηριστικά διαθέσιμων ηλεκτρικών κουζινών

	Κατανάλωση ανά κύκλο (kWh)	Κύκλους της μέρα	Απόδοση ηλεκτρικού κινητήρα (%)	Κόστος (Euro)
washing1	1.05	0.50	70	387
washing2	0.73	0.50	70	516
washing3	0.79	0.50	70	495

Πίνακας 18 Χαρακτηριστικά διαθέσιμων πλυντηρίων

	Ημερήσια κατανάλωση	Απόδοση ηλεκτρικού κινητήρα (%)	Κόστος (Euro)
refrig1	0.51	70	445
refrig2	0.42	70	516
refrig3	0.50	70	477

Πίνακας 19 Χαρακτηριστικά διαθέσιμων ψυγείων

	Απόδοση(%)	Κόστος (Euro)
NEHS1	83	838
NEHS2	85	877
NEHS3	91	1019

Πίνακας 20 Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων θέρμανσης

	Απόδοση(%)	Κόστος (Euro)
NEHWS1	88	1032
NEHWS2	89	1096
NEHWS3	95	1225

Πίνακας 21 Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων θέρμανσης-ZNX

	Απόδοση(%)	Κόστος (Euro)
NEWS1	83	645
NEWS2	60	451
NEWS3	80	580

Πίνακας 22 Χαρακτηριστικά διαθέσιμων μη ηλεκτρικών συστημάτων ZNX

Το πρόβλημα λύθηκε με την γλώσσα μοντελοποίησης GAMS (General Algebraic Modeling System). Το πρόβλημα είναι Μικτού Ακέραιου Προγραμματισμού (MIP). Λύθηκαν 2 περιπτώσεις του προβλήματος, μία με τις αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους, της συνολικής πρωτογενούς κατανάλωσης ενέργειας και των εκπομπών διοξειδίου του άνθρακα, και μία με τις αντικειμενικές του συνολικού ετήσιου κόστους και της πρωτογενούς κατανάλωσης ενέργειας.

Στην περίπτωση των 3 αντικειμενικών συναρτήσεων, το πρόβλημα έχει 128 μεταβλητές, 224 εξισώσεις και λύθηκε σε 21,59 δευτερόλεπτα, χρησιμοποιώντας τον επιλύτη GUROBI. Στην περίπτωση των 2 αντικειμενικών συναρτήσεων, έχει 124 μεταβλητές, 219 εξισώσεις και λύθηκε 5,19 δευτερόλεπτα, χρησιμοποιώντας τον επιλύτη GUROBI. Εκτελέστηκε σε υπολογιστή με επεξεργαστή Intel Core 2 Quad Q6600 με 4GB Ram στα 64-bit.

6.2 Αποτελέσματα Περίπτωσης 3 αντικειμενικών συναρτήσεων

Τα αποτελέσματα από την εκτέλεση του κώδικα στην γλώσσα GAMS για τις αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους, της κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα παρατίθενται στους πίνακες 23, 24, 25.

TAC (Euro/year)	ENERGY (MJ/year)	EMISSIONS (kgCO ₂ /year)	TOT_CAPCOST (Euro/year)	TOT_OPERCOST (Euro/year)
2286,9	168518,25	17399,51	2025,42	261,48
2287,52	156366,73	16144,86	2044,89	242,63
2289,07	144211,35	14889,82	2065,3	223,77
2291,2	133036,31	13736	2084,77	206,43
2323,78	122016,36	12598,19	2134,45	189,33
2358,55	105872,25	10085,97	2045,36	313,19
2391,13	94852,29	8948,16	2095,04	296,09
2405,89	88189,15	8260,19	2120,14	285,75
2493,03	77221,94	7973,17	2373,21	119,82
2641,95	65950,37	6809,38	2539,62	102,33
4271,08	54582,48	5635,64	4186,39	84,69
2327,11	116915	12071,47	2145,7	181,41
2356,42	117047,29	11239,8	2025,89	330,53
2394,38	94736,06	8936,16	2098,47	295,91
2465,6	83106,2	7735,38	2187,74	277,86
2536,95	74666,7	7709,34	2421,09	115,86
2612,97	71279,11	6514,23	2353,46	259,51
2690,66	63213,65	6526,81	2592,57	98,09
4356,76	64194,35	5319,65	4026,67	330,09

Πίνακας 23 Αποτελέσματα προβλήματος με 3 αντικειμενικές

INVCOST (Euro)	QH _{el} (MJ/year)	QH _{nel} (MJ/year)	QC _{el} (MJ/year)	Qw _{el} (MJ/year)	Qw _{nel} (MJ/year)	Ql _{el} (MJ/year)	QA _{el} (MJ/year)
28660,25	23673,18	0	6886,3	16269,39	0	3279,74	8872,79
28967,15	19860,46	0	6445,98	16269,39	0	3279,74	8872,79
29288,9	15970,74	0	6081,32	16269,39	0	3279,74	8872,79
29595,8	12345,99	0	5794,8	16269,39	0	3279,74	8872,79
30427,1	8133,56	0	6150,25	16269,39	0	3279,74	8872,79
29143,8	12345,99	0	5794,8	0	19319,9	3279,74	8872,79
29975,1	8133,56	0	6150,25	0	19319,9	3279,74	8872,79
30152,1	7116,87	0	5381,47	0	19319,9	2733,12	8872,79
33073,13	6508,5	0	5996,03	3863,98	0	2733,12	7926,05
35106,31	6096,33	0	5992,96	3863,98	0	2733,12	8251,92
56183,97	5269,83	0	5831,63	1832,27	0	2459,81	7566,01
30556,1	7116,87	0	5381,47	16269,39	0	3279,74	8872,79
28836,9	15970,74	0	6081,32	0	19319,9	3279,74	8872,79
30029,1	8095,86	0	6147,27	0	19319,9	3279,74	8872,79
31070,31	6096,33	0	5243,84	0	19319,9	2733,12	8251,92
33827,79	5688,26	0	5921,93	3863,98	0	2733,12	7926,05
32912,31	6163,02	0	5219,22	0	19319,9	2733,12	7926,05
35769,79	5672,75	0	5902,82	3863,98	0	2733,12	7807,79
54351,97	0	22188,74	5831,63	0	7714,84	2459,81	7566,01

Πίνακας 24 Αποτελέσματα προβλήματος με 3 αντικειμενικές

CO2_HS (kgCO2/year)	CO2_WS (kgCO2/year)	CO2_GRID (kgCO2/year)
0	0	17399,51
0	0	16144,86
0	0	14889,82
0	0	13736
0	0	12598,19
0	1149,44	8936,53
0	1149,44	7798,72
0	1149,44	7110,75
0	0	7973,17
0	0	6809,38
0	0	5635,64
0	0	12071,47
0	1149,44	10090,35
0	1149,44	7786,72
0	1149,44	6585,94
0	0	7709,34
0	1149,44	5364,79
0	0	6526,81
1320,13	459	3540,52

Πίνακας 25 Αποτελέσματα προβλήματος με 3 αντικειμενικές

Οι επιλογές των συστημάτων για την περίπτωση των αντικειμενικών συναρτήσεων του συνολικού ετήσιου κόστους, της καταναλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα δίνεται στους πίνακες 26 έως 34.

DOOR1	DOOR2	DOOR3	WIND1	WIND2	WIND3	WALL1	WALL2	WALL3	WALL4	WALL5
1	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	1
1	0	0	0	0	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	1	0
1	0	0	0	0	1	0	0	0	0	0
0	0	1	0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0	0	0	1

Πίνακας 26 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

WALL6	WALL7	WALL8	WALL9	WALL10	WALL11	WALL12	CEIL1	CEIL2	CEIL3
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0

Πίνακας 27 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

CEIL4	CEIL5	CEIL6	CEIL7	CEIL8	CEIL9	CEIL10	CEIL11	CEIL12	FLOOR1
0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	0	0

Πίνακας 28 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

FLOOR2	FLOOR3	FLOOR4	FLOOR5	FLOOR6	FLOOR7	FLOOR8	FLOOR9	FLOOR10
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0

Πίνακας 29 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

FLOOR11	FLOOR12	EHS1	EHS2	EHS3	NEHS1	NEHS2	NEHS3	ECS1	ECS2	ECS3
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	1

Πίνακας 30 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

EWS1	EWS2	EWS3	NEWS1	NEWS2	NEWS3	EHCS1	EHCS2	EHCS3
0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	0	1
0	0	1	0	0	0	0	1	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	1	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	0	1
0	0	0	0	0	1	0	1	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Πίνακας 31 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

EHWS1	EHWS2	EHWS3	NEHWS1	NEHWS2	NEHWS3	SLC1	SLC2	SLC3	L1	L2	L3	TV1	TV2	TV3
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	1	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	1	1	0	0	0	0	1	0	1	0

Πίνακας 32 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

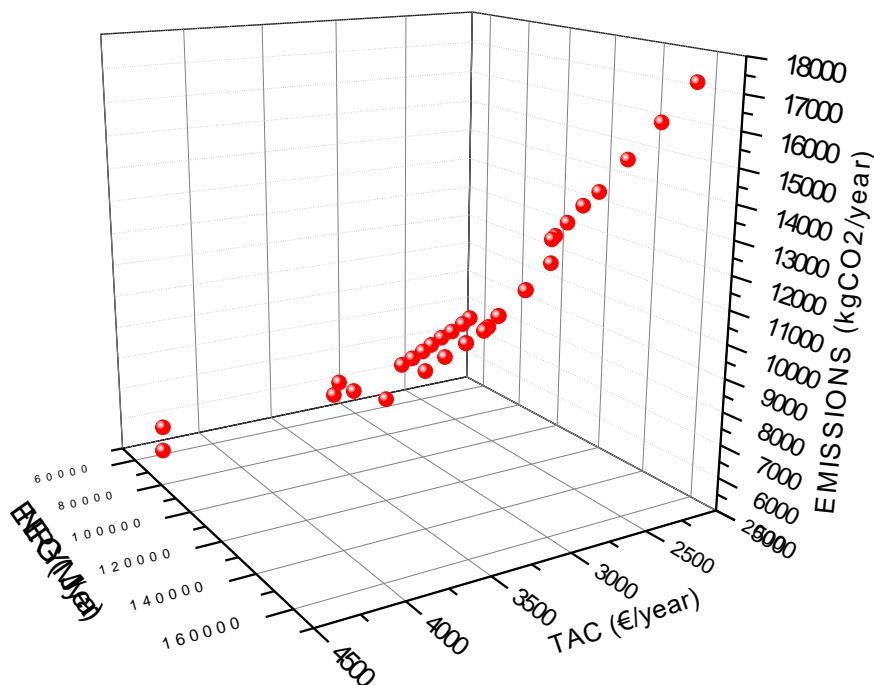
COOK1	COOK2	COOK3	WASH1	WASH2	WASH3	REFRIG1	REFRIG2	REFRIG3
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	0	1	0	0	1	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	0	1	0
1	0	0	0	1	0	0	1	0

Πίνακας 33 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

PHOTOV1	PHOTOV2	PHOTOV3
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	1
0	0	1
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	1
0	0	1
0	0	1

Πίνακας 34 Επιλογές συστημάτων προβλήματος με 3 αντικειμενικές

Στην Εικόνα 6 δίνεται το διάγραμμα με τις κατά Pareto ικανές λύσεις του προβλήματος με τις αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους, της κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα.



Εικόνα 6 Διάγραμμα Pareto με τις 3 αντικειμενικές συναρτήσεις

PAYOFF TABLE	TAC(Euro/year)	Primary Energy Consumption (MJ/year)	Emissions (kgCO2/year)
minTAC	2286,9	168518,25	17399,51
minEnergy	4271,08	54582,48	5635,64
minEmissions	4356,76	64194,35	5319,65

Πίνακας 35 Payoff Table- Πίνακας Πληρωμών προβλήματος με 3 αντικειμενικές

6.3 Αποτελέσματα Περίπτωσης 2 αντικειμενικών συναρτήσεων

Τα αποτελέσματα από την εκτέλεση του κώδικα στην γλώσσα GAMS για τις αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας παρατίθενται στους πίνακες 36 και 37.

TAC (Euro/year)	ENERGY (MJ/year)	TOT_CAPCOST (Euro/year)	TOT_OPERCOST (Euro/year)	INVCOST (Euro/year)	QHel (MJ/year)	QHnel (MJ/year)	QCel (MJ/year)
2286,9	168518,3	2025,42	261,48	28660,25	23673,18	0	6886,3
2287,52	156366,7	2044,89	242,63	28967,15	19860,46	0	6445,98
2289,07	144211,4	2065,3	223,77	29288,9	15970,74	0	6081,32
2291,2	133036,3	2084,77	206,43	29595,8	12345,99	0	5794,8
2318,31	128276,4	2119,26	199,04	29844,8	12378,27	0	5768,72
2323,78	122016,4	2134,45	189,33	30427,1	8133,56	0	6150,25
2327,11	116915	2145,7	181,41	30556,1	7116,87	0	5381,47
2358,55	105872,3	2045,36	313,19	29143,8	12345,99	0	5794,8
2361,82	105770,5	2048,79	313,03	29197,8	12313	0	5792,19
2391,13	94852,29	2095,04	296,09	29975,1	8133,56	0	6150,25
2394,46	89750,93	2106,29	288,17	30104,1	7116,87	0	5381,47
2405,89	88189,15	2120,14	285,75	30152,1	7116,87	0	5381,47
2436,9	82782,92	2308,45	128,45	32387,1	7149,14	0	6120,45
2493,03	77221,94	2373,21	119,82	33073,13	6508,5	0	5996,03
2575,09	71649,42	2463,91	111,18	34074,1	7116,15	0	6117,46
2641,95	65950,37	2539,62	102,33	35106,31	6096,33	0	5992,96
3117,86	60202,95	3024,45	93,41	40813,97	5342,78	0	5890,73
4271,08	54582,48	4186,39	84,69	56183,97	5269,83	0	5831,63

Πίνακας 36 Αποτελέσματα προβλήματος με 2 αντικειμενικές

Qwel (MJ/year)	Qwnel (MJ/year)	QLel (MJ/year)	QAel (MJ/year)
16269,39	0	3279,74	8872,79
16269,39	0	3279,74	8872,79
16269,39	0	3279,74	8872,79
16269,39	0	3279,74	8872,79
15455,92	0	2733,12	8560,71
16269,39	0	3279,74	8872,79
16269,39	0	3279,74	8872,79
0	19319,9	3279,74	8872,79
0	19319,9	3279,74	8872,79
0	19319,9	3279,74	8872,79
0	19319,9	3279,74	8872,79
0	19319,9	2733,12	8872,79
3863,98	0	3279,74	8560,71
3863,98	0	2733,12	7926,05
3863,98	0	2733,12	8560,71
3863,98	0	2733,12	8251,92
2492,49	0	2733,12	7926,05
1832,27	0	2459,81	7566,01

Πίνακας 37 Αποτελέσματα προβλήματος με 2 αντικειμενικές

Οι επιλογές των συστημάτων για την περίπτωση των αντικειμενικών συναρτήσεων του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας δίνονται στους πίνακες 38, έως 45.

DOOR1	DOOR2	DOOR3	WIND1	WIND2	WIND3	WALL1	WALL2	WALL3	WALL4	WALL5	WALL6
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	1	0	0	0	0	0
0	0	1	0	0	1	1	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0	0
0	0	1	0	0	1	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	0	0	0	1	0

Πίνακας 38 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

WALL7	WALL8	WALL9	WALL10	WALL11	WALL12	CEIL1	CEIL2	CEIL3	CEIL4	CEIL5	CEIL6
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	1

Πίνακας 39 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

CEIL7	CEIL8	CEIL9	CEIL10	CEIL11	CEIL12	FLOOR1	FLOOR2	FLOOR3	FLOOR4	FLOOR5	FLOOR6
0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0

Πίνακας 40 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

FLOOR7	FLOOR8	FLOOR9	FLOOR10	FLOOR11	FLOOR12	EHS1	EHS2	EHS3	NEHS1	NEHS2	NEHS3
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

Πίνακας 41 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

ECS1	ECS2	ECS3	EWS1	EWS2	EWS3	NEWS1	NEWS2	NEWS3	EHCS1	EHCS2	EHCS3
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	1
0	0	0	0	0	1	0	0	0	0	1	0
0	0	0	0	0	1	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	1	0	1	0
0	0	0	0	0	0	0	0	1	0	0	1
0	0	0	0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0

Πίνακας 42 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

EHWS1	EHWS2	EHWS3	NEHWS1	NEHWS2	NEHWS3	SLC1	SLC2	SLC3	L1	L2	L3	TV1	TV2	TV3
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
0	0	1	0	0	0	0	0	0	0	1	0	1	0	0
0	0	1	0	0	0	0	0	0	1	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	0	1	0	1	0	0	0	0	1
0	0	1	0	0	0	1	0	0	0	0	1	0	1	0

Πίνακας 43 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

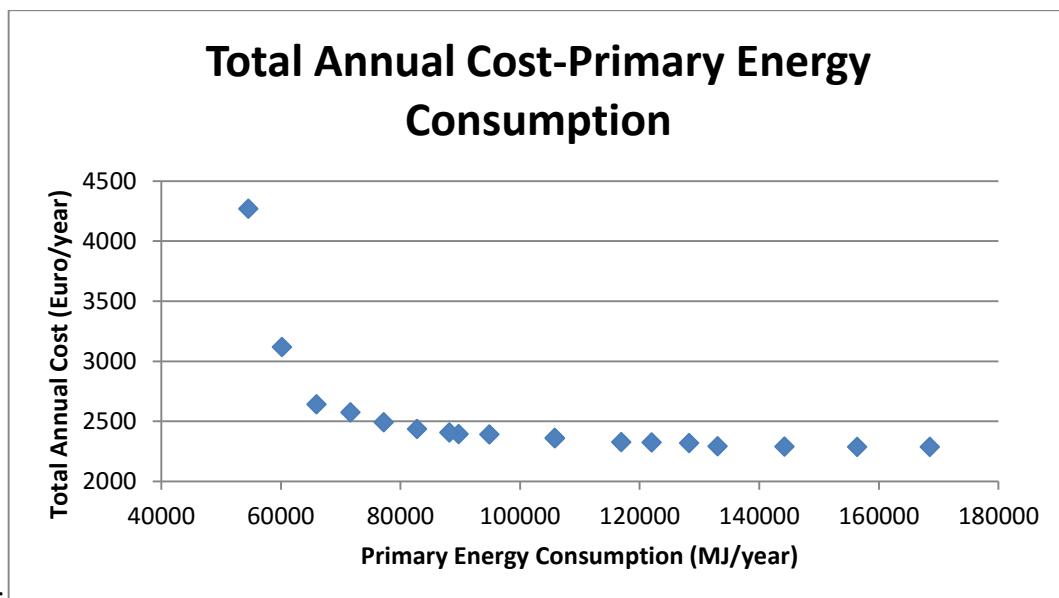
COOK1	COOK2	COOK3	WASH1	WASH2	WASH3	REFRIG1	REFRIG2	REFRIG3
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	0	1	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	1	0	1	0	0	1	0	0
0	0	1	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
0	0	1	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	1	0	0	1	0	0
1	0	0	0	1	0	0	1	0

Πίνακας 44 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

PHOTOV1	PHOTOV2	PHOTOV3
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	0
0	0	1
0	1	0
0	0	1

Πίνακας 45 Επιλογές συστημάτων προβλήματος με 2 αντικειμενικές

Το διάγραμμα με τις κατά Pareto ικανές λύσεις για την περίπτωση των αντικειμενικών συναρτήσεων του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας δίνεται στην Εικόνα 7.



