



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

Μοντελοποίηση ενεργειακής κατανάλωσης σπιτιών

Πτυχιακή εργασία

Σωτήρης Καλαμπόκης

Αθήνα, 2020



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

**Ηρακλής Βαρλάμης, Δημοσθένης Αναγνωστόπουλος, Γεώργιος
Δημητρακόπουλος**

Ηρακλής Βαρλάμης, Αναπληρωτής Καθηγητής του τμήματος Πληροφορικής και
Τηλεματικής του Χαροκοπείου Πανεπιστημίου

Δημοσθένης Αναγνωστόπουλος, Καθηγητής του τμήματος Πληροφορικής και
Τηλεματικής του Χαροκοπείου Πανεπιστημίου

Γεώργιος Δημητρακόπουλος, Επίκουρος Καθηγητής του τμήματος Πληροφορικής
και Τηλεματικής του Χαροκοπείου Πανεπιστημίου

Ο Σωτήρης Καλαμαπόκης

δηλώνω υπεύθυνα ότι:

- 1) Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλλει τα πνευματικά δικαιώματα τρίτων.
- 2) Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

Κεφαλαίο 1 : Εισαγωγή

1.1 Εισαγωγή

1.2 Περιγραφή εργασίας

Κεφαλαίο 2 : Σχετικές εργασίες

2.1 Simulating human activities to investigate household energy consumption

2.2 A Method for Modeling Household Occupant Behavior to Simulate Residential Energy Consumption

2.3 Simulating Appliance-Based Power Consumption Records for Energy Efficiency Awareness

Κεφαλαίο 3: Λειτουργία και Μεταβλητές

3.1 Μεταβλητές Καταχώρησης

3.2 Μεταβλητές Κατανάλωσης

3.3 Μεταβλητές Χρόνου

3.4 Περιγραφή Λειτουργίας

Κεφαλαίο 4 Περίπτωσης και Ολική Κατανάλωση ανά Περίπτωση

4.1 Περίπτωσης Κατανάλωση

4.2 Ολική Κατανάλωση

Κεφαλαίο 5 Συμπεράσματα

Κεφαλαίο 6 Αναφορές

Κεφαλαίο 1 : Εισαγωγή

1.1 Εισαγωγή

Χρησιμοποιώντας μεγάλα δεδομένα μπορούμε να εξάγουμε πρότυπα για την κατανάλωση ρεύματος ενός χρήστη και μέσα από αυτά να καταλάβουμε ποσό αποτελεσματική ή καταχρηστική είναι η χρήση κάθε συσκευής στο σπίτι του. Υπάρχουν δυο τρόποι για να πάρουμε δεδομένα από κάθε σπίτι: πρώτος είναι με βάση τη μακροοικονομία του σπιτιού, το κόστος του ρεύματος και το κλίμα της περιοχής, και δεύτερος χρησιμοποιώντας πολλά σπίτια από την ίδια περιοχή. Πιο αποτελεσματικός τρόπος είναι με χρήση συσκευών μέτρησης για κάθε συσκευή μέσα σε ένα σπίτι. Καθώς δεν είναι πάντοτε εφικτό να χρησιμοποιούμε πλήθος αισθητήρων και μετρητών για να παρακολουθούμε διαρκώς κάθε συσκευή, χρησιμοποιούμε προσομοιωτές ώστε να παράγουμε δεδομένα που μπορούμε να χρησιμοποιούμε και να μελετήσουμε για να βελτιώσουμε την κατανάλωση κάθε σπιτιού.

Υπερέχουν πολλοί τρόποι να προσδιορίσουμε και να καταγραφούμε την κατανάλωση ρεύματος σε ένα σπίτι. Στην προσπάθεια να δημιουργήσουμε δεδομένα κατανάλωσης που είναι κοντά σε πραγματικά δεδομένα χρησιμοποιούμε γεννήτριες δημιουργίας δεδομένων που μας βοηθούν να δημιουργήσουμε δεδομένα κοντά στα αληθινά καθώς αυτά είναι δύσκολα να καταγραφούν εξαρχής.

Για να μπορέσουμε να προσομοιάσουμε τις συνήθειες χρήσης των συσκευών και να δημιουργήσουμε συνθετικά σύνολα δεδομένων κατανάλωσης, σχεδιάσαμε μια γεννήτρια δεδομένων η οποία δημιουργεί δεδομένα κατανάλωσης ενέργειας για διαφορετικές ρυθμίσεις κατανάλωσης με βάση χρονικές περιόδους για κάθε συσκευή. Όπως καιρικές συνθήκες, αλλαγές στην θερμοκρασία, απασχόλησης του δωματίου αυτές είναι μερικές από τις μεταβλητές του, μπορούμε να χρησιμοποιήσουμε για την γεννήτρια δεδομένων ώστε να έχουμε πιο ακριβεί δεδομένα σε σχέση με την κατανάλωση ανά συσκευή σε ένα δωμάτιο ή σπίτι.

Με την χρήση γεννητριών μπορούμε να δημιουργήσουμε διαφορετικές καταστάσεις κατανάλωσης για διαφορετικά σπίτια άπλα καθορίζοντας τις μεταβλητές της κατανομής που χρησιμοποιούσαμε για να δημιουργήσουμε καταστάσεις κατανάλωσης όπως υπερκατανάλωση, εξοικονόμηση ενέργειας και της κανονικής κατανάλωσης μόνο με μερικές αλλαγές στην γεννήτρια. Όπως και στις καιρικές συνθήκες στην ώρα λειτουργίας και αναμονή μιας συσκευής και στην απασχόληση του δωματίου για να παρατηρήσουμε και να εξάγουμε διαφορετικά δεδομένα από κάθε περίπτωση.