Εικόνα 7 Διάγραμμα Pareto - Συνολικό ετήσιο κόστος - Συνολική κατανάλωση πρωτογενούς ενέργειας

PAYOFF TABLE	TAC	Energy
minTAC	2286,9	168518,3
minEnergy	4271,08	54582,48

Πίνακας 46 Payoff Table- Πίνακας Πληρωμών προβλήματος με 2 αντικειμενικές

Στον πίνακα πληρωμών (πίνακας 35) για την περίπτωση των 3 αντικειμενικών, βλέπουμε την τιμή της κάθε αντικειμενικής όταν ελαχιστοποιείται ανεξάρτητα. Η ελάχιστη τιμή της αντικειμενικής του ετήσιου κόστους είναι 2286,9 euro/year. Παρατηρείται ότι για αυτή την τιμή της αντικειμενικής του κόστους, η αντικειμενική της πρωτογενούς κατανάλωσης παίρνει την τιμή 168518,25 MJ/year και η αντικειμενική των εκπομπών CO₂ παίρνει την τιμή 17399,51 kgCO₂/year. Δηλαδή όταν ελαχιστοποιείται η αντικειμενική του κόστους, οι άλλες δύο αντικειμενικές μεγιστοποιούνται. Αυτό είναι λογικό καθότι επιλέγονται τα συστήματα EWS1 και EHCS3, τα οποία είναι ηλεκτρικά, άρα έχουν περισσότερες εκπομπές διοξειδίου του άνθρακα. Επιλέγεται η πόρτα Door1, ο τοίχος Wall1, η οροφή Ceil2, το δάπεδο Floor1, δηλαδή μέρη του κτιριακού κελύφους που έχουν τη υψηλότερη τιμή U, καθότι δεν έχουν μόνωση. Δεν επιλέγεται κανένα μη ηλεκτρικό σύστημα θέρμανσης ή παροχής ζεστού νερού χρήσης. Επίσης δεν επιλέγεται κάποιο φωτοβολταϊκό πάνελ ή ηλιακός θερμοσίφωνας καθώς θα αυξανότανε το κόστος. Όταν η αντικειμενική της κατανάλωσης πρωτογενούς ενέργειας ελαχιστοποιείται, παίρνει την τιμή 54582,48 MJ/year. Για αυτήν την τιμή, οι αντικειμενικές των εκπομπών CO₂ και του κόστους παίρνουν τις τιμές 5635,64 kgCO₂/year και 4271,08 euro/year αντίστοιχα. Όμως η τιμή 4271,08 euro/year δεν είναι η μέγιστη τιμή της αντικειμενικής του κόστους, όπως θα περίμενε κανείς. Αυτό συμβαίνει διότι δεν επιλέγεται ξεχωριστό σύστημα για την θέρμανση του χώρου και ξεχωριστό για την θέρμανση νερού, αλλά επιλέγεται το σύστημα EHWS3 που παρέχει και τα 2 και έχει την μεγαλύτερη απόδοση από τις υπόλοιπες επιλογές, και έτσι μειώνεται το συνολικό κόστος. Παρόλο που η αντικειμενική της ενέργειας ελαχιστοποιείται δεν συμβαίνει το ίδιο και με την αντικειμενική των εκπομπών. Αυτό συμβαίνει διότι επιλέγονται συστήματα ηλεκτρικά, τα οποία έχουν όμως μεγαλύτερο αποτύπωμα άνθρακα από τα αντίστοιχα μη ηλεκτρικά. Τέλος στην περίπτωση ελαχιστοποίησης της αντικειμενικής των εκπομπών CO₂, η ελάχιστη τιμή της βρίσκεται ίση με 5319,65 kgCO₂/year. Παρατηρείται ότι για αυτήν την τιμή η αντικειμενική του κόστους μεγιστοποιείται, ενώ η αντικειμενική της ενέργειας όχι. Το κόστος μεγιστοποιείται διότι τα συστήματα που έχουν το ελάχιστο ανθρακικό αποτύπωμα έχουν το υψηλότερο κόστος. Επιλέγεται το μη ηλεκτρικό σύστημα NEHWS3 για την θέρμανση του χώρου και του νερού και αυτό οδηγεί στην ελαχιστοποίηση της ηλεκτρικής ενέργειας που αγοράζεται από το δίκτυο παροχής ηλεκτρικής ενέργειας, άρα και μείωση των εκπομπών CO₂

που εκλύονται. Με βάση τα παραπάνω αποτελέσματα οδηγούμαστε στο συμπέρασμα ότι κάθε προσπάθεια μείωσης της κατανάλωσης πρωτογενούς ενέργειας ή των εκπομπών διοξειδίου θα αυξήσει το συνολικό ετήσιο κόστος. Αν επιθυμούμε την μείωση των εκπομπών CO₂, πρέπει να στραφούμε σε μη ηλεκτρικά συστήματα, κυρίως φυσικού αερίου, αλλά και να προβούμε στην εγκατάσταση φωτοβολταϊκού πάνελ και ηλιακού θερμοσίφωνα. Επίσης παρατηρείται ότι δεν επιλέγεται σε καμία περίπτωση ηλεκτρικό σύστημα θέρμανσης, αλλά συστήματα που συνδυάζουν θέρμανση με ψύξη ή θέρμανση νερού. Λύνοντας το παραπάνω πρόβλημα με τις αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας, παρατηρούμε από τον πίνακα πληρωμών (πίνακας 46) ότι οι ελάχιστες τιμές των αντικειμενικών της ενέργειας και του κόστους είναι ίδιες με αυτές της περίπτωσης των 3 αντικειμενικών. Αυτό συμβαίνει διότι, όλες οι παράμετροι και οι περιορισμοί είναι ίδιοι. Η διαφορά που παρατηρείται είναι ότι στην περίπτωση των 2 αντικειμενικών δεν επιλέγεται σε καμία περίπτωση μη ηλεκτρικό σύστημα θέρμανσης χώρου αλλά το μόνο μη ηλεκτρικό σύστημα που επιλέγεται είναι το σύστημα NEWS3 για την θέρμανση του νερού. Πράγμα λογικό διότι σκοπός της επίλυσης είναι η ελαχιστοποίηση του κόστους και της πρωτογενούς κατανάλωσης και όχι οι εκπομπές CO₂.

ΚΕΦΑΛΑΙΟ 7: ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία μελετήθηκε το πρόβλημα του ενεργειακού σχεδιασμού κτιρίου με χρήση Μικτού Ακέραιου Γραμμικού Προγραμματισμού. Συγκεκριμένα, μελετήθηκε η περίπτωση ενός νέου κτιρίου στην περιοχή της Αθήνας, σε 2 περιπτώσεις. Μια περίπτωση με σκοπό την ελαχιστοποίηση της αντικειμενικής συνάρτησης του συνολικού ετήσιου κόστους, της κατανάλωσης πρωτογενούς ενέργειας και των εκπομπών διοξειδίου του άνθρακα, και μία με σκοπό την ελαχιστοποίηση της αντικειμενικής συνάρτησης του συνολικού ετήσιου κόστους και της κατανάλωσης πρωτογενούς ενέργειας. Και οι 2 περιπτώσεις επιλύθηκαν σε γλώσσα GAMS με την μέθοδο AUGMECON2. Από την επίλυση των προβλήματων καταλήγουμε στο συμπέρασμα ότι οι αντικειμενικές συναρτήσεις του συνολικού κόστους και της κατανάλωσης πρωτογενούς ενέργειας είναι αλληλοσυγκρουόμενες. Δηλαδή η μείωση της κατανάλωσης πρωτογενούς ενέργειας οδηγεί στην αύξηση του συνολικού ετήσιου κόστους. Όταν ελαχιστοποιείται η κατανάλωση ενέργειας το συνολικό ετήσιο κόστος γίνεται μέγιστο. Επιλέγονται μέρη του κτιριακού κελύφους με μόνωση, καθώς και το ηλεκτρικό σύστημα ψύξης ECS3, το οποίο έχει την μεγαλύτερη απόδοση από τις υπόλοιπες επιλογές. Για θέρμανση χώρου και νερού επιλέγεται το ηλεκτρικό σύστημα EHWS3, το οποίο έχει την μεγαλύτερη απόδοση από τις υπόλοιπες επιλογές. Όταν ελαχιστοποιείται το συνολικό ετήσιο κόστος, μεγιστοποιείται η κατανάλωση πρωτογενούς ενέργειας και οι εκπομπές διοξειδίου του άνθρακα. Σε αυτήν την περίπτωση επιλέγονται τα συστήματα που έχουν το χαμηλότερο κόστος, τα οποία όμως είναι ηλεκτρικά και έχουν μεγαλύτερο ανθρακικό αποτύπωμα. Συμπεραίνουμε ότι οι αντικειμενικές συναρτήσεις του συνολικού ετήσιου κόστους και των εκπομπών διοξειδίου του άνθρακα είναι αλληλοσυγκρουόμενες. Όταν ελαχιστοποιείται η αντικειμενική των εκπομπών διοξειδίου του άνθρακα, μεγιστοποιείται το συνολικό ετήσιο κόστος, καθότι επιλέγονται μέρη του κτιριακού κελύφους με μόνωση, και συστήματα θέρμανσης χώρου και νερού μη ηλεκτρικά επειδή έχουν μικρότερο αποτύπωμα άνθρακα. Επίσης σε αυτήν την περίπτωση επιλέγεται ηλιακός θερμοσίφωνας και φωτοβολταικά. Τα φωτοβολταικά δεν παράγουν διοξείδιο του άνθρακα, για αυτό και επιλέγονται στην περίπτωση ελαχιστοποίησης των εκπομπών. Παρατηρείται ότι δεν επιλέγεται σε καμία περίπτωση σύστημα θέρμανσης που παρέχει αποκλειστικά θέρμανση, αλλά σε κάθε περίπτωση επιλέγεται σύστημα θέρμανσης που συνδυάζει θέρμανση με ψύξη ή θέρμανση του νερού. Επίσης επιλέγονται μέρη του κτιριακού κελύφους με μόνωση, όταν το κριτήριο της μείωσης της κατανάλωσης πρωτογενούς ενέργειας ή των εκπομπών διοξειδίου γίνεται πιο σημαντικό. Ανακεφαλαιώνοντας, συμπεραίνουμε ότι οποιαδήποτε προσπάθεια μείωσης

της κατανάλωσης πρωτογενούς ενέργειας ή των εκπομπών διοξειδίου, αυξάνει το συνολικό ετήσιο κόστος. Η επιλογή μη ηλεκτρικών συστημάτων για θέρμανση του χώρου και του νερού μειώνει τις εκπομπές διοξειδίου, αλλά αυξάνει το κόστος. Επίσης η επιλογή μερών του κτιριακού κελύφους με μόνωση, μειώνει την κατανάλωση πρωτογενούς ενέργειας και των εκπομπών διοξειδίου, αλλά αυξάνει το συνολικό ετήσιο κόστος.

Προτάσεις για το μέλλον

Η βελτίωση της ενεργειακής απόδοσης των κτιρίων είναι ανάμεσα στις πρώτες προτεραιότητες της παγκόσμιας ενεργειακής πολιτικής. Για αυτό πρέπει να γίνουν προσπάθειες για να την περαιτέρω βελτίωση της. Αυτό μπορεί να επιτευχθεί με την χρήση συστημάτων συμπαραγωγής ενέργειας, καθώς και με την χρήση μικρο-ανεμογεννητριών. Στην παρούσα εργασία έγινε η παραδοχή ότι οι τιμές των καυσίμων είναι σταθερές. Προφανώς αυτό δεν ισχύει, καθότι η αγορά των καυσίμων χαρακτηρίζεται από αβεβαιότητα. Στοχαστική ανάλυση με την μέθοδο monte carlo μπορεί να δώσει ορθότερα αποτελέσματα. Επίσης έγινε η παραδοχή ότι απαιτείται θέρμανση, ψύξη και ζεστό νερό χρήσης όλους τους μήνες του χρόνου. Μια προσέγγιση που θα λάμβανε υπόψιν την μεταβολή της ζήτησης της ενέργειας θα έδινε πιο ακριβή αποτελέσματα.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] M. Karmellos, A. Kiprakis, and G. Mavrotas, "A multi-objective approach for optimal prioritization of energy efficiency measures in buildings: Model, software and case studies," *Appl. Energy*, vol. 139, pp. 131–150, 2015.
- [2] C. Diakaki, E. Grigoroudis, N. Kabelis, D. Kolokotsa, K. Kalaitzakis, and G. Stavrakakis, "A multi-objective decision model for the improvement of energy efficiency in buildings," *Energy*, vol. 35, no. 12, pp. 5483–5496, 2010.
- [3] E. Asadi, M. G. Da Silva, C. H. Antunes, and L. Dias, "Multi-objective optimization for building retrofit strategies: A model and an application," *Energy Build.*, vol. 44, no. 1, pp. 81–87, 2012.
- [4] D. Kolokotsa, D. Rovas, E. Kosmatopoulos, and K. Kalaitzakis, "A roadmap towards intelligent net zero- and positive-energy buildings," *Sol. Energy*, vol. 85, no. 12, pp. 3067–3084, 2011.
- [5] E. M. Malatji, J. Zhang, and X. Xia, "A multiple objective optimisation model for building energy efficiency investment decision," *Energy Build.*, vol. 61, pp. 81–87, 2013.
- [6] G. Mavrotas, D. Diakoulaki, K. Florios, and P. Georgiou, "A mathematical programming framework for energy planning in services' sector buildings under uncertainty in load demand: The case of a hospital in Athens," *Energy Policy*, vol. 36, no. 7, pp. 2415–2429, 2008.
- [7] G. Mavrotas, K. Florios, and D. Vlachou, "Energy planning of a hospital using Mathematical Programming and Monte Carlo simulation for dealing with uncertainty in the economic parameters," *Energy Convers. Manag.*, vol. 51, no. 4, pp. 722–731, 2010.
- [8] Σπυρίδων Αλιβιζάτος, "Εφαρμογές Πολυκριτηριακού Μαθηματικού Προγραμματισμού με τη χρήση της γλώσσας μοντελοποίησης GAMS ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ," 2011.
- [9] S. S. Rao, *Engineering Optimization*, vol. 56, no. 772801201. 2009.
- [10] Γ. Μαυρωτάς, "Διδακτορική Διατριβή: Πολυκριτηριακός Προγραμματισμός σε Συνθήκες Αβεβαιότητας- Κατασκευή Συστήματος Υποστήριξης Αποφάσεων και Εφαρμογή στον Ενεργειακό Σχεδιασμό," 2000.
- [11] G. Mavrotas and K. Florios, "An improved version of the augmented s-constraint method (AUGMECON2) for finding the exact pareto set in multi-objective integer

programming problems,” *Appl. Math. Comput.*, vol. 219, no. 18, pp. 9652–9669, 2013.

[12] Γ. Μ. Κοντορούπης, *Ενεργειακός – Βιοκλιματικός Σχεδιασμός Κτιρίων και Οικισμών*. Αθήνα, 1998.

[13] Α. Μανωλίτσης, “ΕΠΙΔΡΑΣΗ ΚΑΙΝΟΤΟΜΩΝ ΜΟΝΩΤΙΚΩΝ ΥΛΙΚΩΝ ΣΤΗΝ ΕΝΕΡΓΕΙΑΚΗ ΑΠΟΔΟΣΗ ΚΤΗΡΙΑΚΟΥ ΚΕΛΥΦΟΥΣ ΜΕ ΧΡΗΣΗ ΤΟΥ ΥΠΟΛΟΓΙΣΤΙΚΟΥ ΠΑΚΕΤΟΥ EnergyPlus,” 2015.

[14] J. A. Çengel, “Chapter 16: Heating and Cooling of Buildings,” *Heat Mass Transf. A Pract. Approach*, 3/e, pp. 1–86, 2013.

[15] L. D. D. Harvey, *Energy Efficiency and the Demand for Energy Services*. 2010.

ΠΑΡΑΡΤΗΜΑ

1. Κώδικας GAMS με 3 αντικειμενικές συναρτήσεις

Παρακάτω παρατίθενται ο κώδικας που χρησιμοποιήθηκε για την επίλυση του προβλήματος με τις 3 αντικειμενικές συναρτήσεις σε γλώσσα GAMS :

```

$eolcom //

* to make the model linear sets of walls, floors and ceilings
have 12 components
* componets 1:3 are without insulation, 4:6 with insulation 1,
7:9 insulation 2, 10:12 insulation 3

Sets
  d          door type
/door1*door3/
  wz         window type
/window1*window3/
  w          wall structure
/wall1*wall12/
  r          structure of ceiling
/ceiling1*ceiling12/
  h          floor structure
/floor1*floor12/
  tv         television           /tv1*tv3/
  cooker     cooker
/cooker1*cooker3/
  washing    washing machine
/wash1*wash3/
  refrig     refrigerator
/refrig1*refrig3/
  pvs        photovoltiacs       /pvs1*pvs3/
  ehsi       electrical heating system /EHS1*EHS3/
  nehsi      non electrical heating system
/NEHS1*NEHS3/
  ecsci      electrical cooling system /ECS1*ECS3/
  ewsi       electrical DHW        /EWS1*EWS3/
  newsi      non electric DHW
/NEWS1*NEWS3/
  ehcsi      electrical heating-cooling
/EHCS1*EHCS3/
  ehws      electrical heating DHW
/EHWS1*EHWS3/
  nehws      non electrical heating DHW
/NEHWS1*NEHWS3/
  slci       solar collector category
/solar1*solar3/
  lj         type of lamps
/lamp1*lamp3/
  month      month                 /1*12/

```

```

hour          period of day          /1*24/

k objective functions /1*3/
;

Parameter dir(k) direction of the objective functions 1 for
max and -1 for min
  / 1  -1
    2  -1
    3  -1
  /

Parameters
To(month)    Average external temperature for each month
(Celsius)
/
1      7.4
2      7.8
3     10.8
4     15.8
5     21.5
6     26.4
7     28.6
8      28
9     24.2
10    18.9
11    13.1
12    8.7
/

td(month)    month duration in days
/
1         31
2         28
3         31
4         30
5         31
6         30
7         31
8         31
9         30
10        31
11        30
12        31
/

Isl(month)   solar radiation kWh * m^2*day)^-1
/
1     1.39
2     1.91
3     2.78
4     3.85
5     5.01
6     5.27
7     4.93

```

8 4.62
9 3.93
10 2.49
11 1.54
12 1.22
/

tm(month) month duration in hours

/
1 744
2 672
3 744
4 720
5 744
6 720
7 744
8 744
9 720
10 744
11 720
12 744
/

Tdcw(month) temperature of cold water inlet to the DHW system
(Celsius)

/
1 11.3
2 10.9
3 11.8
4 14.3
5 17.7
6 21.6
7 24.7
8 25.4
9 24.2
10 21.1
11 16.9
12 13.5
/

WSm(month) indicator for DHW demand each month

/
1 1
2 1
3 1
4 1
5 1
6 1
7 1
8 1
9 1
10 1
11 1
12 1
/

CS(month) indicator for cooling demand each month

/

1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1

/

HS(month) indicator for heating demand each month

/

1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1

/

Wo(month) specific humidity outdoors each month (kgwater per kgair)

/

1	0.00400
2	0.00420
3	0.00450
4	0.00520
5	0.00630
6	0.00730
7	0.00820
8	0.00850
9	0.00770
10	0.00690
11	0.00590
12	0.00495

/

Udoors(d) heat transfer coefficient for doors $W \cdot m^{-2} \cdot K$

/

door1	2.04
door2	0.86
door3	1.64

/

Costdoor(d) Cost for doors per square meter

/
door1 226
door2 510
door3 244
/

UWindows(wz) Heat transfer coefficient for windows

/
window1 1.28
window2 0.79
window3 1.15
/

CostWindows(wz) cost for windows per square meter

/
window1 209
window2 258
window3 161
/

Gwin(wz) Solar heat gain coefficient

/
window1 0.44
window2 0.64
window3 0.44
/

Uwalls(w) heat transfer coefficient of walls

/
wall1 1.4731
wall2 1.4203
wall3 1.5295
wall4 0.3388
wall5 0.3359
wall6 0.3417
wall7 0.5587
wall8 0.5509
wall9 0.5666
wall10 0.4926
wall11 0.4865
wall12 0.4987
/

CostWalls(w) cost of walls euros per m2

/
wall1 90.25
wall2 140.50
wall3 100.00
wall4 99.09
wall5 149.34
wall6 108.84
wall7 97.21
wall8 147.46
wall9 106.96

wall10 95.20
wall11 145.45
wall12 104.95
/

UFloor(h) heat transfer coefficient of floors

/
floor1 2.6940
floor2 2.6676
floor3 2.7005
floor4 0.3782
floor5 0.3777
floor6 0.3784
floor7 0.6746
floor8 0.6730
floor9 0.6750
floor10 0.5805
floor11 0.5793
floor12 0.5808
/

CostFloors(h) cost of floors euros per m2

/
floor1 26.25
floor2 28.25
floor3 27.25
floor4 35.09
floor5 37.09
floor6 36.09
floor7 33.21
floor8 35.21
floor9 34.21
floor10 31.20
floor11 33.20
floor12 32.20
/

UCeiling(r) heat transfer coefficient of ceilings

/
ceiling1 2.6901
ceiling2 2.7079
ceiling3 2.4296
ceiling4 0.3781
ceiling5 0.3785
ceiling6 0.3725
ceiling7 0.6744
ceiling8 0.6755
ceiling9 0.6567
ceiling10 0.5804
ceiling11 0.5812
ceiling12 0.5672
/

CostCeiling(r) cost of ceilings euros per m2

/

```
ceiling1 20.95
ceiling2 20.50
ceiling3 44.25
ceiling4 29.79
ceiling5 29.34
ceiling6 53.09
ceiling7 27.91
ceiling8 27.46
ceiling9 51.21
ceiling10 25.90
ceiling11 25.45
ceiling12 49.20
/
```

```
LampsPower(lj) power of lamps
/
lamp1 20
lamp2 24
lamp3 18
/
```

```
CostLamps(lj) cost of lamps
/
lamp1 13
lamp2 7
lamp3 77
/
```

```
EfficiencyElecHeatSys(ehsi) efficiency of electrical heating
systems
/
EHS1 1
EHS2 0.98
EHS3 0.9
/
```

```
CostElecHeatSys(ehsi) cost of electrical heating
systems
/
EHS1 1290
EHS2 1226
EHS3 1161
/
```

```
EfficiencyNonElecHeatSys(nehsi) efficiency of non-
electrical heating systems
/
NEHS1 0.83
NEHS2 0.85
NEHS3 0.91
/
```

```
* For non elec systems the lhv of respective fuel is used
* for oil is 34560 MJ per m3 and for gas =32.4 MJ per m3
lhvnonelecheatsys(nehsi) low heating value of non elec heat
sys
```

```
/
NEHS1 34560
NEHS2 32.4
NEHS3 32.4
/
```

```
CostNonElecHeatSys(nehsi) cost of non- electrical heating
systems
```

```
/
NEHS1 839
NEHS2 680
NEHS3 1019
/
```

```
EfficiencyElecCoolingSys(ecsi) efficiency of electrical
cooling systems
```

```
/
ECS1 2.5
ECS2 3
ECS3 3.5
/
```

```
CostElecCoolingSys(ecsi) cost of electrical cooling systems
```

```
/
ECS1 387
ECS2 516
ECS3 645
/
```

```
EfficiencyElecHeatCoolingSys(ehcsi) efficiency of electrical
heating cooling systems
```

```
/
EHCS1 3
EHCS2 3.5
EHCS3 4
/
```

```
CostElecHeatCoolingSys(ehcsi) cost of electrical heating
cooling systems
```

```
/
EHCS1 645
EHCS2 774
EHCS3 903
/
```

```
EfficiencySolar(slci) efficiency of solar collectors
```

```
/
solar1 0.827
solar2 0.642
solar3 0.791
/
```

```
AreaSolar(slci) area of solar collectors
```

```
/
```



```
solar1 2.3
solar2 2
solar3 2.35
/
```

```
CostSolar(slci) cost of solar collectors
```

```
/
solar1 7224
solar2 5160
solar3 6966
/
```

```
EfficiencyElecDHW(ewsi) efficiency of electrical DHW
```

```
/
EWS1 1
EWS2 0.98
EWS3 0.95
/
```

```
CostElecDHW(ewsi) cost of electrical DHW
```

```
/
EWS1 1169
EWS2 1096
EWS3 1032
/
```

```
EfficiencyNonElecDHW(newsi) efficiency of non-electrical DHW
```

```
/
NEWS1 0.83
NEWS2 0.6
NEWS3 0.8
/
```

```
CostNonElecDHW(newsi) cost of non-electrical DHW
```

```
/
NEWS1 645
NEWS2 451
NEWS3 580
/
```

```
lhvNonElecDHW(newsi) fuel of non-electrical DHW
```

```
/
NEWS1 34560
NEWS2 34560
NEWS3 32.4
/
```

```
EfficiencyElecHeatingDHW(ehws) efficiency of electrical heating DHW
```

```
/
EHWS1 1
EHWS2 0.98
EHWS3 4
/
```

```
CostElecHeatingDHW(ehws) cost of electrical heating DHW
```

/
EHWS1 1548
EHWS2 1277
EHWS3 3057
/

EfficiencyNonElecHeatingDHW(nehwsi) efficiency of non-electrical heating DHW

/
NEHWS1 0.88
NEHWS2 0.89
NEHWS3 0.95
/

lhvNonElecHeatingDHW(nehwsi) fuel of non-electrical heating DHW

/
NEHWS1 34560
NEHWS2 32.4
NEHWS3 32.4
/

CostNonElecHeatingDHW(nehwsi) cost of non-electrical heating DHW

/
NEHWS1 1032
NEHWS2 1096
NEHWS3 1225
/

PowerTV(tv) energy consumption of televisions W

/
tv1 120
tv2 52
tv3 58
/

CostTV(tv) costs of tv

/
tv1 437
tv2 551
tv3 516
/

PowerCooker(cooker) consumption of cookers W

/
cooker1 9530
cooker2 10475
cooker3 10000
/

CostCooker(cooker) cost of cookers

/
cooker1 645
cooker2 516

cooker3 580
/

PowerWashing(washing) consumption per cycle for washing
machines KWh

/
wash1 1.05
wash2 0.73
wash3 0.79
/

EfficiencyWashing(washing) efficiency of washing machines
motors efficiency (%)

/
wash1 0.7
wash2 0.7
wash3 0.7
/

CostWashing(washing) cost of washing machines

/
wash1 387
wash2 516
wash3 495
/

PowerRefrig(refrig) daily consumption of refrigerators KWh

/
refrig1 0.51
refrig2 0.42
refrig3 0.50
/

EfficiencyRefrig(refrig) efficiency of refrigerators motors

/
refrig1 0.7
refrig2 0.7
refrig3 0.7
/

CostRefrig(refrig) cost of refrigerators

/
refrig1 445
refrig2 516
refrig3 477
/

GenerationEfficiencyPhotoV(pvs) generation efficiency of
photovoltaics systems(%)

/
pvs1 0.1565
pvs2 0.1616
pvs3 0.1480
/

```

AreaPhotoV(pvs) area of photovoltaics systems m^2
/
pvs1  6.4
pvs2  6.4
pvs3  8.13
/

```

```

CostPhotoV(pvs) cost of photovoltaics
/
pvs1  1548
pvs2  1585
pvs3  1817
/

```

```

annfactor_systems annuity of energy systems
annfactor_structure annuity of structure components
annfactor_win annuity of windows
annfactor_lamps annuity of lamps
annfactor_ea annuity of electrical appliances
;

```

```

Scalars
ACH          air change per hour (h^-1)
/1.5/
Awin         area of windows (m^2)
/13/
Fconv       Conversion Factor MJ*kWh^-1
/3.6/
Adoor       area of doors m^2
/3/
Awallint    area of internal walls m^2
/99/
Awallext    area of external walls
/194/
Aceil       area of ceilings m^
/62/
Afloor      area of floors m^2
/127/
ShadingFactorWindows Windows Shading Factors
/1/
UsageLamps  dialy usage of lamps hours
/13/
Nlamps      number of lamps
/8/
UsageTV     daily usage of tv hours
/4/
UsageCooker dialy usage of cooker
/0.5/
ShadingFactorSolar shading factor of solar
collectors      /1/
CPair        specific heat of air at
constant pressure kj*(kg*k)^-1      /1.0035/

```

CPwater	specific heat of water at
constant pressure $\text{kJ} \cdot (\text{kg} \cdot \text{K})^{-1}$	/4.18/
FCm	window correction factor for
movable devices (%)	/1/
Ff	frame factor for windows (%)
/1/	
dair	air density ($\text{kg} \cdot \text{m}^{-3}$)
/1.2/	
dwater	water density ($\text{kg} \cdot \text{m}^{-3}$)
/1000/	
ho	outdoors combined convection-
radiation coefficient $\text{W} \cdot (\text{m}^2 \cdot \text{K})^{-1}$	/28/
Mw	daily need of hot water (m^3
per second)	/0.00000278/
Npeople	number of people
/4/	
Hfg	latent heat of vaporization
(kJ per kg)	/2340/
wi	specific humidity indoors
(kg water per kg air)	/0.00834/
ngrid	efficiency of the electricity
supply grid	/0.35/
SEAel	assuming no losses from
supply to consumption	/1/
Tih	internal design temperature
for heating season (Celsius)	/18/
Tic	internal design temperature
for cooling season (Celsius)	/26/
Tdhw	temperature for dhw
/60/	
V	internal volume of the
building (m^3)	/344/
PRphotov	PR Value for photovoltaics
/0.75/	
Qhuman	heat emitted from people (W)
/115/	
CyclesWashing	usage cycles per day for
washing machines	/0.5/
mneg	negative number for if
statement	/-1000000/
mpos	positive number for if
statement	/1000000/
life_systems	life for energy systems (years) /20/
life_structure	life for walls ceils and floors /50/
life_win	life for windows /30/
life_lamps	life for lamps /4/
life_ea	life for electrical appliances /10/
rate	rate for annualized cost /0.06/
cost_oil	cost of oil in euros per m^3 /740/
cost_ng	cost of natural gas in euros per m^3 /0.3/
cost_elec	electricity cost when buying from grid per kWh
/0.01596/	
*lhv_oil	low heating value of oil kWh per m^3 (for boilers)
/9600/	

```

*lhv_gas low heating value of gas kwh per m3 (assume 0.55 of
air_density) /9/
elec_sale_price price for electricity sales per kWh /0.02/

*from diakaki
oil_co2 oil emission factor kg co2 per kg of oil /3.142/
ng_co2 ng emission factor kg co2 per kg of ng /2.715/
grid_co2 elec emission factor kg co2 per MJ /0.295/

*from springer energy and other
oil_dens density of oil in kg per m3 (assumed) /800/
ng_dens density of oil in kg per m3 assumed 0.55 of air /0.71/
;

```

Binary Variables

```

XDOOR(d)
XWIN(wz)
XWALL(w)
XCEIL(r)
XFLOOR(h)
XEHS(ehsi)
XNEHS(nehsi)
XECS(ecsi)
XEWS(ewsi)
XNEWS(newsi)
XEHCS(ehcsi)
XEHWS(ehws)
XNEHWS(nehws)
XSLC(slci)
XL(lj)
XTV(tv)
XCOOKER(cooker)
XWASHING(washing)
XREFRIG(refrig)
XPHOTOV(pvs)
;

```

positive Variables

```

Qc          annual primary energy consumption for
cooling (Mj*year^-1)
Qh          annual primary energy consumption for
heating (Mj*year^-1)
Qcel       annual primary energy consumption for
cooling consumed by an electrical system (Mj*year^-1)
Qdhw       annual primary energy consumption for DHW
(Mj*year^-1)
Qwel       annual primary energy consumption for DHW
by a electrical system (Mj*year^-1)
Qownel     annual primary energy consumption for DHW
by a non-electrical system (Mj*year^-1)

Qhel       annual primary energy consumption for
heating from a electrical system (Mj*year^-1)

```

Qlel	energy consumption for lighting	Mj per year
QDel	annual demand for electricity	Mj per year
QCDtot	annual cooling demand	(Mj*year ⁻¹)
QHDtot	annual heating demand	(Mj*year ⁻¹)
QLDtot	annual energy demand for lighting	(Mj*year ⁻¹)
QWDtot	annual energy demand for DHW	(Mj*year ⁻¹)
QEaDTot	total annual energy demand for electrical appliances	(Mj*year ⁻¹)
QAel	annual energy consumption for electrical appliances	(Mj*year ⁻¹)
QCD(month)	cooling demand each month	MJ
QHD(month)	heating demand each month	MJ
QSL(month)	solar heat gain each month	(kWh)
QeaD(month)	energy for electrical appliances each month	MJ
DQdhw(month)	net DHW demand	MJ
QLD(month)	energy demand for lighting each month	MJ
QdSLC(month)	DHW demand provided by a solar collector	kWh
QPhoto(month)	energy produced by a photovoltaic cell	kWh
Qelgrid	energy provided by the grid	MJ per year
BLC	Building Load Factor	(W*K ⁻¹)
QINHG(month)	internal heat gain each month	kWh
QTV(month)	energy demand for TV each month	kWh
QCooker(month)	energy demand for cooker each month	kWh
Qwash(month)	energy demand for washing machine each month	kWh
Qrefrig(month)	energy demand for refrigerator each month	kWh
QHel	energy consumption of electrical systems	kWh
QHnel	energy consumption of non-electrical systems	kWh
pvsupply	photovoltaic supply	kWh
Q_TV_HEAT(month)	heat from tv	kWh
Q_COOKER_HEAT(month)	heat from cooker	kWh
Q_WASH_HEAT(month)	heat from washing machine	kWh
Q_REFRIG_HEAT(month)	heat from refrigerator	kWh
QEA_HEAT(month)	heat from electric appliances	kWh
COSTfordoor	Investment cost for doors	
COSTforwindows	Investment cost for windows	
COSTforwalls	Investment cost for walls	

COSTforceilings	Investment cost for ceilings
COSTforfloors	Investment cost for floors
COSTforheatingsystem systems	Investment cost for heating
COSTforcoolingsystem systems	Investment cost for cooling
COSTforwatersystem	Investment cost for dhw systems
COSTforheatingcooling cooling systems	Investment cost for heating-
COSTforheatingwatersystem dhw systems	Investment cost for for heating
COSTforsolar collectors	Investment cost for solar
COSTforlight	Investment cost for lighting
COSTforPhotov photovoltaics	Investment cost for
COSTforTV	Investment cost for TVs
COSTforcooker	Investment cost for cookers
COSTforWashing machines	Investment cost for washing
COSTforRefrig refrigerators	Investment cost for
TOT_OPERCOST	total operational cost
INVCOST	investmenet cost
TOT_CAPCOST	total capital cost
FUEL_DHW_NEL_COST	cost for fuel water systems
FUEL_HEAT_NEL_COST	cost for fuel heat systems
OPERCOST_FUEL	operational cost for fuel
OPERCOST_EL	operational cost for electricity
Qalt	total alternative energy
generation in MJ	
Q_pe_sys_el	total primary energy consumption
of electrical systems	
Q_pe_sys_nel	total primary energy consumption
of non electrical systems	
CO2_HEAT_NEL	emissions from non electrical
heating systems	
CO2_DHW_NEL	emissions from non electrical
DHW systems	
CO2_GRID	emissions from grid
y1	variable for linearization
y2	variable for linearization
y3	variable for linearization
y4	variable for linearization
y5	variable for linearization
y6	variable for linearization
y7	variable for linearization
y8	variable for linearization


```

y9          variable for linearization
y10         variable for linearization
y11         variable for linearization
y12         variable for linearization
y13         variable for linearization
y14         variable for linearization
y15         variable for linearization
y16         variable for linearization
y17         variable for linearization
y18         variable for linearization
y19         variable for linearization
y20         variable for linearization
y21         variable for linearization
y22         variable for linearization
y23         variable for linearization
y24         variable for linearization
y25         variable for linearization
y26         variable for linearization
y27         variable for linearization
y28         variable for linearization
y29         variable for linearization
y30         variable for linearization
y31         variable for linearization
y32         variable for linearization
y33         variable for linearization
;

```

Free Variables

```

Qdhwu(month)    DHW demand    kWh
QTClosses(month)  heat transmittance losses each month
heating season kWh
QTHlosses(month)  heat transmittance losses each month
cooling season kWh
QCVEN(month)     ventilation losses each month cooling
season (kWh)
QHVEN(month)     ventilation losses each month heating
season (kWh)

```

```

Z (K)           objective functions
TAC             total annual cost
Qt             total primary energy consumption
TOT_CO2        total emissions
;

```

Equations

```

*-----Orismos eksisoseon-----
-----

```

```

doors          Decision equation
for door
windows        Decision equation
for window
walls          Decision equation
for walls
ceilings       Decision equation
for ceilings

```

floors	Decision equation
for floors	
Photov	Decision equation
for photovoltaics	
TVs	Decision equation
for TVs	
Cooker_eq	Decision equation
for cookers	
Washing_eq	Decision equation
for washing machines	
Refrig_eq	Decision equation
for refrigerators	
lamps	Decision equation
for lamps	
spaceheating	Decision equation
for spaceheating	
spacecooling	Decision equation
for spacecooling	
dhwsystems	Decision equation
for DHW systems	
solarcollector	Decision equation
for Solar Collectors	
COSTfordoor_eq	Equation of
investment cost for doors	
COSTforwindows_eq	Equation of
Investment cost for windows	
COSTforwalls_eq	Equation of
Investment cost for walls	
COSTforceilings_eq	Equation of
Investment cost for ceilings	
COSTforfloors_eq	Equation of
Investment cost for floors	
COSTforheatingsystem_eq	Equation of
Investment cost for heating systems	
COSTforcoolingsystem_eq	Equation of
Investment cost for cooling systems	
COSTforwatersystem_eq	Equation of
Investment cost for dhw systems	
COSTforheatingcooling_eq	Equation of
Investment cost for heating-cooling systems	
COSTforheatingwatersystem_eq	Equation of
Investment cost for for heating dhw systems	
COSTforsolar_eq	Equation of
Investment cost for solar collectors	
COSTforlight_eq	Equation of
Investment cost for lighting	
COSTforPhotov_eq	Equation of
Investment cost for photovoltaics	
COSTforTV_eq	Equation of
Investment cost for TVs	
COSTforcooker_eq	Equation of
Investment cost for cookers	

COSTforWashing_eq		Equation of
Investment cost for washing machines		
COSTforRefrig_eq		Equation of
Investment cost for refrigerators		
BLC_eq		Equation of BLC
value of a building		

*-----TECHNICAL PARAMETERS -----

QTClosses_eq(month) cooling season	Equation of	monthly Transimission
QCVEN_eq(month) cooling season	Equation of	ventilation losses
QTHlosses_eq(month) heating season	Equation of	monthly Transimission
QHVEN_eq(month) heating season	Equation of	ventilation losses
QCDtot_eq kWh	Equation of	annual cooling demand
QHDtot_eq kWh	Equation of	annual heating demand
QINHG_eq(month)	Equation of	internal heat gain
QSL_eq(month)	Equation of	solar gains each month
QTV_eq(month) each month kWh	Equation of	energy demand for TV
QCooker_eq(month) each month kWh	Equation of	energy demand for cooker
Qwash_eq(month) washing machine each month kWh	Equation of	energy demand for
Qrefrig_eq(month) refrigerator each month kWh	Equation of	energy demand for
Q_tv_heat_eq(month)		
Q_cooker_heat_eq(month)		
Q_wash_heat_eq(month)		
Q_refrig_eq(month)		
Qea_heat_eq(month)		
QLD_eq(month) lighting each month kWh	Equation of	energy demand for
QLDtot_eq lighting Mj per year	Equation of	annual energy demand for
QLe1_eq lighting Mj per year	Equation of	energy consumption for
QWDtot_eq DHW (Mj*year ⁻¹)	Equation of	annual energy demand for
QPhoto_eq(month) photovoltaic cell per month	Equation of	energy generation by a
Qalt_eq of a photovoltaic cell	Equation of	total energy generation
QEaDTot_eq electrical appliances	Equation of	annual energy demand for

```

QAel_eq           Equation of  annual energy
consumption for electrical appliances

altenergy_eq      Equation of  power supply from
alternative energy generation systems

QdSLC_eq(month)   Equation of  DHW demand provided by a
solar collector kWh
Qdhwu_eq(month)   Equation of  DHW demand
Qea_eq(month)     Equation of  energy for electrical
appliaces each month kWh

QDel_eq          Equation of  annual demand for
electricity

ifequation1(month)   if equation 1
ifequation2(month)   if equation 2
ifequation3(month)   if equation 3
ifequation4          if equation 4
*-----linearization-----
-----
-----

linear_eq1         linearization equation
linear_eq2         linearization equation
linear_eq3         linearization equation
linear_eq4         linearization equation
linear_eq5         linearization equation
linear_eq6         linearization equation
linear_eq7         linearization equation
linear_eq8         linearization equation
linear_eq9         linearization equation
linear_eq10        linearization equation
linear_eq11        linearization equation
linear_eq12        linearization equation
linear_eq13        linearization equation
linear_eq14        linearization equation
linear_eq15        linearization equation
linear_eq16        linearization equation
linear_eq17        linearization equation
linear_eq18        linearization equation
linear_eq19        linearization equation
linear_eq20        linearization equation
linear_eq20        linearization equation
linear_eq21        linearization equation
linear_eq22        linearization equation
linear_eq23        linearization equation
linear_eq24        linearization equation
linear_eq25        linearization equation
linear_eq26        linearization equation
linear_eq27        linearization equation
linear_eq28        linearization equation

```

linear_eq29	linearization equation
linear_eq30	linearization equation
linear_eq31	linearization equation
linear_eq32	linearization equation
linear_eq33	linearization equation
linear_eq34	linearization equation
linear_eq35	linearization equation
linear_eq36	linearization equation

Qhel_eq Equation of annual primary energy
consumption for heating from a electrical system (Mj*year⁻¹)

linear_eq37	linearization equation
linear_eq38	linearization equation
linear_eq39	linearization equation
linear_eq40	linearization equation
linear_eq41	linearization equation
linear_eq42	linearization equation
linear_eq43	linearization equation
linear_eq44	linearization equation
linear_eq45	linearization equation
linear_eq46	linearization equation
linear_eq47	linearization equation
linear_eq48	linearization equation
linear_eq49	linearization equation
linear_eq50	linearization equation
linear_eq51	linearization equation
linear_eq52	linearization equation
linear_eq53	linearization equation
linear_eq54	linearization equation
linear_eq55	linearization equation
linear_eq56	linearization equation
linear_eq57	linearization equation
linear_eq58	linearization equation
linear_eq59	linearization equation
linear_eq60	linearization equation

QHnel_eq Equation of energy consumption of
non-electrical systems kWh

linear_eq61	linearization equation
linear_eq62	linearization equation
linear_eq63	linearization equation
linear_eq64	linearization equation
linear_eq65	linearization equation
linear_eq66	linearization equation
linear_eq67	linearization equation
linear_eq68	linearization equation
linear_eq69	linearization equation
linear_eq70	linearization equation
linear_eq71	linearization equation
linear_eq72	linearization equation
linear_eq73	linearization equation
linear_eq74	linearization equation
linear_eq75	linearization equation
linear_eq76	linearization equation

linear_eq77	linearization equation
linear_eq78	linearization equation
linear_eq79	linearization equation
linear_eq80	linearization equation
linear_eq81	linearization equation
linear_eq82	linearization equation
linear_eq83	linearization equation
linear_eq84	linearization equation

Qcel_eq Equation of annual primary energy
consumption for cooling consumed by an electrical system
(Mj*year⁻¹)

linear_eq85	linearization equation
linear_eq86	linearization equation
linear_eq87	linearization equation
linear_eq88	linearization equation
linear_eq89	linearization equation
linear_eq90	linearization equation
linear_eq91	linearization equation
linear_eq92	linearization equation
linear_eq93	linearization equation
linear_eq94	linearization equation
linear_eq95	linearization equation
linear_eq96	linearization equation
linear_eq97	linearization equation
linear_eq98	linearization equation
linear_eq99	linearization equation
linear_eq100	linearization equation
linear_eq101	linearization equation
linear_eq102	linearization equation
linear_eq103	linearization equation
linear_eq104	linearization equation
linear_eq105	linearization equation
linear_eq106	linearization equation
linear_eq107	linearization equation
linear_eq108	linearization equation

Qwel_eq Equation of annual primary energy
consumption for DHW by a electrical system (Mj*year⁻¹)

linear_eq109	linearization equation
linear_eq110	linearization equation
linear_eq111	linearization equation
linear_eq112	linearization equation
linear_eq113	linearization equation
linear_eq114	linearization equation
linear_eq115	linearization equation
linear_eq116	linearization equation
linear_eq117	linearization equation
linear_eq118	linearization equation
linear_eq119	linearization equation
linear_eq120	linearization equation
linear_eq121	linearization equation
linear_eq122	linearization equation
linear_eq123	linearization equation

```

linear_eq124          linearization equation
linear_eq125          linearization equation
linear_eq126          linearization equation
linear_eq127          linearization equation
linear_eq128          linearization equation
linear_eq129          linearization equation
linear_eq130          linearization equation
linear_eq131          linearization equation
linear_eq132          linearization equation

```

```

Qownel_eq            Equation of annual primary energy
consumption for DHW by a non-electrical system (Mj*year^-1)

```

```

*-----if statements-----
-----

```

```

invcost_eq          investment cost
capcost_eq          capital cost (annuity)
tot_opercost_eq     annual operational cost
opercost_fuel_eq    operational cost of fuel based systems
opercost_el_eq      operational cost electricity
tot_ann_cost_eq     total annual cost

```

```

fuel_heat_nel_cost_eq  cost for fuel based heating systems
fuel_dhw_nel_cost_eq   cost for fuel based water systems

```

```

co2_heat_nel_cost_eq  co2 emissions from fuel heating sys
co2_dhw_nel_cost_eq   co2 emissions from fuel dhw sys
co2_grid_eq           co2 emissions from grid
tot_co2_eq            total co2 emissions

```

```

Q_pe_sys_el_eq       Equation of annual primary
consumption from systems consuming electricity
Q_pe_sys_nel_eq      Equation of annual primary
consumption from systems consuming fuels
obj2_energy          objective function for energy
consumption
obj1                 definition of Z1
obj2                 definition of Z2
obj3                 definition of Z3

```

```

;
```

```

doors..             sum(d,XDOOR(d))=e=1 ;
windows..           sum(wz,XWIN(wz))=e=1;
walls..             sum(w,XWALL(w))=e=1 ;
ceilings..          sum(r,XCEIL(r))=e=1;
floors..            sum(h, XFLOOR(h))=e=1 ;
TVs..               sum(tv,XTV(tv))=e=1;
Cooker_eq..         sum(cooker,XCOOKER(cooker))=e=1;
Washing_eq..        sum(washing,XWASHING(washing))=e=1;
Refrig_eq..         sum(refrig,XREFRIG(refrig))=e=1;
lamps..             sum((lj), XL(lj)) =e=1 ;