Παρατηρώντας τα δεδομένα που περνούμε μπορούμε να κατανέμουμε διαφορετικές καταστάσεις και να τις αξιολογήσουμε ανάλογα ώστε να μπορέσουμε να δημιουργήσουμε πρότυπα κατανάλωσης για σπίτια που είναι κοντά στα δεδομένα που μας προμηθεύει η γεννήτρια ακόμα και αν αυτά δεν είναι πραγματικά δεδομένα. Με βάση αυτό μπορούμε να προτείνουμε τρόπους εξοικονόμησης ενέργειας ή καλύτερους τρόπους χρήσης συγκεκριμένων συσκευών.

Χρησιμοποιώντας χρονικές στιγμές για να μπορέσουμε να προσδιορίσουμε την κατανάλωση σε χρονικές στιγμές ώστε να είναι πιο κατανοητά τα δεδομένα και πιο συγκεκριμένα. Με χρονικές στιγμές των είκοσι λεπτών που χρησιμοποιήθηκαν σε αυτήν την εργασία. Οπου τα λεπτά είναι και η βάση για τις κατανομές που χρησιμοποιήθηκαν ανάλογα με την κατάσταση μελέτης, η βάση αλλάζει έχοντας πάντα είκοσι λεπτά λειτουργίας. Για παράδειγμα η κατάσταση της εξοικονόμησης ενέργειας έχει βάση το δεκαεννιά άλλα εξακολούθησε να έχει χρονικές στιγμές των είκοσι λεπτών

Για τις θερμοκρασίες κάθε εκτέλεσης και κάθε συσκευής χρησιμοποιούσαμε ομαδοποίηση τριών μηνών σε εποχές για να μπορέσουμε να προσδιορίσουμε την κανονική κατανομή και να πάρουμε μια τιμή που έχει μια κύμανση κοντά στην μέση τιμή της εποχής. Μπορούμε να προσομοιάσουμε διαφορετικές συσκευές σε ένα δωμάτιο για

διαφορετικές εποχές του χρόνου ακόμα και αν αυτές είναι στο ίδιο δωμάτιο μπορούμε να πούμε ότι η X συσκευή τρέχει τον χειμώνα και η Y το καλοκαίρι.

Τα δεδομένα που δημιουργήθηκαν είναι κοντά στα πραγματικά αλλά δεν είναι πραγματικά είναι δεδομένα που προέρχονται από μια γενέτειρα αρκετά από αυτά είναι εκτός κανονικότητας αλλά πια πραγματικά δεδομένα ακολουθούν μια κατανομή ή μια συνάρτηση;

1.2 Περίληψη εργασίας

Με χρήση γνωστών μεταβλητών όπως η κατανάλωση ρεύματος ανά ώρα, καιρικές συνθήκες, απασχόληση του δωματίου έγινε η προσπάθεια να δημιουργήσουμε μια προσομοίωση όπου θα μελετάτε η κατανάλωση του ρεύματος κάθε συσκευής σε ένα σπίτι. Με χρήση κατανομών για την δημιουργία όλων των δεδομένων για να διασφαλίσουμε την τυχαία διανομή των δεδομένων κατανάλωσης. Όλα τα δεδομένα εισαγωγής και εξαγωγής δηλώνονται σε αρχεία JSON για να γίνει η χρήση τους σε άλλα προγράμματα.

Μπορούμε να ομαδοποιούμε δεδομένα με την βοήθεια μεταβλητών για να μπορέσουμε να τα αναζητήσουμε μετά την εκτέλεση του προγράμματος. Καθώς και να δώσουμε μοναδικές τιμές ώστε να γίνει αναζήτηση σε συγκεκριμένες συσκευές.

Σε 3 διαφορετικές καταστάσεις κατανάλωσης η οποία είναι κανονική κατανάλωση ρεύματος, υπερκατανάλωση ρεύματος και εξοικονόμηση ρεύματος. Χρησιμοποιήθηκαν χρονικές στιγμές των είκοσι λεπτών με μέτρησις ανά μέρα και κατάστασις συσκευών όπου αλλάζουμε την βάση της κανονικής κατανομής σε είκοσι δύο για την υπερκατανάλωση και δεκαεννιά για την εξοικονόμηση ενέργειας για να πραγματοποιηθεί η προσομοίωση.

Για την προσομοίωση των καιρικών συνθηκών χρησιμοποιούμε μια μεταβλητή που δίνετε από τον χρήστη για να προσδιορίσουμε την εποχή και με βάση την μέση θερμοκρασία της εποχής για την κανονική κατανομή, μας δίνετε μια θερμοκρασία κοντά στην μέση τιμή.

Στο τέλος μπορούμε να παρατηρήσουμε ότι κάθε κατάσταση έχει περίπου τις προβλεπόμενες τιμές που της αναλογούν, άλλα και ότι αν οι στιγμές που τρέχει μια συσκευή μέσα σε μια μέρα είναι ελάχιστες τότε μπορούμε να δούμε ότι κάποιες καταστάσεις ξεφεύγουν από τα κανονικά ή αναμενόμενα αποτελέσματα.

Κεφαλαίο 2 : Σχετικές εργασίες

2.1 Simulating human activities to investigate household energy consumption[1]

Μια μελέτη σχετικά με τις δραστηριότητες κάθε ατόμου και την επίδραση αυτών στην κατανάλωση ρεύματος ανάλογα με την δραστηριότητα με μεταβλητές το ρυθμό που γίνονται οι δραστηριότητες και αν υπάρχει κάποιο γεγονός όπου κάποιες δραστηριότητες γίνονται ή όχι. Χρησιμοποιούνται AI για την προσομοίωση, ανθρωπίνων συμπεριφορών για καθημερινές δραστηριότητες.