```

```

spaceheating.. sum(ehsi, XEHS(ehsi)) + sum(nehsi,
XNEHS(nehsi)) + sum(ehcsi, XEHCS(ehcsi)) +
sum(ehwsii, XEHWS(ehwsii)) + sum(nehwsii, XNEHWS(nehwsii))=e=1 ;
spacecooling.. sum(ecsi, XECS(ecsi)) + sum(ehcsi,
XEHCS(ehcsi)) =e= 1 ;
dhwsystems.. sum(ewsi, XEWS(ewsi)) + sum(newsi,
XNEWS(newsi)) + sum(ehwsii, XEHWS(ehwsii)) + sum(nehwsii,
XNEHWS(nehwsii)) =e= 1 ;

solarcollector.. sum(slci, XSLC(slci)) =l= 1 ;
Photov.. sum(pvs, XPHOTOV(pvs)) =l= 1;

```

```

*-----OBJECTIVE 1: COST-----
-----

```

```

COSTfordoor_eq..
COSTfordoor=e=Adoor*sum(d, XDOOR(d) * Costdoor(d)) ;
COSTforwindows_eq..
COSTforwindows=e=Awin*sum(wz, XWIN(wz) *CostWindows(wz)) ;
COSTforwalls_eq.. COSTforwalls=e=Awallext*sum(w,
XWALL(w) *CostWalls(w)) ;
COSTforceilings_eq..
COSTforceilings=e=Aceil*sum(r, XCEIL(r) *CostCeiling(r)) ;
COSTforfloors_eq..
COSTforfloors=e=Afloor*sum(h, XFLOOR(h) *CostFloors(h)) ;
COSTforheatingsystem_eq..
COSTforheatingsystem=e=sum(ehsi, XEHS(ehsi) *CostElecHeatSys(eh
si))+sum(nehsi, XNEHS(nehsi) * CostNonElecHeatSys(nehsi)) ;
COSTforcoolingsystem_eq..
COSTforcoolingsystem=e=sum(ecsi, XECS(ecsi) *CostElecCoolingSys(
ecsi)) ;
COSTforwatersystem_eq..
COSTforwatersystem=e=sum(ewsi, XEWS(ewsi) *CostElecDHW(ewsi)) +su
m(newsi, XNEWS(newsi) *CostNonElecDHW(newsi)) ;
COSTforheatingcooling_eq..
COSTforheatingcooling=e=sum(ehcsi, XEHCS(ehcsi) *CostElecHeatCoo
lingSys(ehcsi)) ;
COSTforheatingwatersystem_eq..
COSTforheatingwatersystem=e=sum(ehwsii, XEHWS(ehwsii) *CostElecHea
tingDHW(ehwsii)) +sum(nehwsii, XNEHWS(nehwsii) *CostNonElecHeatingDH
W(nehwsii)) ;
COSTforsolar_eq..
COSTforsolar=e=sum(slci, XSLC(slci) * CostSolar(slci));
COSTforlight_eq..
COSTforlight=e=sum(lj, XL(lj) *CostLamps(lj)) *Nlamps ;
COSTforPhotov_eq..
COSTforPhotov=e=sum(pvs, XPHOTOV(pvs) *CostPhotoV(pvs)) ;
COSTforTV_eq..
COSTforTV=e=sum(tv, XTV(tv) *CostTV(tv)) ;
COSTforcooker_eq..
COSTforcooker=e=sum(cooker, XCOOKER(cooker) *CostCooker(cooker))
;

```



```

COSTforWashing_eq..
COSTforWashing=e=sum(washing,XWASHING(washing)*
CostWashing(washing));
COSTforRefrig_eq..
COSTforRefrig=e=sum(refrig,XREFRIG(refrig)*CostRefrig(refrig)
;

invcost_eq..          INVCOST =e= COSTforlight +
COSTforwindows + (COSTfordoor + COSTforwalls +
COSTforceilings + COSTforfloors)
                    + (COSTforheatingsystem +
COSTforcoolingsystem + COSTforwatersystem +
COSTforheatingcooling + COSTforheatingwatersystem +
COSTforsolar + COSTforPhotov)
                    + (COSTforTV + COSTforcooker
+ COSTforWashing + COSTforRefrig);

capcost_eq..          TOT_CAPCOST =e=
COSTforlight*annfactor_lamps + COSTforwindows*annfactor_win +
(COSTfordoor + COSTforwalls + COSTforceilings +
COSTforfloors)*annfactor_structure
                    + (COSTforheatingsystem +
COSTforcoolingsystem + COSTforwatersystem +
COSTforheatingcooling + COSTforheatingwatersystem +
COSTforsolar + COSTforPhotov)*annfactor_systems
                    + (COSTforTV + COSTforcooker
+ COSTforWashing + COSTforRefrig)*annfactor_ea;

tot_opercost_eq..    TOT_OPERCOST =e= OPERCOST_FUEL +
OPERCOST_EL;
opercost_fuel_eq..   OPERCOST_FUEL =e= FUEL_HEAT_NEL_COST +
FUEL_DHW_NEL_COST;
opercost_el_eq..     OPERCOST_EL =e= 1/fconv*cost_elec*Qelgrid;

tot_ann_cost_eq..    TAC =e= TOT_CAPCOST + TOT_OPERCOST;

obj1..              Z('1')=e= TAC;
*----OBJECTIVE 2: ENERGY CONSUMPTION-----*

BLC_eq..            BLC =e=
Adoor*sum(d,XDOOR(d)*Udoors(d))+Awin*sum(wz,XWIN(wz)*
UWindows(wz))+Awallext*sum(w,XWALL(w)*Uwalls(w))+
Aceil*sum(r,XCEIL(r)*UCeiling(r))+
Afloor*sum(h,XFLOOR(h)*UFloor(h));

QTClosses_eq(month).. QTClosses(month)=e=BLC*(Tic-
To(month))*tm(month)/1000 ; //kWh

QCVEN_eq(month)..   QCVEN(month)=e=dair*CPair*ACH*V*(Tic-
To(month))*tm(month)/3600 + dair*Hfg*ACH*V*(wi-
Wo(month))*tm(month)/3600; //kWh

QTHlosses_eq(month).. QTHlosses(month)=e=BLC*(Tih-
To(month))*tm(month)/1000; //kWh
QHVEN_eq(month)..   QHVEN(month)=e= dair*CPair*ACH*V*(Tih-
To(month))*tm(month)/3600 ; //kWh

```

```

QINHG_eq(month)..
QINHG(month)=e=(Npeople*Qhuman/1000)*tm(month)+QEA_HEAT(month)
; //kWh

QSL_eq(month)..          QSL(month)=e=sum(wz, XWIN(wz)*Gwin(wz))
* Awin*Ff*ShadingFactorWindows*FCm*Isl(month)*td(month)
; //kWh

QTV_eq(month)..          QTV(month)=e=sum(tv,
PowerTV(tv)*XTV(tv))*UsageTV *td(month)/1000 ; //kWh
QCooker_eq(month)..
QCooker(month)=e=sum(cooker, PowerCooker(cooker)*XCOOKER(cooker
))*UsageCooker*td(month)/1000 ; //kWh
Qwash_eq(month)..          Qwash(month)=e=td(month)*CyclesWashing
*sum(washing, PowerWashing(washing))*XWASHING(washing) ; //kWh
Qrefrig_eq(month)..
Qrefrig(month)=e=td(month)*sum(refrig, PowerRefrig(refrig)*XREF
RIG(refrig)) ; //kWh
Qea_eq(month)..          QeaD(month)=e=Fconv*(QTV(month)+
QCooker(month)+Qwash(month)+Qrefrig(month)) ; //MJ

Q_tv_heat_eq(month)..    Q_TV_HEAT(month) =e= QTV(month);
//kWh
Q_cooker_heat_eq(month).. Q_COOKER_HEAT(month) =e=
QCooker(month); //kWh
Q_wash_heat_eq(month)..  Q_WASH_HEAT(month) =e=
td(month)*CyclesWashing
*sum(washing, PowerWashing(washing))*XWASHING(washing)/Efficienc
yWashing(washing); //kWh
Q_refrig_eq(month)..     Q_REFRIG_HEAT(month) =e=
td(month)*sum(refrig, PowerRefrig(refrig)*XREFRIG(refrig)/Effic
iencyRefrig(refrig)) ; //kWh
Qea_heat_eq(month)..     QEA_HEAT(month) =e=
Q_TV_HEAT(month) + Q_COOKER_HEAT(month) + Q_WASH_HEAT(month) +
Q_REFRIG_HEAT(month); //kWh , converted to MJ in another eq

QLD_eq(month)..
QLD(month)=e=Fconv*td(month)*Nlamps*UsageLamps*sum(lj,
LampsPower(lj)* XL(lj))/1000 ; //MJ

*-----PV and SLC ANNUAL GENERATION-----*
QPhoto_eq(month)..       QPhoto(month) =e= sum(pvs,
XPHOTOV(pvs))*
AreaPhotoV(pvs)*GenerationEfficiencyPhotoV(pvs))*PRphotov*Shad
ingFactorSolar*Isl(month)*td(month); // kWh
Qalt_eq..                 pvsupply=e=sum(month, QPhoto(month)) ;
//kWh
altenergy_eq..           Qalt =e= pvsupply*Fconv; //MJ

QdSLC_eq(month)..        QdSLC(month)=e=ShadingFactorSolar
*Isl(month)*td(month)*sum(slci, AreaSolar(slci)*XSLC(slci)*
EfficiencySolar (slci)); //kWh
Qdhwu_eq(month)..        Qdhwu(month)=e=Mw*dwater*CPwater*(Tdhw-
Tdcw(month))*tm(month) ; //kWh

```

----IF EQUATIONS----

```
ifequation1(month)..          QHD(month) =g=  
HS(month)*Fconv*(QTHlosses(month)+ QHVEN(month)-QINHG(month)-  
QSL(month)); //MJ  
ifequation2(month)..          QCD(month) =g=  
CS(month)*Fconv*(QINHG(month)+QSL(month)- QTClosses(month)-  
QCVEN(month)); //MJ  
ifequation3(month)..          DQdhw(month) =g=  
WSm(month)*Fconv*(Qdhwu(month)-QdSLC(month)); //MJ
```

*-----ENERGY -----
-----*

```
QHDtot_eq..          QHDtot =e= sum(month,QHD(month)) ;  
// MJ  
QCDtot_eq..          QCDtot =e= sum(month,QCD(month)) ; //  
MJ  
QLDtot_eq..          QLDtot =e= sum(month,QLD(month)); // MJ  
QEaDTot_eq..        QEaDTot =e= sum(month,QeaD(month)); //  
MJ  
QWDtot_eq..          QWDtot =e= sum(month, DQdhw(month)) ;  
// MJ  
  
QDel_eq..            QDel=e=QHel+QCel+Qwel+QLel+QAel ;  
//MJ, electricity demand  
ifequation4..        Qelgrid =g= QDel - Qalt; //MJ
```

```
QHel_eq..            QHel=e=(y1/EfficiencyElecHeatSys('EHS1'))+  
(y2/EfficiencyElecHeatSys('EHS2'))+  
(y3/EfficiencyElecHeatSys('EHS3')) +  
  
(y4/EfficiencyElecHeatCoolingSys('EHCS1'))+  
(y5/EfficiencyElecHeatCoolingSys('EHCS2'))  
+(y6/EfficiencyElecHeatCoolingSys('EHCS3')) +  
  
(y7/EfficiencyElecHeatingDHW('EHWS1'))+  
(y8/EfficiencyElecHeatingDHW('EHWS2')) +  
(y9/EfficiencyElecHeatingDHW('EHWS3')); //MJ
```

```
QHnel_eq..           QHnel=e=(y10/EfficiencyNonElecHeatSys('NEHS1'))+(y11/Efficienc  
yNonElecHeatSys('NEHS2'))+  
(y12/EfficiencyNonElecHeatSys('NEHS3'))+  
  
(y13/EfficiencyNonElecHeatingDHW('NEHWS1'))+  
(y14/EfficiencyNonElecHeatingDHW('NEHWS2')) +  
(y15/EfficiencyNonElecHeatingDHW('NEHWS3')); //MJ
```

```
fuel_heat_nel_cost_eq..      FUEL_HEAT_NEL_COST =e=  
y10*cost_oil/(EfficiencyNonElecHeatSys('NEHS1')*lhvNonElecHeat
```

Sys('NEHS1'))+y11*cost_ng/(EfficiencyNonElecHeatSys('NEHS2')*1
 hvNonElecHeatSys('NEHS2'))+

y12*cost_ng/(EfficiencyNonElecHeatSys('NEHS3')*1hvNonElecHeats
 Sys('NEHS3'))+

y13*cost_oil/(EfficiencyNonElecHeatingDHW('NEHWS1')*1hvNonElec
 HeatingDHW('NEHWS1'))

+

y14*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS2')*1hvNonElecH
 eatingDHW('NEHWS2')) +

y15*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS3')*1hvNonElecH
 eatingDHW('NEHWS3')); //euros (based on fuel consumption per
 m3)

QCel_eq.. QCel=e=
 (y16/EfficiencyElecCoolingSys('ECS1'))
 +(y17/EfficiencyElecCoolingSys('ECS2'))+
 (y18/EfficiencyElecCoolingSys('ECS3'))
 +
 (y19/EfficiencyElecHeatCoolingSys('EHCS1')) +
 (y20/EfficiencyElecHeatCoolingSys('EHCS2')) +
 (y21/EfficiencyElecHeatCoolingSys('EHCS3')); //MJ

Qwel_eq.. Qwel=e=(y22/EfficiencyElecDHW('EWS1')) +
 (y23/EfficiencyElecDHW('EWS2'))+ (y24/EfficiencyElecDHW('EWS3'))
 +

(y25/EfficiencyElecHeatingDHW('EHWS1')) +
 (y26/EfficiencyElecHeatingDHW('EHWS2'))+
 (y27/EfficiencyElecHeatingDHW('EHWS3')); //MJ

Qwnel_eq.. Qwnel=e=(y28/EfficiencyNonElecDHW
 ('NEWS1')) + (y29/EfficiencyNonElecDHW('NEWS2')) +
 (y30/EfficiencyNonElecDHW('NEWS3'))+

(y31/EfficiencyNonElecHeatingDHW ('NEHWS1'))+
 (y32/EfficiencyNonElecHeatingDHW('NEHWS2')) +
 (y33/EfficiencyNonElecHeatingDHW('NEHWS3')); //MJ

fuel_dhw_nel_cost_eq.. FUEL_DHW_NEL_COST =e=
 y28*cost_oil/(EfficiencyNonElecDHW ('NEWS1')*1hvNonElecDHW
 ('NEWS1')) +
 y29*cost_oil/(EfficiencyNonElecDHW('NEWS2')*1hvNonElecDHW('NEWS
 2')) +

y30*cost_ng/(EfficiencyNonElecDHW('NEWS3')*1hvNonElecDHW('NEWS3
 '))+ y31*cost_oil/(EfficiencyNonElecHeatingDHW
 ('NEHWS1')*1hvNonElecHeatingDHW ('NEHWS1'))+

y32*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS2')*1hvNonElecH
 eatingDHW('NEHWS2')) +

y33*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS3')*1hvNonElecH
 eatingDHW('NEHWS3')); //euros (based on fuel consumption per
 m3)

```

QLel_eq..          QLel=e=QLDtot*SEAel; //MJ
QAel_eq..          QAel=e=QEaDTot*SEAel; //MJ

*----- PRIMARY ENERGY CONSUMPTION-----*
Q_pe_sys_el_eq..   Q_pe_sys_el =e= Qelgrid/ngrid; //MJ
Q_pe_sys_nel_eq..   Q_pe_sys_nel =e= QHnel+Qwnel; //MJ

obj2_energy..      Qt =e= Q_pe_sys_el + Q_pe_sys_nel;
//MJ
obj2..             Z('2')=e=Qt;

*-----EMISSIONS-----*
co2_heat_nel_cost_eq.. CO2_HEAT_NEL =e=
y10*oil_co2/(EfficiencyNonElecHeatSys('NEHS1')*lhvNonElecHeatS
ys('NEHS1')/oil_dens)+y11*ng_co2/(EfficiencyNonElecHeatSys('NE
HS2')*lhvNonElecHeatSys('NEHS2')/ng_dens)+

y12*ng_co2/(EfficiencyNonElecHeatSys('NEHS3')*lhvNonElecHeatSy
s('NEHS3')/ng_dens)+
y13*oil_co2/(EfficiencyNonElecHeatingDHW('NEHWS1')*lhvNonElecHe
atingDHW('NEHWS1')/oil_dens)
+
y14*ng_co2/(EfficiencyNonElecHeatingDHW('NEHWS2')*lhvNonElecHe
atingDHW('NEHWS2')/ng_dens) +
y15*ng_co2/(EfficiencyNonElecHeatingDHW('NEHWS3')*lhvNonElecHe
atingDHW('NEHWS3')/ng_dens); // kg CO2

co2_dhw_nel_cost_eq.. CO2_DHW_NEL =e=
y28*oil_co2/(EfficiencyNonElecDHW ('NEWS1')*lhvNonElecDHW
('NEWS1')/oil_dens) +
y29*oil_co2/(EfficiencyNonElecDHW('NEWS2')*lhvNonElecDHW('NEWS2
')/oil_dens) +

y30*ng_co2/(EfficiencyNonElecDHW('NEWS3')*lhvNonElecDHW('NEWS3'
)/ng_dens)+ y31*oil_co2/(EfficiencyNonElecHeatingDHW
('NEHWS1')*lhvNonElecHeatingDHW ('NEHWS1')/oil_dens)+

y32*ng_co2/(EfficiencyNonElecHeatingDHW('NEHWS2')*lhvNonElecHe
atingDHW('NEHWS2')/ng_dens) +
y33*ng_co2/(EfficiencyNonElecHeatingDHW('NEHWS3')*lhvNonElecHe
atingDHW('NEHWS3')/ng_dens); //kgCO2

co2_grid_eq.. CO2_GRID =e= Qelgrid * grid_co2; // kg CO2

tot_co2_eq.. TOT_CO2 =e= CO2_GRID + CO2_DHW_NEL +
CO2_HEAT_NEL;
obj3..           Z('3') =e= TOT_CO2;
*-----LINEARIZATION FOR SYSTEM SELECTION-----*

linear_eq1..      y1=l=mpos*XEHS('EHS1');
linear_eq2..      y1=l=QHDtot ;
linear_eq3..      y1=g=QHDtot-mpos*(1-XEHS('EHS1')) ;

```

```

linear_eq4..      y1=g=0 ;

linear_eq5..      y2=1=mpos*XEHS ('EHS2');
linear_eq6..      y2=1=QHDtot ;
linear_eq7..      y2=g=QHDtot-mpos*(1-XEHS ('EHS2')) ;
linear_eq8..      y2=g=0 ;

linear_eq9..      y3=1=mpos*XEHS ('EHS3');
linear_eq10..     y3=1=QHDtot ;
linear_eq11..     y3=g=QHDtot-mpos*(1-XEHS ('EHS3')) ;
linear_eq12..     y3=g=0 ;

linear_eq13..     y4=1=mpos*XEHCS ('EHCS1');
linear_eq14..     y4=1=QHDtot ;
linear_eq15..     y4=g=QHDtot-mpos*(1-XEHCS ('EHCS1'));
linear_eq16..     y4=g=0 ;

linear_eq17..     y5=1=mpos*XEHCS ('EHCS2');
linear_eq18..     y5=1=QHDtot ;
linear_eq19..     y5=g=QHDtot-mpos*(1-XEHCS ('EHCS2'));
linear_eq20..     y5=g=0 ;

linear_eq21..     y6=1=mpos*XEHCS ('EHCS3');
linear_eq22..     y6=1=QHDtot ;
linear_eq23..     y6=g=QHDtot-mpos*(1-XEHCS ('EHCS3'));
linear_eq24..     y6=g=0 ;

linear_eq25..     y7=1=mpos*XEHWS ('EHWS1') ;
linear_eq26..     y7=1=QHDtot ;
linear_eq27..     y7=g=QHDtot-mpos*(1-XEHWS ('EHWS1')) ;
linear_eq28..     y7=g=0 ;

linear_eq29..     y8=1=mpos*XEHWS ('EHWS2') ;
linear_eq30..     y8=1=QHDtot ;
linear_eq31..     y8=g=QHDtot-mpos*(1-XEHWS ('EHWS2')) ;
linear_eq32..     y8=g=0 ;

linear_eq33..     y9=1=mpos*XEHWS ('EHWS3') ;
linear_eq34..     y9=1=QHDtot ;
linear_eq35..     y9=g=QHDtot-mpos*(1-XEHWS ('EHWS3')) ;
linear_eq36..     y9=g=0 ;

linear_eq37..     y10=1=mpos*XNEHS ('NEHS1');
linear_eq38..     y10=1=QHDtot ;
linear_eq39..     y10=g=QHDtot-mpos*(1-XNEHS ('NEHS1')) ;
linear_eq40..     y10=g=0 ;

linear_eq41..     y11=1=mpos*XNEHS ('NEHS2');
linear_eq42..     y11=1=QHDtot ;
linear_eq43..     y11=g=QHDtot-mpos*(1-XNEHS ('NEHS2')) ;
linear_eq44..     y11=g=0 ;

linear_eq45..     y12=1=mpos*XNEHS ('NEHS3');
linear_eq46..     y12=1=QHDtot ;
linear_eq47..     y12=g=QHDtot-mpos*(1-XNEHS ('NEHS3')) ;
linear_eq48..     y12=g=0 ;

```

```

linear_eq49..      y13=1=mpos*XNEHWS('NEHWS1') ;
linear_eq50..      y13=1=QHDtot           ;
linear_eq51..      y13=g=QHDtot-mpos*(1-XNEHWS('NEHWS1'))
;
linear_eq52..      y13=g=0      ;

linear_eq53..      y14=1=mpos*XNEHWS('NEHWS2') ;
linear_eq54..      y14=1=QHDtot           ;
linear_eq55..      y14=g=QHDtot-mpos*(1-XNEHWS('NEHWS2'))
;
linear_eq56..      y14=g=0      ;

linear_eq57..      y15=1=mpos*XNEHWS('NEHWS3') ;
linear_eq58..      y15=1=QHDtot           ;
linear_eq59..      y15=g=QHDtot-mpos*(1-XNEHWS('NEHWS3'))
;
linear_eq60..      y15=g=0      ;

linear_eq61..      y16=1=mpos*XECS('ECS1');
linear_eq62..      y16=1=QCDtot      ;
linear_eq63..      y16=g=QCDtot-mpos*(1-XECS('ECS1'));
linear_eq64..      y16=g=0      ;

linear_eq65..      y17=1=mpos*XECS('ECS2');
linear_eq66..      y17=1=QCDtot      ;
linear_eq67..      y17=g=QCDtot-mpos*(1-XECS('ECS2'));
linear_eq68..      y17=g=0      ;

linear_eq69..      y18=1=mpos*XECS('ECS3');
linear_eq70..      y18=1=QCDtot      ;
linear_eq71..      y18=g=QCDtot-mpos*(1-XECS('ECS3'));
linear_eq72..      y18=g=0      ;

linear_eq73..      y19=1=mpos*XEHCS('EHCS1');
linear_eq74..      y19=1=QCDtot      ;
linear_eq75..      y19=g=QCDtot-mpos*(1-XEHCS('EHCS1')) ;
linear_eq76..      y19=g=0      ;

linear_eq77..      y20=1=mpos*XEHCS('EHCS2');
linear_eq78..      y20=1=QCDtot      ;
linear_eq79..      y20=g=QCDtot-mpos*(1-XEHCS('EHCS2')) ;
linear_eq80..      y20=g=0      ;

linear_eq81..      y21=1=mpos*XEHCS('EHCS3');
linear_eq82..      y21=1=QCDtot      ;
linear_eq83..      y21=g=QCDtot-mpos*(1-XEHCS('EHCS3')) ;
linear_eq84..      y21=g=0      ;

linear_eq85..      y22=1=mpos*XEWS('EWS1') ;
linear_eq86..      y22=1=QWDtot      ;
linear_eq87..      y22=g=QWDtot-mpos*(1-XEWS('EWS1')) ;
linear_eq88..      y22=g=0      ;

linear_eq89..      y23=1=mpos*XEWS('EWS2') ;
linear_eq90..      y23=1=QWDtot      ;

```

```

linear_eq91..      y23=g=QWDtot-mpos*(1-XEWS('EWS2')) ;
linear_eq92..      y23=g=0 ;

linear_eq93..      y24=l=mpos*XEWS('EWS3') ;
linear_eq94..      y24=l=QWDtot ;
linear_eq95..      y24=g=QWDtot-mpos*(1-XEWS('EWS3')) ;
linear_eq96..      y24=g=0 ;

linear_eq97..      y25=l=mpos*XEHWS('EHWS1') ;
linear_eq98..      y25=l=QWDtot ;
linear_eq99..      y25=g=QWDtot-mpos*(1-XEHWS('EHWS1')) ;
linear_eq100..     y25=g=0 ;

linear_eq101..     y26=l=mpos*XEHWS('EHWS2') ;
linear_eq102..     y26=l=QWDtot ;
linear_eq103..     y26=g=QWDtot-mpos*(1-XEHWS('EHWS2')) ;
linear_eq104..     y26=g=0 ;

linear_eq105..     y27=l=mpos*XEHWS('EHWS3') ;
linear_eq106..     y27=l=QWDtot ;
linear_eq107..     y27=g=QWDtot-mpos*(1-XEHWS('EHWS3')) ;
linear_eq108..     y27=g=0 ;

linear_eq109..     y28=l=mpos*XNEWS('NEWS1') ;
linear_eq110..     y28=l=QWDtot ;
linear_eq111..     y28=g=QWDtot-mpos*(1-XNEWS('NEWS1')) ;
linear_eq112..     y28=g=0 ;

linear_eq113..     y29=l=mpos*XNEWS('NEWS2') ;
linear_eq114..     y29=l=QWDtot ;
linear_eq115..     y29=g=QWDtot-mpos*(1-XNEWS('NEWS2')) ;
linear_eq116..     y29=g=0 ;

linear_eq117..     y30=l=mpos*XNEWS('NEWS3') ;
linear_eq118..     y30=l=QWDtot ;
linear_eq119..     y30=g=QWDtot-mpos*(1-XNEWS('NEWS3')) ;
linear_eq120..     y30=g=0 ;

linear_eq121..     y31=l=mpos*XNEHWS('NEHWS1') ;
linear_eq122..     y31=l=QWDtot ;
linear_eq123..     y31=g=QWDtot-mpos*(1-XNEHWS('NEHWS1')) ;
;
linear_eq124..     y31=g=0 ;

linear_eq125..     y32=l=mpos*XNEHWS('NEHWS2') ;
linear_eq126..     y32=l=QWDtot ;
linear_eq127..     y32=g=QWDtot-mpos*(1-XNEHWS('NEHWS2')) ;
;
linear_eq128..     y32=g=0 ;

linear_eq129..     y33=l=mpos*XNEHWS('NEHWS3') ;
linear_eq130..     y33=l=QWDtot ;
linear_eq131..     y33=g=QWDtot-mpos*(1-XNEHWS('NEHWS3')) ;
;
linear_eq132..     y33=g=0 ;

```



```

model example /all/;
*option mip=gurobi;
annfactor_systems=rate*((1+rate)**life_systems)/(((1+rate)**life_systems)-1);
annfactor_structure=rate*((1+rate)**life_structure)/(((1+rate)**life_structure)-1);
annfactor_win=rate*((1+rate)**life_win)/(((1+rate)**life_win)-1);
annfactor_lamps=rate*((1+rate)**life_lamps)/(((1+rate)**life_lamps)-1);
annfactor_ea=rate*((1+rate)**life_ea)/(((1+rate)**life_ea)-1);

```

```

*-----
-----

```

```

$STitle eps-constraint method

```

```

Set k1(k) the first element of k, km1(k) all but the first
elements of k;
k1(k)$ (ord(k)=1) = yes; km1(k)=yes; km1(k1) = no;
Set kk(k) active objective function in constraint allobj
set diff different objective function values per problem
/1*1000/

```

```

Parameter

```

```

    rhs(k)          right hand side of the constrained obj
functions in eps-constraint
    maxobj(k)       maximum value from the payoff table
    minobj(k)       minimum value from the payoff table
    intervals(k)    number of intervals that we divide the k-1
objective functions
    bestobj(k)      the best objective function value (maxobj for
dir=1 minobj for dir=-1)
    worstobj(k)     the worst objective function value (minobj for
dir=1 maxobj for dir=-1)
    step(k)         the step obtained from range divided by
intervals
    jump(k)         the jump for augmecon2
* added on 13/08/13
*****
*****
    loopworst(k)    the worst value from the loop
*****
*****
* added on 17/09/13
*****
*****
    progress(k)     the progress for loop k
*****
*****
zdiff(diff,k)     values of objective functions in different POS
zstar(k)          auxiliary parameter for the comparisons;

```

```

zdiff(diff,k)=0;
zstar(k)=0;

```

Scalar

```

iter    total number of iterations
infeas  total number of infeasibilities
elapsed_time elapsed time for payoff and e-sonstraint
start   start time
finish  finish time
summax  auxiliary parameter
firstOffMax, lastZero some counters
numdiff number of different POS
dcount  counter for different POS
issamenum auxiliary parameter counting how many obj. values
are same with already found

```

Variables

```

a_objval auxiliary variable for the objective function
obj       auxiliary variable during the construction of
the payoff table

```

Positive Variables

```

sl(k)    slack or surplus variables for the eps-
constraints

```

Equations

```

con_obj(k) constrained objective functions
augm_obj   augmented objective function to avoid weakly
efficient solutions
allobj     all the objective functions in one expression;

```

```

con_obj(km1).. z(km1) - dir(km1)*sl(km1) =e= rhs(km1);

```

```

* We optimize the first objective function and put the others
as constraints

```

```

* the second term is for avoiding weakly efficient points

```

```

*augm_obj..

```

```

* sum(k1,dir(k1)*z(k1))+1e-3*sum(km1,sl(km1)/(maxobj(km1)-
minobj(km1))) =e= a_objval;

```

```

augm_obj..

```

```

sum(k$(ord(k)=1),dir(k)*z(k)) +
sum(k$(ord(k)>1),power(10,-(ord(k)-1))*sl(k)/(maxobj(k)-
minobj(k))) =e= a_objval;

```

```

* sum(k$(ord(k)=1),dir(k)*z(k)) + 1.0e-
3*sum(k$(ord(k)>1),power(10,-(ord(k)-1))*sl(k)/(maxobj(k)-
minobj(k))) =e= a_objval;

```

```

allobj.. sum(kk, dir(kk)*z(kk)) =e= obj;

```

```

Model mod_payoff / example, allobj / ;

```

```

Model mod_epsmethod / example, con_obj, augm_obj / ;

```

Parameter

```

payoff(k,k) payoff tables entries;

```

```

Alias(k,kp);

```

```

option optcr=0.000;

* Generate payoff table applying lexicographic optimization
loop(kp,
  kk(kp)=yes;
  repeat
    solve mod_payoff using mip maximizing obj;
    payoff(kp, kk) = z.l(kk);
    z.fx(kk) = z.l(kk); // freeze the value of the last
objective optimized
    kk(k+1) = kk(k); // cycle through the objective
functions
  until kk(kp); kk(kp) = no;
* release the fixed values of the objective functions for the
new iteration
  z.up(k) = inf; z.lo(k) = -inf;
);
if (mod_payoff.modelstat<>1 and mod_payoff.modelstat<>8, abort
'no optimal solution for mod_payoff');

File fx / c:\gams\ziozas_dipl_log.out /;
File dx / c:\gams\ziozas_dipl.out /;

PUT dx ' PAYOFF TABLE' / ;
loop (kp,
  loop(k, put dx payoff(kp, k):12:2);
  put dx /;
);
put dx /;

*new 28-11-2013 for finding the actual range
$ontext
loop(kp,
  kk(kp)=yes;
  solve mod_payoff using mip minimizing obj;
  if (dir(kp)=1,
    maxobj(kp)=smax(k, payoff(k, kp));
    minobj(kp)=obj.l
  );
  if (dir(kp)=-1,
    maxobj(kp)=-obj.l;
    minobj(kp)=smin(k, payoff(k, kp))
  );
  kk(kp)=no;
);
$offtext

*display payoff;
minobj(k)=smin(kp, payoff(kp, k));
*minobj(k)=0;
maxobj(k)=smax(kp, payoff(kp, k));

*-----
-----
*new 17.03.2013

```

```

*-----
-----
*loop(k, intervals(k)=maxobj(k)-minobj(k));
loop(k, intervals(k)=10);

loop(k,
  if (dir(k)=1,
    bestobj(k)=maxobj(k);
    worstobj(k)=minobj(k);
  else
    bestobj(k)=minobj(k);
    worstobj(k)=maxobj(k)
  );
  step(k)=(maxobj(k)-minobj(k))/intervals(k)
);

rhs(k)=worstobj(k);
iter=0;
infeas=0;
start=jnow;

// added 2/3/14 for printing only the different values. New
items are:
numdiff=0; // number of different POS
issamenum=0; // counter for objective function values in the
comparison with the recently found
infeas=0;
zdiff(diff,k)=0;
zstar(k)=0;

*new 13/08/13
*****
*****
loopworst(k)=bestobj(k);
*****
*****
Put fx ' #'
loop(k$(ord(k)>1), put fx ' Progress', ord(k):1:0);
loop(k, put fx ' OBJ', ord(k):1:0);
put fx ' ';
loop(k$(ord(k)>1), put fx ' SLACK', ord(k):1:0);
put fx /;

dx.pw=20000;
put dx ' # TAC ENERGY EMISSIONS ';
put ' TOT_CAPCOST TOT_OPERCOST INVCOST
QHel QHnel QCel';
put ' Qwel Qwnel QLe1 QAel';
put ' CO2_HS CO2_WS CO2_GRID ';
loop(d, put ' DOOR', ord(d):1:0);
loop(wz, put ' WIND', ord(wz):1:0);
loop(w, put ' WALL', ord(w):1:0);
loop(r, put ' CEIL', ord(r):1:0);
loop(h, put ' FLOOR', ord(h):1:0);
loop(ehsi, put ' EHS', ord(ehsi):1:0);
loop(nehsi, put ' NEHS', ord(nehsi):1:0);

```

```

loop(ecsi, put ' ECS',ord(ecsi):1:0);
loop(ewsi, put ' EWS',ord(ewsi):1:0);
loop(newsi,put ' NEWS',ord(newsi):1:0);
loop(ehcsi,put ' EHCS',ord(ehcsi):1:0);
loop(ehws,put ' EHWS',ord(ehws):1:0);
loop(nehws,put ' NEHWS',ord(nehws):1:0);
loop(slci, put ' SLC',ord(slci):1:0);
loop(lj, put ' L',ord(lj):1:0);
loop(tv, put ' TV',ord(tv):1:0);
loop(cooker,put ' COOK',ord(cooker):1:0);
loop(washing,put ' WASH',ord(washing):1:0);
loop(refrig,put ' REFRIG',ord(refrig):1:0);
loop(pvs, put ' PHOTOV',ord(pvs):1:0);
put/;

repeat
  solve mod_epsmethod maximizing a_objval using mip;
  iter=iter+1;
  if (mod_epsmethod.modelstat<>1 and mod_payoff.modelstat<>8,
// not optimal is in this case infeasible
  infeas=infeas+1;
  put fx iter:5:0, ' infeasible'/;
  lastZero = 0;
  loop(k$(ord(k)>1),
    if(abs(rhs(k)-worstobj(k))>0.00001 and lastzero=0,
lastzero=ord(k)
  );
  loop(k$(ord(k)>1 and ord(k)<=lastzero),rhs(k)=bestobj(k));
  else
* update the worst value found so far - added 13/08/13
*****
*****
  loop(k$(ord(k)>2),
    if(dir(k)=1 and z.L(k)<loopworst(k),
loopworst(k)=z.L(k));
    if(dir(k)=-1 and z.L(k)>loopworst(k),
loopworst(k)=z.L(k));
  );
*****
*****
  put fx iter:7:0;
* added on 17.09.13 for reporting
*****
*****
loop(k$(ord(k)>1),
  if (dir(k)=1, progress(k)=100*(rhs(k)-
minobj(k))/(maxobj(k)-minobj(k))
  else progress(k)=100*(maxobj(k)-rhs(k))/(maxobj(k)-
minobj(k))
  );
  put fx progress(k):10:2,'%';
  );
*****
*****

```

```

loop(k, put fx z.l(k):12:2);
put fx ' *** '; // put /;
loop(km1, put fx sl.l(km1):12:2);

//*****
*****
// added 2/3/14 for printing only the different values. New
items are:
// set diff /1*1000/
// parameters zdiff(diff,k), zstar(k)
// scalars dcount, issamenum,numdiff

for (dcount=1 to numdiff,
loop(diff$(ord(diff)=dcount), loop(k,
zstar(k)=zdiff(diff,k)));
issamenum=0;
loop(k,
if(abs(z.L(k)-zstar(k))<0.001,
issamenum=issamenum+1);
);
if (issamenum=card(k), dcount=numdiff); //in order to
exit the loop
); // for loop
if (issamenum<card(k),
numdiff=numdiff+1;
loop(diff$(ord(diff)=numdiff), loop(k,
zdiff(diff,k)=z.L(k)));
put dx numdiff:5:0;
loop(k, put dx z.l(k):12:2);
put dx ' ';
put dx TOT_CAPCOST.1:9:2;
put dx ' ';
put dx TOT_OPERCOST.1:9:2;
put dx ' ';
put dx INVCOST.1:9:2;
put dx ' ';
put dx QHel.1:9:2;
put dx ' ';
put dx QHnel.1:9:2;
put dx ' ';
put dx QCel.1:9:2;
put dx ' ';
put dx Qwel.1:9:2;
put dx ' ';
put dx Qwnel.1:9:2;
put dx ' ';
put dx QLe1.1:9:2;
put dx ' ';
put dx QAel.1:9:2;
put dx ' ';
put dx CO2_HEAT_NEL.1:9:2;
put dx ' ';
put dx CO2_DHW_NEL.1:9:2;
put dx ' ';
put dx CO2_GRID.1:9:2;

```

```

        put dx '  ';
        loop(d, put dx XDOOR.l(d):9:0);
        loop(wz, put dx XWIN.l(wz):9:0);
        loop(w, put dx XWALL.l(w):9:0);
        loop(r, put dx XCEIL.l(r):9:0);
        loop(h, put dx XFLOOR.l(h):9:0);
        loop(ehssi, put dx XEHS.l(ehssi):9:0);
        loop(nehssi, put dx XNEHS.l(nehssi):9:0);
        loop(ecsi, put dx XECS.l(ecsi):9:0);
        loop(ewsi, put dx XEWS.l(ewsi):9:0);
        loop(newsisi, put dx XNEWS.l(newsisi):9:0);
        loop(ehcsi, put dx XEHCS.l(ehcsi):9:0);
        loop(ehwsisi, put dx XEHWS.l(ehwsisi):9:0);
        loop(nehwsisi, put dx XNEHWS.l(nehwsisi):9:0);
        loop(slci, put dx XSLC.l(slci):9:0);
        loop(lj, put dx XL.l(lj):9:0);
        loop(tv, put dx XTV.l(tv):9:0);
        loop(cooker, put dx XCOOKER.l(cooker):9:0);
        loop(washing, put dx XWASHING.l(washing):9:0);
        loop(refrig, put dx XREFRIG.l(refrig):9:0);
        loop(pvs, put dx XPHOTOV.l(pvs):9:0);

*
        );

*
        put dx (sum(p, cost(p)*X.L(p))):10:0,
(sum(p, X.L(p))):5:0;
*
        loop(p, put dx X.L(p):4:0);
        put /;
    );

*****
*****

*
    the jump is for AUGMECON2
    jump(k)=1;
*
    The jump is calculated for the innermost objective
function (ord(k)=2)
    jump(k)$(ord(k)=2) = 1+abs(floor(sl.L(k)/step(k)));
    loop(k$(jump(k)>1), put fx '    jump');
    put fx /;
    );
*
    Proceed forward in the grid
    firstOffMax = 0;
    loop(k$(ord(k)>1),
        if(abs(rhs(k)-bestobj(k))>0.000001 and firstOffMax=0,
            if (dir(k)=1,
rhs(k)=min((rhs(k)+jump(k)*step(k)), bestobj(k))
                else rhs(k)=max((rhs(k)-jump(k)*step(k)), bestobj(k))
            );
            firstOffMax=ord(k)
        );
    );

    loop(k$(ord(k)>1),
        if(ord(k)< firstOffMax, rhs(k)=worstobj(k))
    );

```

```

*new 13/08/13 go to the worst value from the previous loop of
k-1
*****
*****
$ontext
  loop(k$(ord(k)>1),
    if (ord(k)=firstOffMax and firstOffMax > 2,
*Florios add 27.11.2013
      if (dir(k)=1, rhs(k)=loopworst(k)+step(k);
loopworst(k)=bestobj(k);
      else rhs(k)=loopworst(k)-step(k);
loopworst(k)=bestobj(k);)
*end Florios add 27.11.2013
**      rhs(k)=loopworst(k)+1;
**      loopworst(k)=bestobj(k);
    );
$offtext
*****
*****
  summax=0;
  loop(k$(ord(k)>1),
    if(abs(rhs(k)-bestobj(k))<0.000001, summax=summax+1);
  );
* for check premature end
* if (iter = 120,
*   summax=card(k)-1;
*   firstOffMax=0;
* );
until summax=card(k)-1 and firstOffMax=0;

finish=jnow;
elapsed_time=(finish-start)*86400;

put /;
put 'Infeasibilities = ', infeas:5:0 /;
put 'Elapsed time: ',elapsed_time:7:2, ' seconds' / ;

putclose fx; // close the point file
putclose dx; // close the point file

```


2. Κώδικας GAMS με 2 αντικειμενικές συναρτήσεις

Παρακάτω παρατίθενται ο κώδικας που χρησιμοποιήθηκε για την επίλυση του προβλήματος με τις 2 αντικειμενικές συναρτήσεις σε γλώσσα GAMS :

```
$eolcom //

* to make the model linear sets of walls, floors and ceilings
have 12 components
* componets 1:3 are without insulation, 4:6 with insulation 1,
7:9 insulation 2, 10:12 insulation 3

Sets
  d          door type
/door1*door3/
  wz        window type
/window1*window3/
  w         wall structure
/wall1*wall12/
  r         structure of ceiling
/ceiling1*ceiling12/
  h         floor structure
/floor1*floor12/
  tv        television           /tv1*tv3/
  cooker    cooker
/cooker1*cooker3/
  washing   washing machine
/wash1*wash3/
  refrig    refrigerator
/refrig1*refrig3/
  pvs       photovoltiacs       /pvs1*pvs3/
  ehsi      electrical heating system /EHS1*EHS3/
  nehsi     non electrical heating system
/NEHS1*NEHS3/
  ecsi      electrical cooling system /ECS1*ECS3/
  ewsi      electrical DHW         /EWS1*EWS3/
  newsi     non electric DHW
/NEWS1*NEWS3/
  ehcsi     electrical heating-cooling
/EHCS1*EHCS3/
  ehwi      electrical heating DHW
/EHWS1*EHWS3/
  nehwi     non electrical heating DHW
/NEHWS1*NEHWS3/
  slci      solar collector category
/solar1*solar3/
  lj        type of lamps
/lamp1*lamp3/
  month     month                 /1*12/
  hour      period of day         /1*24/

k objective functions /1*2/
```

;

Parameter dir(k) direction of the objective functions 1 for
max and -1 for min

```
/ 1 -1
   2 -1
/
```

Parameters

To(month) Average external temperature for each month
(Celsius)

```
/
1      7.4
2      7.8
3     10.8
4     15.8
5     21.5
6     26.4
7     28.6
8      28
9     24.2
10     18.9
11     13.1
12     8.7
/
```

td(month) month duration in days

```
/
1      31
2      28
3      31
4      30
5      31
6      30
7      31
8      31
9      30
10     31
11     30
12     31
/
```