2.2 A Method for Modeling Household Occupant Behavior to Simulate Residential Energy Consumption[2]

Μια μελέτη με πραγματικά δεδομένα για το αν η παρουσία επηρεάζει την κατανάλωση ρεύματος ανάλογα με την ηλικία, φύλο και εργασία κάθε ατόμου με χρήση Markov αλυσίδων. Χρησιμοποιήθηκαν 10 διαφορετικές ενέργειες που μπορεί να κάνει κάθε άτομο ανά στιγμή.

Μελετώντας την κάθε συσκευή ξεχωριστά για να βρούμε το σύνολο της κατανάλωσης του σπίτι. Προσπαθώντας να καταλάβουμε ποιος είναι ο ιδανικός αριθμός προσομοίωσης για να έχουμε δεδομένα κοντά σε πραγματικά δεδομένα. Ωστε να χρησιμοποιούν για την δημιουργία δεδομένων κατανάλωση ρεύματος. Οπού βρήκανε ότι ο καλύτερος αριθμός κάτοικων είναι 100 και κατοικημένων σπιτιών 40.

2.3 Simulating Appliance-Based Power Consumption Records for Energy Efficiency Awareness[3]

2.3.1

Σε αυτήν την μελέτη έγινε προσπάθεια να δημιουργηθούν δεδομένα κατανάλωσης για να χρησιμοποιηθούν για μελέτη δεδομένων και βελτιστοποίηση κατανάλωσης. Χρησιμοποιήθηκαν γεννήτριες δεδομένων για να δημιουργία τα δεδομένα, με δεδομένα την κατανάλωση, την απασχόληση του δωματίου και πέντε διαφορετικές κατάστασης κατανάλωσης.

2.3.2

Πρώτα πρέπει να καθορίσουμε ποσά δωμάτια θα χρησιμοποιούμε για την δημιουργία των δεδομένων μετά την κατανάλωση ανά συσκευή με min/max και κατάσταση αναμονής οι οποίες μπορούν να δοθούν από τον κατασκευαστή των συσκευών. Ύστερα πρέπει να ορίσουμε προφίλ χρήστη για κάθε συσκευή με βάση τον χρόνο χρήσης του στο εικοσιτετράωρο για κάθε λειτουργία της συσκευής . Μετά πρέπει να καθορίσουμε αν κάποιος είναι στο δωμάτιο. Τώρα δημηγορούμε τυχαία τιμές κατανάλωσης για κάθε χρονική περίοδο και ομοιόμορφα μεταξύ λειτουργίας και αναμονής και 4 προς 1 μεταξύ βαρείας χρήσης και κανονικής χρήσης.

Μικρό καταστάσεις χρησιμοποιούνται για την κατανόηση της κατανάλωσης και είναι η έξις: καλή κατανάλωση, ενεργεί κατάσταση, ανενεργή κατάσταση , υπερκατανάλωση , κατανάλωση εκτός δωματίου και χρησιμοποιούνται για να γίνει καλύτερη κατανόηση των δεδομένων που παράγονται . Δεν χρησιμοποιείτε ο καιρός ούτε οι καιρικές συνθήκες για την δημιουργία των δεδομένων .

Κεφαλαίο 3

3.1 Μεταβλητές Καταχώρησης

devid. Χρησιμοποιείτε για να προσδιορίσει τον αριθμό των συσκευών που ζητάμε να μοντελοποιηθούν σε κάθε εκτέλεση. Είναι μοναδικός αριθμός για κάθε συσκευή και ξεκινά από το ένα μέχρι X (οπού X ο αριθμός όλων των συσκευών). Ακόμα και αν μια συσκευή είναι ορισμένη περισσότερες από μια φορές πρέπει να έχει μοναδικό αριθμό καθώς το id είναι μοναδικό για κάθε συσκευή και χρησιμοποιείτε για την εύρεση αυτής στα δεδομένα που δημιουργούνται.

devtype. Είναι ο τύπος της συσκευής (π.χ. ψυγείο, κουζίνα). Μπορούν να υπάρχουν πολλές συσκευές με τον ίδιο τύπο και διαφορετικές ή και ίδιες τιμές αυτή η μεταβλητή βοηθάει να βρεθούν όλες αυτές οι συσκευές με τον ίδιο τύπο πιο εύκολα. Μπορούμε να το θεωρήσουμε και σαν ομαδοποιήσει συσκευών (π.χ. δωμάτιο 1, δωμάτιο 2) και να καταχωρίσουμε πολλές συσκευές κάτω από μια ομάδα ώστε να μπορούμε να τις αναζητήσουμε στα αποτελέσματα του προγράμματος.

period. Είναι μια από τις τεσσάρες εποχές του χρόνου που κάθε συσκευή προσομοιώνεται. Ξεκινάει από 1 οπού είναι ο χειμώνας 2 είναι η άνοιξη 3 είναι το καλοκαίρι και 4 τον φθινόπωρο με βάση αυτής της μεταβλητής μπορούμε να πάρουμε την θερμοκρασία έξω από το σπίτι ή κτίριο που γίνονται οι μέτρησις. Το πρόγραμμα έχει σταθερές τιμές για κάθε εποχή οπού αυτές είναι οι μέσες τιμές της θερμοκρασίας ανά εποχή για την Αθηνά. Αφού πάρουμε την τιμή της μεταβλητής χρησιμοποιούμε μια κανονική κατανομή με ± 2 και βάση την εποχική θερμοκρασία για να βρούμε την τιμή της προσομοίωσης.

3.2 Μεταβλητές Κατανάλωσης

devkw. Είναι η κατανάλωση λειτουργίας ρεύματος ανά ώρα κάθε συσκευής. Στην συνεχεία διαιρείτε με το εξήντα για να βρεθεί η κατανάλωση ανά λεπτό και το πολλαπλασιάζουμε με το αποτέλεσμά της κανονικής κατανομής με χρονική περίοδο που έχει οριστεί ως 20 λεπτά σε αυτό το πρόγραμμα. Με χρήση κατανομών βρίσκουμε την ακριβή χρονική διάρκεια που μια συσκευή έτρεξε και πολλαπλασιάζουμε με το αποτέλεσμα με την κατανάλωση που θα βρούμε χρησιμοποιώντας την συνάρτηση $(devkw/60)*distributionResults$ και αυτό θα μας δώσει την κατανάλωση ανά χρονική περίοδο.

stbkw. Είναι η κατανάλωση αναμονής ρεύματος ανά ώρα κάθε συσκευής. Στην συνεχεία διαιρείτε με το εξήντα για να βρεθεί η κατανάλωση ανά λεπτό και το πολλαπλασιάζουμε με το αποτέλεσμά της κανονικής κατανομής με χρονική περίοδο που έχει οριστεί ως 20 λεπτά σε αυτό το πρόγραμμα. Με χρήση κατανομών βρίσκουμε την ακριβή χρονική διάρκεια που μια συσκευή έτρεξε και πολλαπλασιάζουμε με τον αποτέλεσμα με την κατανάλωση που θα βρούμε χρησιμοποιώντας την συνάρτηση $(stbkw/60)*distributionResults$ και αυτό θα μας δώσει την κατανάλωση ανά χρονική περίοδο.

3.3 Μεταβλητές Χρόνου

5)timesrun. Δίνουμε πόσες φορές θα τρέξει η κάθε συσκευή, με μέγιστο το 73 καθώς έχουμε χωρίσει την χρονική διάρκεια σε 20 λεπτά. Αναφέρετε και στις δυο κατάστασης λειτουργίας OnHours και STBHours το πρόγραμμα επιλέγει τυχαία από της 73 πιθανές τιμές ανεξάρτητα της κατάστασης άρα υπάρχει πιθανότητα να τρέχει μια συσκευή μονό σε STBHours ή μονό σε OnHours δίνοντας σε διαφορετικές εκτέλεσις διαφορετικές τελικές τιμές αν δεν οριστεί σε μεγάλη τιμή.

7)onhours, stbhours. Χρησιμοποιείτε για να καθορίσει σε τη κατάσταση βρίσκετε κάθε συσκευή ανά εξάωρο με χρήση ενός πίνακα μεταβλητών από 0 και 1 με μέγιστο μεταξύ των δυο μεταβλητών το τέσσερα μοναδικά 1 μοναδικά σε κάθε στήλη δηλαδή αν το onhours έχει 1 στην θέση μηδέν το stbhours δεν μπορεί να έχει 1 στην θέση μηδέν. Μπορούμε να έχουμε λιγότερες από τεσσάρους μεταβλητές ως 1 με ρίσκο να χάσουμε timesrun.

8)onhouse Μια ένδειξη αν ο χρήστης είναι μέσα στο σπίτι ή όχι ανά εξάωρο με χρήση ενός πίνακα με τεσσάρους πιθανές τιμές για κάθε εξάωρο . Οπού 1 δηλώνουμε ότι ο χρήστης είναι στο σπίτι του και 0 την απουσία. Ο αλγόριθμος δεν επηρεάζει τις τιμές αυτές οπότε κάθε μεταβλητή που είναι 1 θα παραμείνει 1 και στα αποτελέσματα αυτά μπορούμε να το παρατηρήσουμε στα παρακάτω παραδείγματα.

```

{
  "Devices": [
    {
      "devid" : 1,
      "period" : 1,
      "devkw" : 500,
      "devtype" : "Fringe",
      "timesrun" : 23,
      "stbkw" : 40,
      "onhours": [1,1,1,1],
      "stbhours": [0,0,0,0],
      "onhouse": [0,1,1,0]
    },
    {
      "devid" : 2,
      "period" : 2,
      "devkw" : 250,
      "devtype" : "A/C",
      "timesrun" : 15,
      "stbkw" : 25,
      "onhours": [0,1,0,0],
      "stbhours": [1,0,1,1],
      "onhouse": [0,1,1,0]
    },
    {
      "devid" : 3,
      "period" : 4,
      "devkw" : 750,
      "devtype" : "A/C",
      "timesrun" : 30,
      "stbkw" : 50,
      "onhours": [1,1,1,0],
      "stbhours": [0,0,0,1],
      "onhouse": [0,1,1,0]
    },
    {
      "devid" : 4,
      "period" : 3,
      "devkw" : 750,
      "devtype" : "PC",
      "timesrun" : 50,
      "stbkw" : 30,
      "onhours": [1,1,0,0],
      "stbhours": [0,0,1,1],
      "onhouse": [0,1,1,0]
    }
  ]
}

```

Παράδειγμα εισόδου

3.4 Περιγραφή Λειτουργιάς

Δημιουργία προγράμματος προσομοίωσης για ηλεκτρικές συσκευές με χρήση JSON αρχείων και δημιουργία JSON αρχείων με χρονικές στιγμές για την λειτουργία των ηλεκτρικών συσκευών. Με μεταβλητές τις καιρικές συνθήκες, χρόνο λειτουργίας, κατανάλωση ρεύματος.