Isl(month) solar radiation kWh * m²*day⁻¹

```
/
1      1.39
2      1.91
3      2.78
4      3.85
5      5.01
6      5.27
7      4.93
8      4.62
9      3.93
10     2.49
11     1.54
12     1.22
```

/

tm(month) month duration in hours

/

1	744
2	672
3	744
4	720
5	744
6	720
7	744
8	744
9	720
10	744
11	720
12	744

/

Tdcw(month) temperature of cold water inlet to the DHW system
(Celsius)

/

1	11.3
2	10.9
3	11.8
4	14.3
5	17.7
6	21.6
7	24.7
8	25.4
9	24.2
10	21.1
11	16.9
12	13.5

/

WSm(month) indicator for DHW demand each month

/

1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1

/

CS(month) indicator for cooling demand each month

/

1	1
2	1
3	1

4 1
5 1
6 1
7 1
8 1
9 1
10 1
11 1
12 1
/

HS(month) indicator for heating demand each month

/
1 1
2 1
3 1
4 1
5 1
6 1
7 1
8 1
9 1
10 1
11 1
12 1
/

Wo(month) specific humidity outdoors each month (kgwater per kgair)

/
1 0.00400
2 0.00420
3 0.00450
4 0.00520
5 0.00630
6 0.00730
7 0.00820
8 0.00850
9 0.00770
10 0.00690
11 0.00590
12 0.00495
/

Udoors(d) heat transfer coefficient for doors $W*m^{-2}*K$

/
door1 2.04
door2 0.86
door3 1.64
/

Costdoor(d) Cost for doors per square meter

/
door1 226
door2 510

door3 244
/

UWindows(wz) Heat transfer coefficient for windows
/
window1 1.28
window2 0.79
window3 1.15
/

CostWindows(wz) cost for windows per square meter
/
window1 209
window2 258
window3 161
/

Gwin(wz) Solar heat gain coefficient
/
window1 0.44
window2 0.64
window3 0.44
/

Uwalls(w) heat transfer coefficient of walls
/
wall1 1.4731
wall2 1.4203
wall3 1.5295
wall4 0.3388
wall5 0.3359
wall6 0.3417
wall7 0.5587
wall8 0.5509
wall9 0.5666
wall10 0.4926
wall11 0.4865
wall12 0.4987
/

CostWalls(w) cost of walls euros per m2
/
wall1 90.25
wall2 140.50
wall3 100.00
wall4 99.09
wall5 149.34
wall6 108.84
wall7 97.21
wall8 147.46
wall9 106.96
wall10 95.20
wall11 145.45
wall12 104.95
/

UFloor(h) heat transfer coefficient of floors

/
floor1 2.6940
floor2 2.6676
floor3 2.7005
floor4 0.3782
floor5 0.3777
floor6 0.3784
floor7 0.6746
floor8 0.6730
floor9 0.6750
floor10 0.5805
floor11 0.5793
floor12 0.5808
/

CostFloors(h) cost of floors euros per m2

/
floor1 26.25
floor2 28.25
floor3 27.25
floor4 35.09
floor5 37.09
floor6 36.09
floor7 33.21
floor8 35.21
floor9 34.21
floor10 31.20
floor11 33.20
floor12 32.20
/

UCeiling(r) heat transfer coefficient of ceilings

/
ceiling1 2.6901
ceiling2 2.7079
ceiling3 2.4296
ceiling4 0.3781
ceiling5 0.3785
ceiling6 0.3725
ceiling7 0.6744
ceiling8 0.6755
ceiling9 0.6567
ceiling10 0.5804
ceiling11 0.5812
ceiling12 0.5672
/

CostCeiling(r) cost of ceilings euros per m2

/
ceiling1 20.95
ceiling2 20.50
ceiling3 44.25
ceiling4 29.79
ceiling5 29.34

```
ceiling6 53.09
ceiling7 27.91
ceiling8 27.46
ceiling9 51.21
ceiling10 25.90
ceiling11 25.45
ceiling12 49.20
/
```

```
LampsPower(lj) power of lamps
/
lamp1 20
lamp2 24
lamp3 18
/
```

```
CostLamps(lj) cost of lamps
/
lamp1 13
lamp2 7
lamp3 77
/
```

```
EfficiencyElecHeatSys(ehsi) efficiency of electrical heating
systems
/
EHS1 1
EHS2 0.98
EHS3 0.9
/
```

```
CostElecHeatSys(ehsi) cost of electrical heating
systems
/
EHS1 1290
EHS2 1226
EHS3 1161
/
```

```
EfficiencyNonElecHeatSys(nehsi) efficiency of non-
electrical heating systems
/
NEHS1 0.83
NEHS2 0.85
NEHS3 0.91
/
```

```
* For non elec systems the lhv of respective fuel is used
* for oil is 34560 MJ per m3 and for gas =32.4 MJ per m3
lhvnonelecheatsys(nehsi) low heating value of non elec heat
sys
/
NEHS1 34560
NEHS2 32.4
NEHS3 32.4
/
```

CostNonElecHeatSys(nehsi) cost of non- electrical heating systems

/

NEHS1 839

NEHS2 680

NEHS3 1019

/

EfficiencyElecCoolingSys(ecsi) efficiency of electrical cooling systems

/

ECS1 2.5

ECS2 3

ECS3 3.5

/

CostElecCoolingSys(ecsi) cost of electrical cooling systems

/

ECS1 387

ECS2 516

ECS3 645

/

EfficiencyElecHeatCoolingSys(ehcsi) efficiency of electrical heating cooling systems

/

EHCS1 3

EHCS2 3.5

EHCS3 4

/

CostElecHeatCoolingSys(ehcsi) cost of electrical heating cooling systems

/

EHCS1 645

EHCS2 774

EHCS3 903

/

EfficiencySolar(slci) efficiency of solar collectors

/

solar1 0.827

solar2 0.642

solar3 0.791

/

AreaSolar(slci) area of solar collectors

/

solar1 2.3

solar2 2

solar3 2.35

/

CostSolar(slci) cost of solar collectors
/
solar1 7224
solar2 5160
solar3 6966
/

EfficiencyElecDHW(ewsi) efficiency of electrical DHW
/
EWS1 1
EWS2 0.98
EWS3 0.95
/

CostElecDHW(ewsi) cost of electrical DHW
/
EWS1 1169
EWS2 1096
EWS3 1032
/

EfficiencyNonElecDHW(newsi) efficiency of non-electrical DHW
/
NEWS1 0.83
NEWS2 0.6
NEWS3 0.8
/

CostNonElecDHW(newsi) cost of non-electrical DHW
/
NEWS1 645
NEWS2 451
NEWS3 580
/

lhvNonElecDHW(newsi) fuel of non-electrical DHW
/
NEWS1 34560
NEWS2 34560
NEWS3 32.4
/

EfficiencyElecHeatingDHW(ehws) efficiency of electrical heating DHW
/
EHWS1 1
EHWS2 0.98
EHWS3 4
/

CostElecHeatingDHW(ehws) cost of electrical heating DHW
/
EHWS1 1548
EHWS2 1277
EHWS3 3057
/

EfficiencyNonElecHeatingDHW(nehwsi) efficiency of non-electrical heating DHW

/
NEHWS1 0.88
NEHWS2 0.89
NEHWS3 0.95
/

lhvNonElecHeatingDHW(nehwsi) fuel of non-electrical heating DHW

/
NEHWS1 34560
NEHWS2 32.4
NEHWS3 32.4
/

CostNonElecHeatingDHW(nehwsi) cost of non-electrical heating DHW

/
NEHWS1 1032
NEHWS2 1096
NEHWS3 1225
/

PowerTV(tv) energy consumption of televisions W

/
tv1 120
tv2 52
tv3 58
/

CostTV(tv) costs of tv

/
tv1 437
tv2 551
tv3 516
/

PowerCooker(cooker) consumption of cookers W

/
cooker1 9530
cooker2 10475
cooker3 10000
/

CostCooker(cooker) cost of cookers

/
cooker1 645
cooker2 516
cooker3 580
/

PowerWashing(washing) consumption per cycle for washing machines KWh

```
/
wash1    1.05
wash2    0.73
wash3    0.79
/
```

EfficiencyWashing(washing) efficiency of washing machines
motors efficiency (%)

```
/
wash1    0.7
wash2    0.7
wash3    0.7
/
```

CostWashing(washing) cost of washing machines

```
/
wash1    387
wash2    516
wash3    495
/
```

PowerRefrig(refrig) daily consumption of refrigerators KWh

```
/
refrig1  0.51
refrig2  0.42
refrig3  0.50
/
```

EfficiencyRefrig(refrig) efficiency of refrigerators motors

```
/
refrig1  0.7
refrig2  0.7
refrig3  0.7
/
```

CostRefrig(refrig) cost of refrigerators

```
/
refrig1  445
refrig2  516
refrig3  477
/
```

GenerationEfficiencyPhotoV(pvs) generation efficiency of
photovoltaics systems (%)

```
/
pvs1    0.1565
pvs2    0.1616
pvs3    0.1480
/
```

AreaPhotoV(pvs) area of photovoltaics systems m²

```
/
pvs1    6.4
pvs2    6.4
```

pvs3 8.13
/

CostPhotoV(pvs) cost of photovoltaics

/
pvs1 1548
pvs2 1585
pvs3 1817
/

annfactor_systems annuity of energy systems
annfactor_structure annuity of structure components
annfactor_win annuity of windows
annfactor_lamps annuity of lamps
annfactor_ea annuity of electrical appliances
;

Scalars

ACH	air change per hour (h ⁻¹)
/1.5/	
Awin	area of windows (m ²)
/13/	
Fconv	Conversion Factor MJ*kWh ⁻¹
/3.6/	
Adoor	area of doors m ²
/3/	
Awallint	area of internal walls m ²
/99/	
Awallext	area of external walls
/194/	
Aceil	area of ceilings m ²
/62/	
Afloor	area of floors m ²
/127/	
ShadingFactorWindows	Windows Shading Factors
/1/	
UsageLamps	dialy usage of lamps hours
/13/	
Nlamps	number of lamps
/8/	
UsageTV	daily usage of tv hours
/4/	
UsageCooker	daily usage of cooker
/0.5/	
ShadingFactorSolar	shading factor of solar
collectors	/1/
CPair	specific heat of air at
constant pressure kj*(kg*k) ⁻¹	/1.0035/
CPwater	specific heat of water at
constant pressure kj*(kg*k) ⁻¹	/4.18/
FCm	window correction factor for
movable devices (%)	/1/

```

Ff                frame factor for windows (%)
/1/
dair              air density (kg*m^-3)
/1.2/
dwater           water density (kg*m^-3)
/1000/
ho              outdoors combined convection-
radiation coefficient W*(m^2*K)^-1 /28/
Mw              daily need of hot water (m3
per second)    /0.00000278/
Npeople         number of people
/4/
Hfg             latent heat of vaporization
(kj per kg)   /2340/
wi             specific humidity indoors
(kgwater per kgair) /0.00834/
ngrid           efficiency of the electricity
supply grid    /0.35/
SEAel          assuming no losses from
supply to consumption /1/
Tih            internal design temperature
for heating season (Celsius) /18/
Tic            internal design temperature
for cooling season (Celsius) /26/
Tdhw           tempreature for dhw
/60/
V              internal volume of the
building (m^3) /344/
PRphotov       PR Value for photovoltaics
/0.75/
Qhuman         heat emmitted from people (W)
/115/
CyclesWashing  usage cycles per day for
washing machines /0.5/
mneg          negative number for if
statement     /-1000000/
mpos          positive number for if
statement     /1000000/

life_systems life for energy systems (years) /20/
life_structure life for walls ceils and floors /50/
life_win      life for windows /30/
life_lamps   life for lamps /4/
life_ea      life for electrical appliances /10/
rate         rate for annualized cost /0.06/

cost_oil     cost of oil in euros per m3 /740/
cost_ng     cost of natural gas in euros per m3 /0.3/
cost_elec   electricity cost when buying from grid per kWh
/0.01596/
*lhv_oil   low heating value of oil kWh per m3 (for boilers)
/9600/
*lhv_gas   low heating value of gas kwh per m3 (assume 0.55 of
air density) /9/

```

;

Binary Variables

XDOOR(d)
XWIN(wz)
XWALL(w)
XCEIL(r)
XFLOOR(h)
XEHS(ehsi)
XNEHS(nehsi)
XECS(ecsi)
XEWS(ewsi)
XNEWS(newsi)
XEHCS(ehcsi)
XEHWS(ehwsi)
XNEHWS(nehwsi)
XSLC(slci)
XL(lj)
XTV(tv)
XCOOKER(cooker)
XWASHING(washing)
XREFRIG(refrig)
XPHOTOV(pvs)
;

positive Variables

Qc annual primary energy consumption for cooling (Mj*year⁻¹)
Qh annual primary energy consumption for heating (Mj*year⁻¹)
Qcel annual primary energy consumption for cooling consumed by an electrical system (Mj*year⁻¹)
Qdhw annual primary energy consumption for DHW (Mj*year⁻¹)
Qwel annual primary energy consumption for DHW by a electrical system (Mj*year⁻¹)
Qwnel annual primary energy consumption for DHW by a non-electrical system (Mj*year⁻¹)

Qhel annual primary energy consumption for heating from a electrical system (Mj*year⁻¹)

Qlel energy consumption for lighting Mj per year
QDel annual demand for electricity Mj per year

QCDtot annual cooling demand (Mj*year⁻¹)
QHDtot annual heating demand (Mj*year⁻¹)
QLDtot annual energy demand for lighting (Mj*year⁻¹)
QWDtot annual energy demand for DHW (Mj*year⁻¹)
QEaDTot total annual energy demand for electrical appliances (Mj*year⁻¹)

QAel annual energy consumption for electrical appliances (Mj*year⁻¹)

QCD(month) cooling demand each month MJ
 QHD(month) heating demand each month MJ
 QSL(month) solar heat gain each month (kWh)
 QeaD(month) energy for electrical appliances each month MJ
 DQdhw(month) net DHW demand MJ
 QLD(month) energy demand for lighting each month MJ
 QdSLC(month) DHW demand provided by a solar collector kWh

QPhoto(month) energy produced by a photovoltaic cell kWh

Qelgrid energy provided by the grid MJ per year

BLC Building Load Factor (W*K⁻¹)

QINHG(month) internal heat gain each month kWh
 QTV(month) energy demand for TV each month kWh
 QCooker(month) energy demand for cooker each month kWh
 Qwash(month) energy demand for washing machine each month kWh
 Qrefrig(month) energy demand for refrigerator each month kWh
 QHel energy consumption of electrical systems kWh
 QHnel energy consumption of non-electrical systems kWh
 pvsupply photovoltaic supply kWh

Q_TV_HEAT(month) heat from tv kWh
 Q_COOKER_HEAT(month) heat from cooker kWh
 Q_WASH_HEAT(month) heat from washing machine kWh
 Q_REFRIG_HEAT(month) heat from refrigerator kWh
 QEA_HEAT(month) heat from electric appliances kWh

COSTfordoor Investment cost for doors
 COSTforwindows Investment cost for windows
 COSTforwalls Investment cost for walls
 COSTforceilings Investment cost for ceilings
 COSTforfloors Investment cost for floors
 COSTforheatingsystem Investment cost for heating systems
 COSTforcoolingsystem Investment cost for cooling systems
 COSTforwatersystem Investment cost for dhw systems
 COSTforheatingcooling Investment cost for heating-cooling systems
 COSTforheatingwatersystem Investment cost for for heating dhw systems
 COSTforsolar Investment cost for solar collectors

COSTforlight	Investment cost for lighting
COSTforPhotov photovoltaics	Investment cost for
COSTforTV	Investment cost for TVs
COSTforcooker	Investment cost for cookers
COSTforWashing machines	Investment cost for washing
COSTforRefrig refrigerators	Investment cost for
TOT_OPERCOST	total operational cost
INVCOST	investmenet cost
TOT_CAPCOST	total capital cost
FUEL_DHW_NEL_COST	cost for fuel water systems
FUEL_HEAT_NEL_COST	cost for fuel heat systems
OPERCOST_FUEL	operational cost for fuel
OPERCOST_EL	operational cost for electricity
Qalt	total alternative energy
generation in MJ	
Q_pe_sys_el	total primary energy consumption
of electrical systems	
Q_pe_sys_nel	total primary energy consumption
of non electrical systems	
y1	variable for linearization
y2	variable for linearization
y3	variable for linearization
y4	variable for linearization
y5	variable for linearization
y6	variable for linearization
y7	variable for linearization
y8	variable for linearization
y9	variable for linearization
y10	variable for linearization
y11	variable for linearization
y12	variable for linearization
y13	variable for linearization
y14	variable for linearization
y15	variable for linearization
y16	variable for linearization
y17	variable for linearization
y18	variable for linearization
y19	variable for linearization
y20	variable for linearization
y21	variable for linearization
y22	variable for linearization
y23	variable for linearization
y24	variable for linearization
y25	variable for linearization
y26	variable for linearization
y27	variable for linearization
y28	variable for linearization


```

y29          variable for linearization
y30          variable for linearization
y31          variable for linearization
y32          variable for linearization
y33          variable for linearization
;

```

Free Variables

```

Qdhwu(month)    DHW demand kWh
QTClosses(month) heat transimittance losses each month
heating season kWh
QTHlosses(month) heat transimittance losses each month
cooling season kWh
QCVEN(month)    ventilation losses each month cooling
season (kWh)
QHVEN(month)    ventilation losses each month heating
season (kWh)

```

```

Z(K)           objective functions
TAC            total annual cost
Qt            total primary energy
consumption
;

```

Equations

```

*-----Orismos eksisoseon-----
-----

```

```

doors          Decision equation
for door
windows        Decision equation
for window
walls          Decision equation
for walls
ceilings       Decision equation
for ceilings
floors         Decision equation
for floors
Photov         Decision equation
for photovoltaics
TVs           Decision equation
for TVs
Cooker_eq      Decision equation
for cookers
Washing_eq     Decision equation
for washing machines
Refrig_eq      Decision equation
for refrigerators
lamps         Decision equation
for lamps

spaceheating   Decision equation
for spaceheating
spacecooling   Decision equation
for spacecooling

```

dhwsystems Decision equation
 for DHW systems
 solarcollector Decision equation
 for Solar Collectors

COSTfordoor_eq Equation of
 investment cost for doors
 COSTforwindows_eq Equation of
 Investment cost for windows
 COSTforwalls_eq Equation of
 Investment cost for walls
 COSTforceilings_eq Equation of
 Investment cost for ceilings
 COSTforfloors_eq Equation of
 Investment cost for floors
 COSTforheatingsystem_eq Equation of
 Investment cost for heating systems
 COSTforcoolingsystem_eq Equation of
 Investment cost for cooling systems
 COSTforwatersystem_eq Equation of
 Investment cost for dhw systems
 COSTforheatingcooling_eq Equation of
 Investment cost for heating-cooling systems
 COSTforheatingwatersystem_eq Equation of
 Investment cost for for heating dhw systems
 COSTforsolar_eq Equation of
 Investment cost for solar collectors
 COSTforlight_eq Equation of
 Investment cost for lighting
 COSTforPhotov_eq Equation of
 Investment cost for photovoltaics
 COSTforTV_eq Equation of
 Investment cost for TVs
 COSTforcooker_eq Equation of
 Investment cost for cookers
 COSTforWashing_eq Equation of
 Investment cost for washing machines
 COSTforRefrig_eq Equation of
 Investment cost for refrigerators

BLC_eq Equation of BLC
 value of a building

*-----TECHNICAL PARAMETERS -----

QTClosses_eq(month) Equation of monthly Transimission
 cooling season
 QCVEN_eq(month) Equation of ventilation losses
 cooling season
 QTHlosses_eq(month) Equation of monthly Transimission
 heating season
 QHVEN_eq(month) Equation of ventilation losses
 heating season

QCDtot_eq kWh	Equation of	annual cooling demand
QHDtot_eq kWh	Equation of	annual heating demand
QINHG_eq(month)	Equation of	internal heat gain
QSL_eq(month)	Equation of	solar gains each month
QTV_eq(month) each month kWh	Equation of	energy demand for TV
QCooker_eq(month) each month kWh	Equation of	energy demand for cooker
Qwash_eq(month) washing machine each month kWh	Equation of	energy demand for
Qrefrig_eq(month) refrigerator each month kWh	Equation of	energy demand for
Q_tv_heat_eq(month)		
Q_cooker_heat_eq(month)		
Q_wash_heat_eq(month)		
Q_refrig_eq(month)		
Qea_heat_eq(month)		
QLD_eq(month) each month kWh	Equation of	energy demand for lighting
QLDtot_eq lighting Mj per year	Equation of	annual energy demand for
QLel_eq lighting Mj per year	Equation of	energy consumption for
QWDtot_eq DHW (Mj*year ⁻¹)	Equation of	annual energy demand for
QPhoto_eq(month) photovoltaic cell per month	Equation of	energy generation by a
Qalt_eq a photovoltaic cell	Equation of	total energy generation of
QEaDTot_eq electrical appliances	Equation of	annual energy demand for
QAel_eq for electrical appliances	Equation of	annual energy consumption
altenergy_eq alternative energy generation systems	Equation of	power supply from
QdSLC_eq(month) solar collector kWh	Equation of	DHW demand provided by a
Qdhwu_eq(month)	Equation of	DHW demand
Qea_eq(month) appliaces each month kWh	Equation of	energy for electrical
QDel_eq electricity	Equation of	annual demand for
ifequation1(month)	if equation	1

```

ifequation2(month)      if equation 2
ifequation3(month)      if equation 3
ifequation4              if equation 4
*-----linearization-----
-----
-----

```

```

linear_eq1              linearization equation
linear_eq2              linearization equation
linear_eq3              linearization equation
linear_eq4              linearization equation
linear_eq5              linearization equation
linear_eq6              linearization equation
linear_eq7              linearization equation
linear_eq8              linearization equation
linear_eq9              linearization equation
linear_eq10             linearization equation
linear_eq11             linearization equation
linear_eq12             linearization equation
linear_eq13             linearization equation
linear_eq14             linearization equation
linear_eq15             linearization equation
linear_eq16             linearization equation
linear_eq17             linearization equation
linear_eq18             linearization equation
linear_eq19             linearization equation
linear_eq20             linearization equation
linear_eq20             linearization equation
linear_eq21             linearization equation
linear_eq22             linearization equation
linear_eq23             linearization equation
linear_eq24             linearization equation
linear_eq25             linearization equation
linear_eq26             linearization equation
linear_eq27             linearization equation
linear_eq28             linearization equation
linear_eq29             linearization equation
linear_eq30             linearization equation
linear_eq31             linearization equation
linear_eq32             linearization equation
linear_eq33             linearization equation
linear_eq34             linearization equation
linear_eq35             linearization equation
linear_eq36             linearization equation