Όλος ο κώδικας βρίσκεται στο GitHub (<https://github.com/SotirisKalampokis/JavaApplication5>).

Αρχικά περνούμε όλα τα αντικείμενα το JSON και τα τοποθετούμε σε μια λίστα από αντικείμενα. Διαιρώντας `denkw` και `stbkw` με το εξήντα έχουμε την κατανάλωση ανά λεπτό κάθε συσκευής. Χρησιμοποιώντας την μεταβλητή `timesrun` και μιας ομοιόμορφης κατανομής αθέρατου βρίσκουμε X μοναδικούς αριθμούς για πόσες φορές θα τρέξει το πρόγραμμα. Ξέροντας την εποχή που θέλουμε την συσκευή να τρέξει και με την χρήση μιας κανονικής κατανομής με ± 2 και βάση το μέσο όρο των τριών μηνών της εποχής βρίσκουμε την θερμοκρασία της συσκευής.

Για κάθε `onhour` και `stbhour` τρέχουμε ένα ρολόι για εικοσιτέσσερις ώρες για είκοσι λεπτά ανά φορά. Κάθε φορά κοιτάμε αν ο αριθμός της επανάληψης είναι ίδιος με τον αριθμό του `timesrun` αν είναι ο ίδιος τότε τρέχουμε την συσκευή σε λειτουργία ή αναμονή. Έχοντας το αν θα τρέξει η συσκευή χρησιμοποιούμε μια κανονική κατανομή με ± 1.5 και βάση το 20 και πολλαπλασιάζουμε με τον `denkw` ανά λεπτό ή `stbkw` ανά λεπτό και περνούμε την κατανάλωση του εικοσαλέπτου.

Μετά κοιτάμε αν ο χρήστης βρίσκεται στο σπίτι και το σημειώνουμε με 1 και 0 διπλά από την κατανάλωση με μορφή πίνακα όπου το πρώτο στοιχείο είναι η κατανάλωση και το δεύτερο η κατοίκηση του σπιτιού.

Στα παρακάτω παραδείγματα εξόδου βλέπουμε τις δυο καταγραφές κατανάλωσης για `OnHours` και `STBHours`. Παρατηρούμε ότι η μέση θερμοκρασία και για τις δυο περιπτώσεις είναι η ίδια. Βλέπουμε όλες τις χρονικές στιγμές από 00:00 έως 24:00 και στην πρώτη στήλη του πίνακα είναι η κατανάλωση και στην δεύτερη στήλη η απασχόληση του δωματίου.

Στα παρακάτω παραδείγματα κώδικα έχουμε τους τρόπους που χρησιμοποιήθηκαν για να δημιουργήσουμε τα αποτελέσματα του προγράμματος. Στο παράδειγμα ένα είναι η κανονική κατανομή όπου `Base` είναι η βάση της κατανομής και το 1,5 η διακύμανση. Στο παράδειγμα δυο είναι ο τρόπος που χρησιμοποιήσαμε για να καταγράψουμε το ρολόι του αλγόριθμου.

```

{
  "devid" : 4,
  "devtype" : "PC",
  "Temp" : "25.604984226211975",
  "OnHours":{
    "00:00": [0.0,0],
    "00:20": [241.4445430864316,0],
    "00:40": [232.64678704436056,0],
    "01:00": [192.0730098027797,0],
    "01:20": [0.0,0],
    "01:40": [245.0120716269392,0],
    "02:00": [267.9275780854373,0],
    "02:20": [0.0,0],
    "02:40": [249.74043385266845,0],
    "03:00": [200.58459498939362,0],
    "03:20": [0.0,0],
    "03:40": [267.55494636576486,0],
    "04:00": [279.56417475523506,0],
    "04:20": [0.0,0],
    "04:40": [232.77098962123293,0],
    "05:00": [260.20543970520976,0],
    "05:20": [275.2999141048328,0],
    "05:40": [231.47246907789486,0],
    "06:00": [253.27164280862397,0],
    "06:20": [255.54639572419939,1],
    "06:40": [259.55017133218854,1],
    "07:00": [0.0,1],
    "07:20": [270.3893729256946,1],
    "07:40": [0.0,1],
    "08:00": [245.52742468904896,1],
    "08:20": [241.01467444559916,1],
    "08:40": [0.0,1],
    "09:00": [0.0,1],
    "09:20": [246.0569352602656,1],
    "09:40": [260.37407299822314,1],
    "10:00": [229.20263381460035,1],
    "10:20": [260.15367787307343,1],
    "10:40": [214.9191215118372,1],
    "11:00": [260.8928720398317,1],
    "11:20": [249.96806954237823,1],
    "11:40": [216.01094825501184,1],
    "12:00": [220.8290629650168,1],
    "12:20": [0.0,1],
    "12:40": [0.0,1],
    "13:00": [0.0,1],
    "13:20": [0.0,1],
    "13:40": [0.0,1],
    "14:00": [0.0,1],
    "14:20": [0.0,1],
    "14:40": [0.0,1],
    "15:00": [0.0,1],
    "15:20": [0.0,1],
    "15:40": [0.0,1],
    "16:00": [0.0,1],
    "16:20": [0.0,1],
    "16:40": [0.0,1],
    "17:00": [0.0,1],
    "17:20": [0.0,1],
    "17:40": [0.0,1],
    "18:00": [0.0,1],
    "18:20": [0.0,0],
    "18:40": [0.0,0],
    "19:00": [0.0,0],
    "19:20": [0.0,0],
    "19:40": [0.0,0],
    "20:00": [0.0,0],
    "20:20": [0.0,0],
    "20:40": [0.0,0],
    "21:00": [0.0,0],
    "21:20": [0.0,0],
    "21:40": [0.0,0],
    "22:00": [0.0,0],
    "22:20": [0.0,0],
    "22:40": [0.0,0],
    "23:00": [0.0,0],
    "23:20": [0.0,0],
    "23:40": [0.0,0],
    "24:00": [0.0,0]
  }
}

```

Παράδειγμα έξοδου OnHours Normal

```

{
  "devid" : 4,
  "devtype" : "PC",
  "Temp" : "25.604984226211975",
  "StbHours":{
    "00:00": [0.0,0],
    "00:20": [0.0,0],
    "00:40": [0.0,0],
    "01:00": [0.0,0],
    "01:20": [0.0,0],
    "01:40": [0.0,0],
    "02:00": [0.0,0],
    "02:20": [0.0,0],
    "02:40": [0.0,0],
    "03:00": [0.0,0],
    "03:20": [0.0,0],
    "03:40": [0.0,0],
    "04:00": [0.0,0],
    "04:20": [0.0,0],
    "04:40": [0.0,0],
    "05:00": [0.0,0],
    "05:20": [0.0,0],
    "05:40": [0.0,0],
    "06:00": [0.0,0],
    "06:20": [0.0,1],
    "06:40": [0.0,1],
    "07:00": [0.0,1],
    "07:20": [0.0,1],
    "07:40": [0.0,1],
    "08:00": [0.0,1],
    "08:20": [0.0,1],
    "08:40": [0.0,1],
    "09:00": [0.0,1],
    "09:20": [0.0,1],
    "09:40": [0.0,1],
    "10:00": [0.0,1],
    "10:20": [0.0,1],
    "10:40": [0.0,1],
    "11:00": [0.0,1],
    "11:20": [0.0,1],
    "11:40": [0.0,1],
    "12:00": [0.0,1],
    "12:20": [9.533158101086002,1],
    "12:40": [10.794478055310105,1],
    "13:00": [10.09043468655567,1],
    "13:20": [8.322796073036546,1],
    "13:40": [10.038173875064617,1],
    "14:00": [10.633952880769515,1],
    "14:20": [0.0,1],
    "14:40": [9.748732316422439,1],
    "15:00": [0.0,1],
    "15:20": [0.0,1],
    "15:40": [0.0,1],
    "16:00": [9.998136009457998,1],
    "16:20": [0.0,1],
    "16:40": [0.0,1],
    "17:00": [8.777428715956619,1],
    "17:20": [8.951194709024858,1],
    "17:40": [0.0,1],
    "18:00": [10.085079280645694,1],
    "18:20": [0.0,0],
    "18:40": [0.0,0],
    "19:00": [10.573216522904543,0],
    "19:20": [9.543372774678735,0],
    "19:40": [11.588285119697234,0],
    "20:00": [0.0,0],
    "20:20": [0.0,0],
    "20:40": [9.631024433259103,0],
    "21:00": [0.0,0],
    "21:20": [9.47025576373035,0],
    "21:40": [9.883956627992394,0],
    "22:00": [9.571849133287756,0],
    "22:20": [10.914369822348066,0],
    "22:40": [0.0,0],
    "23:00": [10.63901642338932,0],
    "23:20": [8.973084401483634,0],
    "23:40": [9.689831813533187,0],
    "24:00": [0.0,0]
  }
}

```

Παράδειγμα έξοδου STBHours Normal

```
NormalDistribution ndOn = new NormalDistribution(NDbase,1.5);
```

Παράδειγμα Κώδικα 1

```
if(hours<10 && mins<20){
    s=s+("\t\t\t\t\t"+"0"+hours+":0"+mins+"\":["+nowKw+","]);
}else if (hours<10){
    s=s+("\t\t\t\t\t"+"0"+hours+": "+mins+"\":["+nowKw+","]);

}else if(hours==24){
    s=s+("\t\t\t\t\t"+hours+":0"+mins+"\":["+nowKw+","]);
}else if(hours>=10 && mins<20){
    s=s+("\t\t\t\t\t"+hours+":0"+mins+"\":["+nowKw+","]);

}else{
    s=s+("\t\t\t\t\t"+hours+": "+mins+"\":["+nowKw+","]);
}

}
```

Παράδειγμα Κώδικα 2

Κεφαλαίο 4 Περίπτωσης και Ολική Κατανάλωση ανά Περίπτωση

4.1 Περίπτωσης Κατανάλωση

Στην ενότητα αυτή επιχειρούμε να δημιουργήσουμε 2 συνθετικά σύνολα δεδομένων που αντιστοιχούν σε διαφορετικά προφίλ χρήσης για το ίδιο σύνολο οικιακών συσκευών. Στο πρώτο σύνολο ο χρήστης κάνει αλόγιστη χρήση ενέργειας, ενώ στο δεύτερο σύνολο ο χρήστης αποφεύγει κάθε άσκοπη χρήση συσκευών.

Θεωρούμε ότι και τα δύο προφίλ χειρίζονται Χ συσκευές (πλυντήριο, κουζίνα κλπ) σε ένα σπίτι με 3 δωμάτια.

Οι παράμετροι στις δύο περιπτώσεις είναι ως εξής:

- Περίπτωση Α: Υπερκατανάλωση ρεύματος όπου η κανονική κατανομή για την εύρεση των λεπτών αλλάζει την βάση από 20 σε 22 με την διακύμανση ± 1.5 .

- Περίπτωση Β: Εξοικονόμηση ρεύματος όπου η κανονική κατανομή αλλάζει την βάση από 20 σε 19 με την διακύμανση ± 1.5 .

Τα αρχεία που παράγονται στις 2 περιπτώσεις έχουν το ίδιο μέγεθος.