```

```

Qhel_eq                Equation of annual primary energy
consumption for heating from a electrical system (Mj*year^-1)

```

```

linear_eq37            linearization equation
linear_eq38            linearization equation
linear_eq39            linearization equation
linear_eq40            linearization equation
linear_eq41            linearization equation
linear_eq42            linearization equation
linear_eq43            linearization equation
linear_eq44            linearization equation

```

linear_eq45	linearization equation
linear_eq46	linearization equation
linear_eq47	linearization equation
linear_eq48	linearization equation
linear_eq49	linearization equation
linear_eq50	linearization equation
linear_eq51	linearization equation
linear_eq52	linearization equation
linear_eq53	linearization equation
linear_eq54	linearization equation
linear_eq55	linearization equation
linear_eq56	linearization equation
linear_eq57	linearization equation
linear_eq58	linearization equation
linear_eq59	linearization equation
linear_eq60	linearization equation

QHnel_eq Equation of energy consumption of
non-electrical systems kWh

linear_eq61	linearization equation
linear_eq62	linearization equation
linear_eq63	linearization equation
linear_eq64	linearization equation
linear_eq65	linearization equation
linear_eq66	linearization equation
linear_eq67	linearization equation
linear_eq68	linearization equation
linear_eq69	linearization equation
linear_eq70	linearization equation
linear_eq71	linearization equation
linear_eq72	linearization equation
linear_eq73	linearization equation
linear_eq74	linearization equation
linear_eq75	linearization equation
linear_eq76	linearization equation
linear_eq77	linearization equation
linear_eq78	linearization equation
linear_eq79	linearization equation
linear_eq80	linearization equation
linear_eq81	linearization equation
linear_eq82	linearization equation
linear_eq83	linearization equation
linear_eq84	linearization equation

Qcel_eq Equation of annual primary energy
consumption for cooling consumed by an electrical system
(Mj*year⁻¹)

linear_eq85	linearization equation
linear_eq86	linearization equation
linear_eq87	linearization equation
linear_eq88	linearization equation
linear_eq89	linearization equation
linear_eq90	linearization equation
linear_eq91	linearization equation

linear_eq92	linearization equation
linear_eq93	linearization equation
linear_eq94	linearization equation
linear_eq95	linearization equation
linear_eq96	linearization equation
linear_eq97	linearization equation
linear_eq98	linearization equation
linear_eq99	linearization equation
linear_eq100	linearization equation
linear_eq101	linearization equation
linear_eq102	linearization equation
linear_eq103	linearization equation
linear_eq104	linearization equation
linear_eq105	linearization equation
linear_eq106	linearization equation
linear_eq107	linearization equation
linear_eq108	linearization equation

Qwel_eq Equation of annual primary energy
consumption for DHW by a electrical system (Mj*year^-1)

linear_eq109	linearization equation
linear_eq110	linearization equation
linear_eq111	linearization equation
linear_eq112	linearization equation
linear_eq113	linearization equation
linear_eq114	linearization equation
linear_eq115	linearization equation
linear_eq116	linearization equation
linear_eq117	linearization equation
linear_eq118	linearization equation
linear_eq119	linearization equation
linear_eq120	linearization equation
linear_eq121	linearization equation
linear_eq122	linearization equation
linear_eq123	linearization equation
linear_eq124	linearization equation
linear_eq125	linearization equation
linear_eq126	linearization equation
linear_eq127	linearization equation
linear_eq128	linearization equation
linear_eq129	linearization equation
linear_eq130	linearization equation
linear_eq131	linearization equation
linear_eq132	linearization equation

Qwnel_eq Equation of annual primary energy
consumption for DHW by a non-electrical system (Mj*year^-1)

*-----if statements-----

invcost_eq	objective function for cost
tot_opercost_eq	annual operational cost
opercost_fuel_eq	operational cost of fuel based systems

```

opercost_el_eq      operational cost electricity
tot_ann_cost_eq    total annual cost
capcost_eq
fuel_heat_nel_cost_eq cost for fuel based heating systems
fuel_dhw_nel_cost_eq cost for fuel based water systems

Q_pe_sys_el_eq      Equation of annual primary
consumption from systems consuming electricity
Q_pe_sys_nel_eq     Equation of annual primary
consumption from systems consuming fuels
obj2_energy         objective function for energy
consumption
obj1                definition of Z1
obj2                definition of Z2

;

doors..            sum(d,XDOOR(d))=e=1 ;
windows..          sum(wz,XWIN(wz))=e=1;
walls..            sum(w,XWALL(w))=e=1 ;
ceilings..         sum(r,XCEIL(r))=e=1;
floors..           sum(h, XFLOOR(h))=e=1 ;
TVs..             sum(tv,XTV(tv))=e=1;
Cooker_eq..        sum(cooker,XCOOKER(cooker))=e=1;
Washing_eq..       sum(washing,XWASHING(washing))=e=1;
Refrig_eq..        sum(refrig,XREFRIG(refrig))=e=1;
lamps..            sum((lj), XL(lj)) =e=1 ;

spaceheating..     sum(ehcsi, XEHS(ehcsi)) + sum(nehcsi,
XNEHS(nehcsi)) + sum(ehcsi,XEHCS(ehcsi)) +
sum(ehswsi,XEHWS(ehswsi)) + sum(nehswsi, XNEHWS(nehswsi))=e=1 ;
spacecooling..     sum(ecsi, XECS(ecsi)) + sum(ehcsi,
XEHCS(ehcsi)) =e= 1 ;
dhwsystems..       sum(ewsi, XEWS(ewsi)) + sum(newsi,
XNEWS(newsi)) + sum(ehswsi,XEHWS(ehswsi)) + sum(nehswsi,
XNEHWS(nehswsi)) =e= 1 ;

solarcollector..  sum(slci, XSLC(slci)) =l= 1 ;
Photov..           sum(pvs, XPHOTOV(pvs)) =l= 1;

*-----OBJECTIVE 1: COST-----
-----

COSTfordoor_eq..
COSTfordoor=e=Adoor*sum(d,XDOOR(d)* Costdoor(d)) ;
COSTforwindows_eq..
COSTforwindows=e=Awin*sum(wz,XWIN(wz)*CostWindows(wz)) ;
COSTforwalls_eq..          COSTforwalls=e=Awallext*sum(w,
XWALL(w)*CostWalls(w)) ;
COSTforceilings_eq..
COSTforceilings=e=Aceil*sum(r,XCEIL(r)*CostCeiling(r)) ;
COSTforfloors_eq..
COSTforfloors=e=Aflood*sum(h,XFLOOR(h)*CostFloors(h)) ;

```

```

COSTforheatingsystem_eq..
COSTforheatingsystem=e=sum(ehcsi,XEHS(ehcsi)*CostElecHeatSys(ehcsi))+sum(nehcsi,XNEHS(nehcsi)*CostNonElecHeatSys(nehcsi));
COSTforcoolingsystem_eq..
COSTforcoolingsystem=e=sum(ecsci,XECS(ecsci)*CostElecCoolingSys(ecsci));
COSTforwatersystem_eq..
COSTforwatersystem=e=sum(ewsi,XEWS(ewsi)*CostElecDHW(ewsi))+sum(newsi,XNEWS(newsi)*CostNonElecDHW(newsi));
COSTforheatingcooling_eq..
COSTforheatingcooling=e=sum(ehcsi,XEHCS(ehcsi)*CostElecHeatCoolingSys(ehcsi));
COSTforheatingwatersystem_eq..
COSTforheatingwatersystem=e=sum(ehws,XEHWS(ehws)*CostElecHeatingDHW(ehws))+sum(nehws,XNEHWS(nehws)*CostNonElecHeatingDHW(nehws));
COSTforsolar_eq..
COSTforsolar=e=sum(slci,XSLC(slci)*CostSolar(slci));
COSTforlight_eq..
COSTforlight=e=sum(lj,XL(lj)*CostLamps(lj))*Nlamps;
COSTforPhotov_eq..
COSTforPhotov=e=sum(pvs,XPHOTOV(pvs)*CostPhotoV(pvs));
COSTforTV_eq..
COSTforTV=e=sum(tv,XTV(tv)*CostTV(tv));
COSTforcooker_eq..
COSTforcooker=e=sum(cooker,XCOOKER(cooker)*CostCooker(cooker));
;
COSTforWashing_eq..
COSTforWashing=e=sum(washing,XWASHING(washing)*CostWashing(washing));
COSTforRefrig_eq..
COSTforRefrig=e=sum(refrig,XREFRIG(refrig)*CostRefrig(refrig));
;

invcost_eq..          INVCOST =e= COSTforlight +
COSTforwindows + (COSTfordoor + COSTforwalls +
COSTforceilings + COSTforfloors)
                    + (COSTforheatingsystem +
COSTforcoolingsystem + COSTforwatersystem +
COSTforheatingcooling + COSTforheatingwatersystem +
COSTforsolar + COSTforPhotov)
                    + (COSTforTV + COSTforcooker
+ COSTforWashing + COSTforRefrig);

capcost_eq..          TOT_CAPCOST =e=
COSTforlight*annfactor_lamps + COSTforwindows*annfactor_win +
(COSTfordoor + COSTforwalls + COSTforceilings +
COSTforfloors)*annfactor_structure
                    + (COSTforheatingsystem +
COSTforcoolingsystem + COSTforwatersystem +
COSTforheatingcooling + COSTforheatingwatersystem +
COSTforsolar + COSTforPhotov)*annfactor_systems
                    + (COSTforTV + COSTforcooker
+ COSTforWashing + COSTforRefrig)*annfactor_ea;

```



```

tot_opercost_eq.. TOT_OPERCOST =e= OPERCOST_FUEL +
OPERCOST_EL;
opercost_fuel_eq.. OPERCOST_FUEL =e= FUEL_HEAT_NEL_COST +
FUEL_DHW_NEL_COST;
opercost_el_eq.. OPERCOST_EL =e= 1/fconv*cost_elec*Qelgrid;

tot_ann_cost_eq.. TAC =e= TOT_CAPCOST + TOT_OPERCOST;

obj1..          Z('1')=e= TAC;
*----OBJECTIVE 2: ENERGY CONSUMPTION-----*

BLC_eq..          BLC =e=
Adoor*sum(d,XDOOR(d)*Udoors(d))+Awin*sum(wz,XWIN(wz)*
UWindows(wz))+Awallext*sum(w,XWALL(w)*Uwalls(w))+
Aceil*sum(r,XCEIL(r)*UCeiling(r))+
Afloor*sum(h,XFLOOR(h)*UFloor(h));

QTClosses_eq(month).. QTClosses(month)=e=BLC*(Tic-
To(month))*tm(month)/1000 ; //kWh

QCVEN_eq(month).. QCVEN(month)=e=dair*CPair*ACH*V*(Tic-
To(month))*tm(month)/3600 + dair*Hfg*ACH*V*(wi-
Wo(month))*tm(month)/3600; //kWh

QTHlosses_eq(month).. QTHlosses(month)=e=BLC*(Tih-
To(month))*tm(month)/1000; //kWh
QHVEN_eq(month).. QHVEN(month)=e= dair*CPair*ACH*V*(Tih-
To(month))*tm(month)/3600 ; //kWh

QINHG_eq(month)..
QINHG(month)=e=(Npeople*Qhuman/1000)*tm(month)+QEA_HEAT(month)
; //kWh

QSL_eq(month).. QSL(month)=e=sum(wz,XWIN(wz)*Gwin(wz))
* Awin*Ff*ShadingFactorWindows*FCm*Isl(month)*td(month)
; //kWh

QTV_eq(month).. QTV(month)=e=sum(tv,
PowerTV(tv)*XTV(tv))*UsageTV *td(month)/1000 ; //kWh
QCooker_eq(month)..
QCooker(month)=e=sum(cooker,PowerCooker(cooker)*XCOOKER(cooker
))*UsageCooker*td(month)/1000 ; //kWh
Qwash_eq(month).. Qwash(month)=e=td(month)*CyclesWashing
*sum(washing,PowerWashing(washing))*XWASHING(washing) ; //kWh
Qrefrig_eq(month)..
Qrefrig(month)=e=td(month)*sum(refrig,PowerRefrig(refrig))*XREF
RIG(refrig) ; //kWh
Qea_eq(month).. QeaD(month)=e=Fconv*(QTV(month)+
QCooker(month)+Qwash(month)+Qrefrig(month)); //MJ

Q_tv_heat_eq(month).. Q_TV_HEAT(month) =e= QTV(month);
//kWh
Q_cooker_heat_eq(month).. Q_COOKER_HEAT(month) =e=
QCooker(month); //kWh

```

```

Q_wash_heat_eq(month)..      Q_WASH_HEAT(month) =e=
td(month)*CyclesWashing
*sum(washing,PowerWashing(washing)*XWASHING(washing)/Efficienc
yWashing(washing)); //kWh
Q_refrig_eq(month)..         Q_REFRIG_HEAT(month) =e=
td(month)*sum(refrig,PowerRefrig(refrig)*XREFRIG(refrig)/Effic
iencyRefrig(refrig)) ; //kWh
Qea_heat_eq(month)..         QEA_HEAT(month) =e=
Q_TV_HEAT(month) + Q_COOKER_HEAT(month) + Q_WASH_HEAT(month) +
Q_REFRIG_HEAT(month); //kWh , converted to MJ in another eq

QLD_eq(month)..
QLD(month)=e=Fconv*td(month)*Nlamps*UsageLamps*sum(lj,
LampsPower(lj)* XL(lj))/1000 ; //MJ

*-----PV and SLC ANNUAL GENERATION-----*
QPhoto_eq(month)..          QPhoto(month) =e= sum(pvs,
XPHOTOV(pvs)*
AreaPhotoV(pvs)*GenerationEfficiencyPhotoV(pvs))*PRphotov*Shad
ingFactorSolar*Isl(month)*td(month); // kWh
Qalt_eq..                   pvsupply=e=sum(month, QPhoto(month)) ;
//kWh
altenergy_eq..              Qalt =e= pvsupply*Fconv; //MJ

QdSLC_eq(month)..          QdSLC(month)=e=ShadingFactorSolar
*Isl(month)*td(month)*sum(slci, AreaSolar(slci)*XSLC(slci)*
EfficiencySolar (slci)); //kWh
Qdhwu_eq(month)..          Qdhwu(month)=e=Mw*dwater*CPwater*(Tdhw-
Tdcw(month))*tm(month) ; //kWh

*----IF EQUATIONS----*

ifequation1(month)..        QHD(month) =g=
HS(month)*Fconv*(QTHlosses(month)+ QHVEN(month)-QINHG(month)-
QSL(month)); //MJ
ifequation2(month)..        QCD(month) =g=
CS(month)*Fconv*(QINHG(month)+QSL(month)- QTClosses(month)-
QCVEN(month)); //MJ
ifequation3(month)..        DQdhw(month) =g=
WSm(month)*Fconv*(Qdhwu(month)-QdSLC(month)); //MJ

*-----ENERGY -----*
*-----*
QHDtot_eq..                 QHDtot =e= sum(month,QHD(month)) ;
// MJ
QCDtot_eq..                 QCDtot =e= sum(month,QCD(month)) ; //
MJ
QLDtot_eq..                 QLDtot =e= sum(month,QLD(month)); // MJ
QEaDTot_eq..               QEaDTot =e= sum(month,QeaD(month)); //
MJ
QWDtot_eq..                 QWDtot =e= sum(month, DQdhw(month)) ;
// MJ

```

```

QDel_eq..          QDel=e=QHel+QCel+Qwel+QLel+QAel ;
//MJ, electricity demand
ifequation4..     Qelgrid =g= QDel - Qalt; //MJ

QHel_eq..
QHel=e=(y1/EfficiencyElecHeatSys('EHS1'))+
(y2/EfficiencyElecHeatSys('EHS2'))+
(y3/EfficiencyElecHeatSys('EHS3')) +
(y4/EfficiencyElecHeatCoolingSys('EHCS1'))+
(y5/EfficiencyElecHeatCoolingSys('EHCS2'))
+(y6/EfficiencyElecHeatCoolingSys('EHCS3')) +
(y7/EfficiencyElecHeatingDHW('EHWS1'))+
(y8/EfficiencyElecHeatingDHW('EHWS2')) +
(y9/EfficiencyElecHeatingDHW('EHWS3')); //MJ

QHnel_eq..
QHnel=e=(y10/EfficiencyNonElecHeatSys('NEHS1'))+(y11/EfficiencyNonElecHeatSys('NEHS2'))+
(y12/EfficiencyNonElecHeatSys('NEHS3'))+
(y13/EfficiencyNonElecHeatingDHW('NEHWS1'))+
(y14/EfficiencyNonElecHeatingDHW('NEHWS2')) +
(y15/EfficiencyNonElecHeatingDHW('NEHWS3')); //MJ

fuel_heat_nel_cost_eq..      FUEL_HEAT_NEL_COST =e=
y10*cost_oil/(EfficiencyNonElecHeatSys('NEHS1')*lhvNonElecHeatSys('NEHS1'))+y11*cost_ng/(EfficiencyNonElecHeatSys('NEHS2')*lhvNonElecHeatSys('NEHS2'))+
y12*cost_ng/(EfficiencyNonElecHeatSys('NEHS3')*lhvNonElecHeatSys('NEHS3'))+
y13*cost_oil/(EfficiencyNonElecHeatingDHW('NEHWS1')*lhvNonElecHeatingDHW('NEHWS1'))
+
y14*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS2')*lhvNonElecHeatingDHW('NEHWS2')) +
y15*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS3')*lhvNonElecHeatingDHW('NEHWS3')); //euros (based on fuel consumption per m3)

QCel_eq..          QCel=e=
(y16/EfficiencyElecCoolingSys('ECS1'))
+(y17/EfficiencyElecCoolingSys('ECS2'))+
(y18/EfficiencyElecCoolingSys('ECS3'))
+
(y19/EfficiencyElecHeatCoolingSys('EHCS1')) +
(y20/EfficiencyElecHeatCoolingSys('EHCS2')) +
(y21/EfficiencyElecHeatCoolingSys('EHCS3')); //MJ

Qwel_eq..         Qwel=e=(y22/EfficiencyElecDHW('EWS1')) +
(y23/EfficiencyElecDHW('EWS2'))+ (y24/EfficiencyElecDHW('EWS3'))
+

```

```

(y25/EfficiencyElecHeatingDHW('EHWS1')) +
(y26/EfficiencyElecHeatingDHW('EHWS2'))+
(y27/EfficiencyElecHeatingDHW('EHWS3')); //MJ

Qwnel_eq..          Qwnel=e=(y28/EfficiencyNonElecDHW
('NEWS1')) + (y29/EfficiencyNonElecDHW('NEWS2')) +
(y30/EfficiencyNonElecDHW('NEWS3'))+

(y31/EfficiencyNonElecHeatingDHW ('NEHWS1'))+
(y32/EfficiencyNonElecHeatingDHW('NEHWS2')) +
(y33/EfficiencyNonElecHeatingDHW('NEHWS3')); //MJ

fuel_dhw_nel_cost_eq..          FUEL_DHW_NEL_COST =e=
y28*cost_oil/(EfficiencyNonElecDHW ('NEWS1')*lhvNonElecDHW
('NEWS1')) +
y29*cost_oil/(EfficiencyNonElecDHW('NEWS2')*lhvNonElecDHW('NEWS
2')) +

y30*cost_ng/(EfficiencyNonElecDHW('NEWS3')*lhvNonElecDHW('NEWS3
'))+ y31*cost_oil/(EfficiencyNonElecHeatingDHW
('NEHWS1')*lhvNonElecHeatingDHW ('NEHWS1'))+

y32*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS2')*lhvNonElech
eatingDHW('NEHWS2')) +
y33*cost_ng/(EfficiencyNonElecHeatingDHW('NEHWS3')*lhvNonElech
eatingDHW('NEHWS3')); //euros (based on fuel consumption per
m3)

QLel_eq..          QLel=e=QLDtot*SEAel; //MJ
QAel_eq..          QAel=e=QEaDTot*SEAel; //MJ

*----- PRIMARY ENERGY CONSUMPTION-----*
Q_pe_sys_el_eq..          Q_pe_sys_el =e= Qelgrid/ngrid; //MJ
Q_pe_sys_nel_eq..          Q_pe_sys_nel =e= QHnel+Qwnel; //MJ

obj2_energy..          Qt =e= Q_pe_sys_el + Q_pe_sys_nel;
//MJ
obj2..          Z('2')=e=Qt;

*-----LINEARIZATION FOR SYSTEM SELECTION-----*

linear_eq1..          y1=1=mpos*XEHS('EHS1');
linear_eq2..          y1=1=QHDtot ;
linear_eq3..          y1=g=QHDtot-mpos*(1-XEHS('EHS1')) ;
linear_eq4..          y1=g=0 ;

linear_eq5..          y2=1=mpos*XEHS('EHS2');
linear_eq6..          y2=1=QHDtot ;
linear_eq7..          y2=g=QHDtot-mpos*(1-XEHS('EHS2')) ;
linear_eq8..          y2=g=0 ;

linear_eq9..          y3=1=mpos*XEHS('EHS3');
linear_eq10..          y3=1=QHDtot ;