Υπολογίζοντας την κατανάλωση για τα δύο νοικοκυριά βρίσκουμε στο Α υπερκατανάλωση και στο Β εξοικονόμηση.

Το παραπάνω πείραμα μπορεί να παραμετροποιηθεί κατάλληλα με παραμέτρους χρήσης που μπορεί να εξαχθούν από τα πραγματικά δεδομένα χρήσης ενός σπιτιού. Με τη χρήση της γεννήτριας μπορούμε να υπολογίσουμε για την περίπτωση Β την αντίστοιχη κατανάλωση της περίπτωσης Α δίνοντας στο χρήστη ρεαλιστικά στοιχεία σε σχέση με την εξοικονόμηση που θα μπορούσε να έχει.

Η κατάσταση Standby παραμένει η ίδια σε όλες της περίπτωσης. Παρατηρούμε από τα παραδείγματα ότι η κατανάλωση στην περίπτωση Α σε σχέση με την περίπτωση Β είναι περίπου +50 μονάδες πιο υψηλά αυτό είναι λόγω της κατανάλωσης της συσκευής και της υπερκατανάλωσης. Παρατηρούμε επίσης ότι η μέση θερμοκρασία έχει αλλάξει σε κάθε περίπτωση καθώς μιλάμε για διαφορετικά σπίτια.

```

{
  "devid" : 4,
  "devtype" : "PC",
  "Temp" : "26.238900478566116",
  "OnHours":{
    "00:00": [289.98477469319965,0],
    "00:20": [0.0,0],
    "00:40": [316.4045420498027,0],
    "01:00": [294.67320308975695,0],
    "01:20": [0.0,0],
    "01:40": [0.0,0],
    "02:00": [0.0,0],
    "02:20": [0.0,0],
    "02:40": [292.34361543235997,0],
    "03:00": [271.69561265540557,0],
    "03:20": [284.5738491038081,0],
    "03:40": [270.1806689832027,0],
    "04:00": [293.9999571371885,0],
    "04:20": [0.0,0],
    "04:40": [0.0,0],
    "05:00": [272.3457435068393,0],
    "05:20": [255.83609191058426,0],
    "05:40": [252.59438472480608,0],
    "06:00": [0.0,0],
    "06:20": [272.8136590546212,1],
    "06:40": [0.0,1],
    "07:00": [267.25880363710496,1],
    "07:20": [295.790947959791,1],
    "07:40": [0.0,1],
    "08:00": [0.0,1],
    "08:20": [274.7934102153095,1],
    "08:40": [259.28452099460844,1],
    "09:00": [253.97285238328408,1],
    "09:20": [266.9504757472229,1],
    "09:40": [254.21793489050964,1],
    "10:00": [300.72703337501736,1],
    "10:20": [0.0,1],
    "10:40": [0.0,1],
    "11:00": [272.94123167659,1],
    "11:20": [244.3958616307332,1],
    "11:40": [251.43845930945824,1],
    "12:00": [288.11872681539194,1],
    "12:20": [0.0,1],
    "12:40": [0.0,1],
    "13:00": [0.0,1],
    "13:20": [0.0,1],
    "13:40": [0.0,1],
    "14:00": [0.0,1],
    "14:20": [0.0,1],
    "14:40": [0.0,1],
    "15:00": [0.0,1],
    "15:20": [0.0,1],
    "15:40": [0.0,1],
    "16:00": [0.0,1],
    "16:20": [0.0,1],
    "16:40": [0.0,1],
    "17:00": [0.0,1],
    "17:20": [0.0,1],
    "17:40": [0.0,1],
    "18:00": [0.0,1],
    "18:20": [0.0,0],
    "18:40": [0.0,0],
    "19:00": [0.0,0],
    "19:20": [0.0,0],
    "19:40": [0.0,0],
    "20:00": [0.0,0],
    "20:20": [0.0,0],
    "20:40": [0.0,0],
    "21:00": [0.0,0],
    "21:20": [0.0,0],
    "21:40": [0.0,0],
    "22:00": [0.0,0],
    "22:20": [0.0,0],
    "22:40": [0.0,0],
    "23:00": [0.0,0],
    "23:20": [0.0,0],
    "23:40": [0.0,0],
    "24:00": [0.0,0]
  }
}

```

Παράδειγμα έξοδου OnHours Περίπτωσης Α

```

{
  "devId" : 4,
  "devType" : "PC",
  "Temp" : "28.656550000958973",
  "OnHours":{
    "00:00": [217.0925627655273,0],
    "00:20": [249.88317753650264,0],
    "00:40": [0.0,0],
    "01:00": [204.21885161176027,0],
    "01:20": [0.0,0],
    "01:40": [0.0,0],
    "02:00": [266.4990714126906,0],
    "02:20": [0.0,0],
    "02:40": [236.79029230276583,0],
    "03:00": [0.0,0],
    "03:20": [0.0,0],
    "03:40": [271.26469983438255,0],
    "04:00": [260.8360637158951,0],
    "04:20": [246.45950776468158,0],
    "04:40": [202.84010026460334,0],
    "05:00": [0.0,0],
    "05:20": [201.10974021713264,0],
    "05:40": [234.94699927204294,0],
    "06:00": [202.78822231405016,0],
    "06:20": [250.35606185247232,1],
    "06:40": [233.93652815668457,1],
    "07:00": [281.3450986542675,1],
    "07:20": [201.3962772623266,1],
    "07:40": [238.6195572292845,1],
    "08:00": [228.22195407362412,1],
    "08:20": [0.0,1],
    "08:40": [215.07134848018352,1],
    "09:00": [216.05829383623677,1],
    "09:20": [251.9680837203028,1],
    "09:40": [0.0,1],
    "10:00": [217.831045693695,1],
    "10:20": [0.0,1],
    "10:40": [0.0,1],
    "11:00": [227.20242890778962,1],
    "11:20": [260.44398618747124,1],
    "11:40": [260.6702761034632,1],
    "12:00": [0.0,1],
    "12:20": [0.0,1],
    "12:40": [0.0,1],
    "13:00": [0.0,1],
    "13:20": [0.0,1],
    "13:40": [0.0,1],
    "14:00": [0.0,1],
    "14:20": [0.0,1],
    "14:40": [0.0,1],
    "15:00": [0.0,1],
    "15:20": [0.0,1],
    "15:40": [0.0,1],
    "16:00": [0.0,1],
    "16:20": [0.0,1],
    "16:40": [0.0,1],
    "17:00": [0.0,1],
    "17:20": [0.0,1],
    "17:40": [0.0,1],
    "18:00": [0.0,1],
    "18:20": [0.0,0],
    "18:40": [0.0,0],
    "19:00": [0.0,0],
    "19:20": [0.0,0],
    "19:40": [0.0,0],
    "20:00": [0.0,0],
    "20:20": [0.0,0],
    "20:40": [0.0,0],
    "21:00": [0.0,0],
    "21:20": [0.0,0],
    "21:40": [0.0,0],
    "22:00": [0.0,0],
    "22:20": [0.0,0],
    "22:40": [0.0,0],
    "23:00": [0.0,0],
    "23:20": [0.0,0],
    "23:40": [0.0,0],
    "24:00": [0.0,0]
  }
}

```

Παράδειγμα έξοδου OnHours Περίπτωσης Β

4.2 Ολική Κατανάλωση

Συγκεντρώνοντας τις τρεις περιπτώσεις και την συνολική κατανάλωση ανά μέρα μπορούμε να καταλάβουμε καλύτερα την περίπτωση κατανάλωσης. Παρατηρούμε ότι η κατανάλωση σε υπερκαταναλωτικές περιπτώσεις είναι χαμηλότερη από κατάσταση χαμηλής κατανάλωσης αυτό συμβαίνει επειδή οι στιγμές κατανάλωσης είναι διαφορετικές. Μπορούν να πέσουν σε στιγμές αδράνειας και αυτό μας δίνει χαμηλές στιγμές κατανάλωσης ενώ η χαμηλές παραπατήσεις έχουν κανονική κατανάλωση οπότε ιδανικά για να παρατηρήσουμε ότι η κατανάλωση είναι “σωστά” κατανεμημένη στην περίπτωση. Το timesrun ή Onhouse πρέπει να είναι το μέγιστο για να έχουμε “σωστές” μέτρησις (στο παράδειγμα μπορούμε να το δούμε αυτό με την συσκευή fridge) .

```

{
  "Devices": [
    {
      "devid" : 1,
      "devtype" : "Fringe",
      "Under" : "3732.2011020792124",
      "Over" : "4221.812643289319",
      "Normal" : "3895.1702012193236"
    },
    {
      "devid" : 2,
      "devtype" : "A/C",
      "Under" : "202.22046643703754",
      "Over" : "382.51865095390025",
      "Normal" : "402.367595274484"
    },
    {
      "devid" : 3,
      "devtype" : "A/C",
      "Under" : "9465.605499122212",
      "Over" : "9742.871887475902",
      "Normal" : "9946.594542818277"
    },
    {
      "devid" : 4,
      "devtype" : "PC",
      "Under" : "4783.575460978255",
      "Over" : "5205.113465686237",
      "Normal" : "5936.044391913321"
    }
  ]
}

```

Παράδειγμα συνολικής κατανάλωσης

Κεφαλαίο 5 Συμπεράσματα

Σε αυτήν την εργασία έγινε προσπάθεια να δημιουργηθεί ένας αλγόριθμος που θα παίρνει γνωστά αρχεία για ηλεκτρικές συσκευές και μεταβλητές γύρω από αυτές της συσκευές. Με χρήση json αρχείων για την είσοδο και την εξόδο του προγράμματος ώστε να μπορούν να χρησιμοποιηθούν σε άλλα παρόμοια προγράμματα. Χρησιμοποιώντας κατανομές για την παράγωγή της κατανάλωσης ώστε τα δεδομένα να πλησιάζουν τα πραγματικά καθώς πραγματικά δεδομένα είναι δύσκολα να καταγραφούν. Με περίπτωσις έγινε η προσπάθεια να δώσουμε περισσότερα δεδομένα κατανάλωσης για διαφορετικές περίπτωσις σπιτιών και μεγαλύτερο εύρος δεδομένων.

Κεφαλαίο 6 Αναφορές

Amouroux, E., Huraux, T., Sempé, F., Sabouret, N., & Haradji, Y. (2013, February). Simulating human activities to investigate household energy consumption.

[1]<https://hal.archives-ouvertes.fr/hal-01852256/document>

Johnson, B. J., Starke, M. R., Abdelaziz, O. A., Jackson, R. K., & Tolbert, L. M. (2014, February). A method for modeling household occupant behavior to simulate residential energy consumption. In *ISGT 2014* (pp. 1-5). IEEE.

[2]<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8322286>

Ramadan, M., Alsalemi, A., Bensaali, F., Amira, A., Sardianos, C., Varlamis, I., ... & Anagnostopoulos, D. (2019). 'Simulating appliance-based power consumption records for energy efficiency awareness. *Västerås, Sweden*.

[3]http://www.energy-proceedings.org/wp-content/uploads/2020/03/1021_Paper_0705045412.pdf