```

```

linear_eq11..      y3=g=QHDtot-mpos*(1-XEHS('EHS3')) ;
linear_eq12..      y3=g=0 ;

linear_eq13..      y4=1=mpos*XEHCS('EHCS1');
linear_eq14..      y4=1=QHDtot ;
linear_eq15..      y4=g=QHDtot-mpos*(1-XEHCS('EHCS1'));
linear_eq16..      y4=g=0 ;

linear_eq17..      y5=1=mpos*XEHCS('EHCS2');
linear_eq18..      y5=1=QHDtot ;
linear_eq19..      y5=g=QHDtot-mpos*(1-XEHCS('EHCS2'));
linear_eq20..      y5=g=0 ;

linear_eq21..      y6=1=mpos*XEHCS('EHCS3');
linear_eq22..      y6=1=QHDtot ;
linear_eq23..      y6=g=QHDtot-mpos*(1-XEHCS('EHCS3'));
linear_eq24..      y6=g=0 ;

linear_eq25..      y7=1=mpos*XEHWS('EHWS1') ;
linear_eq26..      y7=1=QHDtot ;
linear_eq27..      y7=g=QHDtot-mpos*(1-XEHWS('EHWS1')) ;
linear_eq28..      y7=g=0 ;

linear_eq29..      y8=1=mpos*XEHWS('EHWS2') ;
linear_eq30..      y8=1=QHDtot ;
linear_eq31..      y8=g=QHDtot-mpos*(1-XEHWS('EHWS2')) ;
linear_eq32..      y8=g=0 ;

linear_eq33..      y9=1=mpos*XEHWS('EHWS3') ;
linear_eq34..      y9=1=QHDtot ;
linear_eq35..      y9=g=QHDtot-mpos*(1-XEHWS('EHWS3')) ;
linear_eq36..      y9=g=0 ;

linear_eq37..      y10=1=mpos*XNEHS('NEHS1');
linear_eq38..      y10=1=QHDtot ;
linear_eq39..      y10=g=QHDtot-mpos*(1-XNEHS('NEHS1')) ;
linear_eq40..      y10=g=0 ;

linear_eq41..      y11=1=mpos*XNEHS('NEHS2');
linear_eq42..      y11=1=QHDtot ;
linear_eq43..      y11=g=QHDtot-mpos*(1-XNEHS('NEHS2')) ;
linear_eq44..      y11=g=0 ;

linear_eq45..      y12=1=mpos*XNEHS('NEHS3');
linear_eq46..      y12=1=QHDtot ;
linear_eq47..      y12=g=QHDtot-mpos*(1-XNEHS('NEHS3')) ;
linear_eq48..      y12=g=0 ;

linear_eq49..      y13=1=mpos*XNEHWS('NEHWS1') ;
linear_eq50..      y13=1=QHDtot ;
linear_eq51..      y13=g=QHDtot-mpos*(1-XNEHWS('NEHWS1'))
;
linear_eq52..      y13=g=0 ;

linear_eq53..      y14=1=mpos*XNEHWS('NEHWS2') ;
linear_eq54..      y14=1=QHDtot ;

```

```

linear_eq55..      y14=g=QHDtot-mpos*(1-XNEHWS('NEHWS2'))
;
linear_eq56..      y14=g=0 ;

linear_eq57..      y15=l=mpos*XNEHWS('NEHWS3') ;
linear_eq58..      y15=l=QHDtot ;
linear_eq59..      y15=g=QHDtot-mpos*(1-XNEHWS('NEHWS3'))
;
linear_eq60..      y15=g=0 ;

linear_eq61..      y16=l=mpos*XECS('ECS1') ;
linear_eq62..      y16=l=QCDtot ;
linear_eq63..      y16=g=QCDtot-mpos*(1-XECS('ECS1')) ;
linear_eq64..      y16=g=0 ;

linear_eq65..      y17=l=mpos*XECS('ECS2') ;
linear_eq66..      y17=l=QCDtot ;
linear_eq67..      y17=g=QCDtot-mpos*(1-XECS('ECS2')) ;
linear_eq68..      y17=g=0 ;

linear_eq69..      y18=l=mpos*XECS('ECS3') ;
linear_eq70..      y18=l=QCDtot ;
linear_eq71..      y18=g=QCDtot-mpos*(1-XECS('ECS3')) ;
linear_eq72..      y18=g=0 ;

linear_eq73..      y19=l=mpos*XEHCS('EHCS1') ;
linear_eq74..      y19=l=QCDtot ;
linear_eq75..      y19=g=QCDtot-mpos*(1-XEHCS('EHCS1')) ;
linear_eq76..      y19=g=0 ;

linear_eq77..      y20=l=mpos*XEHCS('EHCS2') ;
linear_eq78..      y20=l=QCDtot ;
linear_eq79..      y20=g=QCDtot-mpos*(1-XEHCS('EHCS2')) ;
linear_eq80..      y20=g=0 ;

linear_eq81..      y21=l=mpos*XEHCS('EHCS3') ;
linear_eq82..      y21=l=QCDtot ;
linear_eq83..      y21=g=QCDtot-mpos*(1-XEHCS('EHCS3')) ;
linear_eq84..      y21=g=0 ;

linear_eq85..      y22=l=mpos*XEWS('EWS1') ;
linear_eq86..      y22=l=QWDtot ;
linear_eq87..      y22=g=QWDtot-mpos*(1-XEWS('EWS1')) ;
linear_eq88..      y22=g=0 ;

linear_eq89..      y23=l=mpos*XEWS('EWS2') ;
linear_eq90..      y23=l=QWDtot ;
linear_eq91..      y23=g=QWDtot-mpos*(1-XEWS('EWS2')) ;
linear_eq92..      y23=g=0 ;

linear_eq93..      y24=l=mpos*XEWS('EWS3') ;
linear_eq94..      y24=l=QWDtot ;
linear_eq95..      y24=g=QWDtot-mpos*(1-XEWS('EWS3')) ;
linear_eq96..      y24=g=0 ;

linear_eq97..      y25=l=mpos*XEHWS('EHWS1') ;

```

```

linear_eq98..      y25=l=QWDtot      ;
linear_eq99..      y25=g=QWDtot-mpos*(1-XEHWS('EHWS1')) ;
linear_eq100..     y25=g=0      ;

linear_eq101..     y26=l=mpos*XEHWS('EHWS2')      ;
linear_eq102..     y26=l=QWDtot      ;
linear_eq103..     y26=g=QWDtot-mpos*(1-XEHWS('EHWS2')) ;
linear_eq104..     y26=g=0      ;

linear_eq105..     y27=l=mpos*XEHWS('EHWS3')      ;
linear_eq106..     y27=l=QWDtot      ;
linear_eq107..     y27=g=QWDtot-mpos*(1-XEHWS('EHWS3')) ;
linear_eq108..     y27=g=0      ;

linear_eq109..     y28=l=mpos*XNEWS('NEWS1')      ;
linear_eq110..     y28=l=QWDtot      ;
linear_eq111..     y28=g=QWDtot-mpos*(1-XNEWS('NEWS1')) ;
linear_eq112..     y28=g=0      ;

linear_eq113..     y29=l=mpos*XNEWS('NEWS2')      ;
linear_eq114..     y29=l=QWDtot      ;
linear_eq115..     y29=g=QWDtot-mpos*(1-XNEWS('NEWS2')) ;
linear_eq116..     y29=g=0      ;

linear_eq117..     y30=l=mpos*XNEWS('NEWS3')      ;
linear_eq118..     y30=l=QWDtot      ;
linear_eq119..     y30=g=QWDtot-mpos*(1-XNEWS('NEWS3')) ;
linear_eq120..     y30=g=0      ;

linear_eq121..     y31=l=mpos*XNEHWS('NEHWS1')      ;
linear_eq122..     y31=l=QWDtot      ;
linear_eq123..     y31=g=QWDtot-mpos*(1-XNEHWS('NEHWS1'))
;
linear_eq124..     y31=g=0      ;

linear_eq125..     y32=l=mpos*XNEHWS('NEHWS2')      ;
linear_eq126..     y32=l=QWDtot      ;
linear_eq127..     y32=g=QWDtot-mpos*(1-XNEHWS('NEHWS2'))
;
linear_eq128..     y32=g=0      ;

linear_eq129..     y33=l=mpos*XNEHWS('NEHWS3')      ;
linear_eq130..     y33=l=QWDtot      ;
linear_eq131..     y33=g=QWDtot-mpos*(1-XNEHWS('NEHWS3'))
;
linear_eq132..     y33=g=0      ;

model example /all/;
*option mip=gurobi;
annfactor_systems=rate*((1+rate)**life_systems)/(((1+rate)**li
fe_systems)-1);
annfactor_structure=rate*((1+rate)**life_structure)/(((1+rate)
**life_structure)-1);
annfactor_win=rate*((1+rate)**life_win)/(((1+rate)**life_win)-
1);

```

```
annfactor_lamps=rate*((1+rate)**life_lamps)/(((1+rate)**life_lamps)-1);
annfactor_ea=rate*((1+rate)**life_ea)/(((1+rate)**life_ea)-1);
```

```
*-----
-----
```

```
$STitle eps-constraint method
```

```
Set k1(k) the first element of k, km1(k) all but the first
elements of k;
k1(k)$ (ord(k)=1) = yes; km1(k)=yes; km1(k1) = no;
Set kk(k) active objective function in constraint allobj
set diff different objective function values per problem
/1*1000/
```

```
Parameter
```

```
rhs(k) right hand side of the constrained obj
functions in eps-constraint
maxobj(k) maximum value from the payoff table
minobj(k) minimum value from the payoff table
intervals(k) number of intervals that we divide the k-1
objective functions
bestobj(k) the best objective function value (maxobj for
dir=1 minobj for dir=-1)
worstobj(k) the worst objective function value (minobj for
dir=1 maxobj for dir=-1)
step(k) the step obtained from range divided by
intervals
jump(k) the jump for augmecon2
```

```
* added on 13/08/13
```

```
*****
*****
```

```
loopworst(k) the worst value from the loop
```

```
*****
*****
```

```
* added on 17/09/13
```

```
*****
*****
```

```
progress(k) the progress for loop k
```

```
*****
*****
```

```
zdiff(diff,k) values of objective functions in different POS
```

```
zstar(k) auxiliary parameter for the comparisons;
```

```
zdiff(diff,k)=0;
```

```
zstar(k)=0;
```

```
Scalar
```

```
iter total number of iterations
```

```
infeas total number of infeasibilities
```

```
elapsed_time elapsed time for payoff and e-constraint
```



```

start start time
finish finish time
summax auxiliary parameter
firstOffMax, lastZero some counters
numdiff number of different POS
dcount counter for different POS
issamenum auxiliary parameter counting how many obj. values
are same with already found

Variables
  a_objval auxiliary variable for the objective function
  obj      auxiliary variable during the construction of
the payoff table
Positive Variables
  sl(k)    slack or surplus variables for the eps-
constraints
Equations
  con_obj(k) constrained objective functions
  augm_obj  augmented objective function to avoid weakly
efficient solutions
  allobj    all the objective functions in one expression;

con_obj(km1)..  z(km1) - dir(km1)*sl(km1) =e= rhs(km1);

* We optimize the first objective function and put the others
as constraints
* the second term is for avoiding weakly efficient points
*augm_obj..
*  sum(k1,dir(k1)*z(k1))+1e-3*sum(km1,sl(km1)/(maxobj(km1)-
minobj(km1))) =e= a_objval;
augm_obj..
  sum(k$(ord(k)=1),dir(k)*z(k)) +
sum(k$(ord(k)>1),power(10,-(ord(k)-1))*sl(k)/(maxobj(k)-
minobj(k))) =e= a_objval;
*  sum(k$(ord(k)=1),dir(k)*z(k)) + 1.0e-
3*sum(k$(ord(k)>1),power(10,-(ord(k)-1))*sl(k)/(maxobj(k)-
minobj(k))) =e= a_objval;

allobj..  sum(kk, dir(kk)*z(kk)) =e= obj;

Model mod_payoff / example, allobj / ;
Model mod_epsmethod / example, con_obj, augm_obj / ;

Parameter
  payoff(k,k) payoff tables entries;
Alias(k,kp);

option optcr=0.000;

* Generate payoff table applying lexicographic optimization
loop(kp,
  kk(kp)=yes;
  repeat
    solve mod_payoff using mip maximizing obj;
    payoff(kp,kk) = z.l(kk);

```

```

    z.fx(kk) = z.l(kk); // freeze the value of the last
objective optimized
    kk(k+1) = kk(k); // cycle through the objective
functions
    until kk(kp); kk(kp) = no;
* release the fixed values of the objective functions for the
new iteration
    z.up(k) = inf; z.lo(k) =-inf;
);
if (mod_payoff.modelstat<>1 and mod_payoff.modelstat<>8, abort
'no optimal solution for mod_payoff');

File fx / c:\gams\ziozas_dipl_log.out /;
File dx / c:\gams\ziozas_dipl.out /;

PUT dx ' PAYOFF TABLE' / ;
loop (kp,
    loop(k, put dx payoff(kp,k):12:2);
    put dx /;
);
put dx /;

*new 28-11-2013 for finding the actual range
$ontext
loop(kp,
    kk(kp)=yes;
    solve mod_payoff using mip minimizing obj;
    if (dir(kp)=1,
        maxobj(kp)=smax(k,payoff(k,kp));
        minobj(kp)=obj.l
    );
    if (dir(kp)=-1,
        maxobj(kp)=-obj.l;
        minobj(kp)=smin(k,payoff(k,kp))
    );
    kk(kp)=no;
);
$offtext

*display payoff;
minobj(k)=smin(kp,payoff(kp,k));
*minobj(k)=0;
maxobj(k)=smax(kp,payoff(kp,k));

*-----
*-----
*new 17.03.2013
*-----
*-----
*loop(k, intervals(k)=maxobj(k)-minobj(k));
loop(k, intervals(k)=20);

loop(k,
    if (dir(k)=1,
        bestobj(k)=maxobj(k);
        worstobj(k)=minobj(k);

```

```

else
    bestobj(k)=minobj(k);
    worstobj(k)=maxobj(k)
    );
step(k)=(maxobj(k)-minobj(k))/intervals(k)
);

rhs(k)=worstobj(k);
iter=0;
infeas=0;
start=jnow;

// added 2/3/14 for printing only the different values. New
items are:
numdiff=0; // number of different POS
issamenum=0; // counter for objective function values in the
comparison with the recently found
infeas=0;
zdiff(diff,k)=0;
zstar(k)=0;

*new 13/08/13
*****
*****
loopworst(k)=bestobj(k);
*****
*****
Put fx ' #'
loop(k$(ord(k)>1), put fx ' Progress', ord(k):1:0);
loop(k, put fx ' OBJ', ord(k):1:0);
put fx ' ';
loop(k$(ord(k)>1), put fx ' SLACK', ord(k):1:0);
put fx /;

dx.pw=20000;
put dx ' # TAC ENERGY ' ;
put ' TOT_CAPCOST TOT_OPERCOST INVCOST
QHel QHnel QCel';
put ' Qwel Qwnel QLe1 QAel ' ;
loop(d, put ' DOOR', ord(d):1:0);
loop(wz, put ' WIND', ord(wz):1:0);
loop(w, put ' WALL', ord(w):1:0);
loop(r, put ' CEIL', ord(r):1:0);
loop(h, put ' FLOOR', ord(h):1:0);
loop(ehsi, put ' EHS', ord(ehsi):1:0);
loop(nehsi, put ' NEHS', ord(nehsi):1:0);
loop(ecsi, put ' ECS', ord(ecsi):1:0);
loop(ewsi, put ' EWS', ord(ewsi):1:0);
loop(newsi, put ' NEWS', ord(newsi):1:0);
loop(ehcsi, put ' EHCS', ord(ehcsi):1:0);
loop(ehws, put ' EHWS', ord(ehws):1:0);
loop(nehws, put ' NEHWS', ord(nehws):1:0);
loop(slci, put ' SLC', ord(slci):1:0);
loop(lj, put ' L', ord(lj):1:0);
loop(tv, put ' TV', ord(tv):1:0);
loop(cooker, put ' COOK', ord(cooker):1:0);

```

```

loop(washing,put ' WASH',ord(washing):1:0);
loop(refrig,put ' REFRIG',ord(refrig):1:0);
loop(pvs, put ' PHOTOV',ord(pvs):1:0);
put/;

repeat
  solve mod_epsmethod maximizing a_objval using mip;
  iter=iter+1;
  if (mod_epsmethod.modelstat<>1 and mod_payoff.modelstat<>8,
// not optimal is in this case infeasible
  infeas=infeas+1;
  put fx iter:5:0, ' infeasible'/;
  lastZero = 0;
  loop(k$(ord(k)>1),
    if(abs(rhs(k)-worstobj(k))>0.000001 and lastzero=0,
lastzero=ord(k)
    );
  loop(k$(ord(k)>1 and ord(k)<=lastzero),rhs(k)=bestobj(k));
  else
* update the worst value found so far - added 13/08/13
*****
*****
  loop(k$(ord(k)>2),
    if(dir(k)=1 and z.L(k)<loopworst(k),
loopworst(k)=z.L(k));
    if(dir(k)=-1 and z.L(k)>loopworst(k),
loopworst(k)=z.L(k));
  );
*****
*****
  put fx iter:7:0;
* added on 17.09.13 for reporting
*****
*****
loop(k$(ord(k)>1),
  if (dir(k)=1, progress(k)=100*(rhs(k)-
minobj(k))/(maxobj(k)-minobj(k))
  else progress(k)=100*(maxobj(k)-rhs(k))/(maxobj(k)-
minobj(k))
  );
  put fx progress(k):10:2,'%';
  );
*****
*****
  loop(k, put fx z.l(k):12:2);
  put fx ' *** ' ; // put /;
  loop(km1, put fx sl.l(km1):12:2);

//*****
*****
// added 2/3/14 for printing only the different values. New
items are:
// set diff /1*1000/
// parameters zdifff(diff,k), zstar(k)

```

```

// scalars dcount, issamenum,numdiff

for (dcount=1 to numdiff,
    loop(diff$(ord(diff)=dcount), loop(k,
zstar(k)=zdifff(diff,k)));
    issamenum=0;
    loop(k,
        if(abs(z.L(k)-zstar(k))<0.001,
issamenum=issamenum+1);
        );
        if (issamenum=card(k), dcount=numdiff); //in order to
exit the loop
    ); // for loop
    if (issamenum<card(k),
        numdiff=numdiff+1;
        loop(diff$(ord(diff)=numdiff), loop(k,
zdifff(diff,k)=z.L(k)));
        put dx numdiff:5:0;
        loop(k, put dx z.l(k):12:2);
        put dx ' ';
        put dx TOT_CAPCOST.1:9:2;
        put dx ' ';
        put dx TOT_OPERCOST.1:9:2;
        put dx ' ';
        put dx ' ';
        put dx INVCOST.1:9:2;
        put dx ' ';
        put dx QHel.1:9:2;
        put dx ' ';
        put dx QHnel.1:9:2;
        put dx ' ';
        put dx QCel.1:9:2;
        put dx ' ';
        put dx Qwel.1:9:2;
        put dx ' ';
        put dx Qwnel.1:9:2;
        put dx ' ';
        put dx QLel.1:9:2;
        put dx ' ';
        put dx QAel.1:9:2;
        put dx ' ';
        loop(d, put dx XDOOR.1(d):9:0);
        loop(wz, put dx XWIN.1(wz):9:0);
        loop(w, put dx XWALL.1(w):9:0);
        loop(r, put dx XCEIL.1(r):9:0);
        loop(h, put dx XFLOOR.1(h):9:0);
        loop(ehcsi, put dx XEHS.1(ehcsi):9:0);
        loop(nehcsi, put dx XNEHS.1(nehcsi):9:0);
        loop(ecsi, put dx XECS.1(ecsi):9:0);
        loop(ewsi, put dx XEWS.1(ewsi):9:0);
        loop(newsi, put dx XNEWS.1(newsi):9:0);
        loop(ehcsi, put dx XEHCS.1(ehcsi):9:0);
        loop(ehws, put dx XEHWS.1(ehws):9:0);
        loop(nehws, put dx XNEHWS.1(nehws):9:0);
        loop(slci, put dx XSLC.1(slci):9:0);

```

```

loop(lj, put dx XL.l(lj):9:0);
loop(tv, put dx XTV.l(tv):9:0);
loop(cooker, put dx XCOOKER.l(cooker):9:0);
loop(washing, put dx XWASHING.l(washing):9:0);
loop(refrig, put dx XREFRIG.l(refrig):9:0);
loop(pvs, put dx XPHOTOV.l(pvs):9:0);

*
);

*
put dx (sum(p, cost(p)*X.L(p)):10:0,
(sum(p, X.L(p)):5:0);
*
loop(p, put dx X.L(p):4:0);
put /;
);

*****
*****

*
the jump is for AUGMECON2
jump(k)=1;
*
The jump is calculated for the innermost objective
function (ord(k)=2)
jump(k)$(ord(k)=2) = 1+abs(floor(sl.L(k)/step(k)));
loop(k$(jump(k)>1), put fx ' jump');
put fx /;
);

*
Proceed forward in the grid
firstOffMax = 0;
loop(k$(ord(k)>1),
if(abs(rhs(k)-bestobj(k))>0.000001 and firstOffMax=0,
if (dir(k)=1,
rhs(k)=min((rhs(k)+jump(k)*step(k)), bestobj(k))
else rhs(k)=max((rhs(k)-jump(k)*step(k)), bestobj(k))
);
firstOffMax=ord(k)
);
);

loop(k$(ord(k)>1),
if(ord(k)< firstOffMax, rhs(k)=worstobj(k))
);
*new 13/08/13 go to the worst value from the previous loop of
k-1
*****
*****
loop(k$(ord(k)>1),
if (ord(k)=firstOffMax and firstOffMax > 2,
*Florios add 27.11.2013
if (dir(k)=1, rhs(k)=loopworst(k)+step(k);
loopworst(k)=bestobj(k);
else
rhs(k)=loopworst(k)-step(k);
loopworst(k)=bestobj(k);)
*end Florios add 27.11.2013
**
rhs(k)=loopworst(k)+1;
**
loopworst(k)=bestobj(k);
);

```

```

    );
*****
*****
    summax=0;
    loop(k$(ord(k)>1),
        if(abs(rhs(k)-bestobj(k))<0.000001, summax=summax+1);
    );
* for check premature end
* if (iter = 120,
*     summax=card(k)-1;
*     firstOffMax=0;
* );
until summax=card(k)-1 and firstOffMax=0;

finish=jnow;
elapsed_time=(finish-start)*86400;

put /;
put 'Infeasibilities = ', infeas:5:0 /;
put 'Elapsed time: ',elapsed_time:7:2, ' seconds' / ;

putclose fx; // close the point file
putclose dx; // close the point file

```