



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ &  
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ



# ΜΕΙΩΣΗ ΥΠΕΡΤΑΣΕΩΝ ΣΕ ΗΛΕΚΤΡΙΚΑ ΔΙΚΤΥΑ ΜΕΣΩ ΔΙΕΠΑΦΗΣ ΜΕ ΔΙΚΤΥΑ ΤΗΛΕΘΕΡΜΑΝΣΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

---

της

**ΣΤΥΛΙΑΝΗ ΖΑΧΟΥ**

**Επιβλέπων:** Γεώργιος Χριστοφορίδης

ΚΟΖΑΝΗ/ΟΚΤΩΒΡΙΟΣ/2024





HELLENIC DEMOCRACY  
UNIVERSITY OF WESTERN MACEDONIA

FUCULTY OF ENGINEERING  
DEPARTMENT OF ELECTRICAL &  
COMPUTER ENGINEERING



# MITIGATION OF OVERVOLTAGES IN ELECTRICAL DISTRIBUTION NETWORKS THROUGH INTERFACE WITH DISTRICT HEATING NETWORKS

THESIS

---

**STYLIANI ZACHOU**

**SUPERVISOR:** Georgios Christoforidis

KOZANI/OKTOBER/2024





ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ  
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

## ΔΗΛΩΣΗ ΜΗ ΛΟΓΟΚΛΟΠΗΣ ΚΑΙ ΑΝΑΛΗΨΗΣ ΠΡΟΣΩΠΙΚΗΣ ΕΥΘΥΝΗΣ

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο “ ΜΕΙΩΣΗ ΥΠΕΡΤΑΣΕΩΝ ΣΕ ΗΛΕΚΤΡΙΚΑ ΔΙΚΤΥΑ ΜΕΣΩ ΔΙΕΠΑΦΗΣ ΜΕ ΔΙΚΤΥΑ ΤΗΛΕΘΕΡΜΑΝΣΗΣ ” καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Χριστοφορίδη Γεώργιο αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Copyright (C) Ονοματεπώνυμο Φοιτητή & Επιβλέποντα, Έτος, Πόλη

Copyright (C) Στυλιανή Ζάχου, Γεώργιος Χριστοφορίδης, 2024, Κοζάνη

Υπογραφή Φοιτητή: \_\_\_\_\_



## Περίληψη

---

Η παρούσα διπλωματική διερευνά τις δυνατότητες παροχής επικουρικών υπηρεσιών από δίκτυα τηλεθέρμανσης σε ηλεκτρικά δίκτυα μέσω ολοκληρωμένης ανάλυσης και μοντελοποίησης και των δύο συστημάτων. Με την αυξανόμενη διείσδυση των ανανεώσιμων πηγών ενέργειας στα ηλεκτρικά δίκτυα, η διατήρηση της σταθερότητας του δικτύου καθίσταται σημαντική πρόκληση λόγω της μεταβλητής φύσης τους. Για την αντιμετώπιση αυτού του προβλήματος, τα πολυενεργειακά συστήματα, που συνδυάζουν διαφορετικές ενεργειακές υποδομές, όπως ηλεκτρικά και θερμικά δίκτυα, προσφέρουν μια λύση. Η παρούσα έρευνα μοντελοποιεί και προσομοιώνει την ενοποίηση ενός ηλεκτρικού δικτύου με ένα σύστημα τηλεθέρμανσης χρησιμοποιώντας την γλώσσα προγραμματισμού Python, και συγκεκριμένα τις βιβλιοθήκες `randarower` και `randaripes`. Το σύστημα ενσωματώνει μια αντλία θερμότητας που μετατρέπει την πλεονάζουσα ηλεκτρική ενέργεια σε θερμική ενέργεια, η οποία αποθηκεύεται σε μια θερμική δεξαμενή για μεταγενέστερη χρήση. Η μελέτη επικεντρώνεται στην αλληλεπίδραση μεταξύ του ελέγχου της τάσης στην ηλεκτρική πλευρά και του θερμικού ελέγχου στο δίκτυο θέρμανσης, αναδεικνύοντας τον ρόλο των ελεγκτών στη βελτιστοποίηση της χρήσης ενέργειας. Τα αποτελέσματα προσομοίωσης καταδεικνύουν πώς το δίκτυο τηλεθέρμανσης μπορεί να υποστηρίξει το ηλεκτρικό δίκτυο, να συνεισφέρει στην περαιτέρω αξιοποίηση της ενέργειας που παράγεται από ανανεώσιμες πηγές ενέργειας και να συμβάλει στην διατήρηση των λειτουργικών ορίων του δικτύου. Αυτή η ολοκληρωμένη προσέγγιση στη διαχείριση του δικτύου συμβάλλει στην ενεργειακή απόδοση και τη σταθερότητα στα σύγχρονα έξυπνα δίκτυα.

**Λέξεις Κλειδιά:** Πολυενεργειακά συστήματα, θερμικά δίκτυα, ηλεκτρικά δίκτυα, ανανεώσιμες πηγές ενέργειας, αντλία θερμότητας, έλεγχος τάσης, αποθήκευση ενέργειας.





## Abstract

---

*This thesis investigates the potential of providing ancillary services from district heating networks to electrical networks through integrated analysis and modeling of both systems. With the increased penetration of renewable energy sources (RES) into electrical grids, maintaining grid stability becomes a significant challenge due to the variable nature of RES. To address this, multi-energy systems (MES), combining different energy infrastructures such as electric and thermal networks, offer a solution. This research models and simulates the integration of an electrical grid with a district heating system using Python programming language, specifically the pandapower and pandapipes libraries. The system incorporates a heat pump that converts surplus electrical energy into thermal energy, stored in a thermal tank for later use. The study focuses on the interaction between voltage control on the electrical side and thermal control in the heating network, highlighting the role of controllers in optimizing energy use. Simulation results demonstrate how the district heating network can support the electrical grid, contribute to further utilization of energy produced from renewable energy sources and help maintain the operational limits of the network. This integrated approach to network management contributes to energy efficiency and stability in modern smart grids.*

**Keywords:** Multi-energy systems, district heating networks, electrical networks, renewable energy sources, heat pump, voltage control, energy storage.



## ***Ευχαριστίες***

---

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες μου σε όλους όσοι στάθηκαν δίπλα μου κατά τη διάρκεια της εκπόνησης αυτής της διπλωματικής εργασίας. Ευχαριστώ τον επιβλέποντα καθηγητή μου, κύριο Γεώργιο Χριστοφορίδη για την καθοδήγηση, την εμπειρία και την εμπιστοσύνη που μου έδειξε, σε όλα τα στάδια της διπλωματικής εργασίας. Επιπλέον, θα ήθελα να θα ήθελα να ευχαριστήσω τον υποψήφιο διδάκτορα, Νίκο Κελεπούρη, για την πολύτιμη βοήθειά του στην ολοκλήρωση αυτής της εργασίας. Τέλος, ευχαριστώ την οικογένειά μου, για την υποστήριξη και την κατανόηση τους, χωρίς τις οποίες δεν θα ήταν δυνατή η πραγματοποίηση και η ολοκλήρωση των ακαδημαϊκών μου σπουδών.



# Περιεχόμενα

---

ΠΕΡΙΛΗΨΗ	- 1 -
ABSTRACT	3
ΕΥΧΑΡΙΣΤΙΕΣ	5
ΠΕΡΙΕΧΟΜΕΝΑ	7
ΚΑΤΑΛΟΓΟΣ ΓΡΑΦΗΜΑΤΩΝ	10
ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ	11
ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ	13
ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ	15
1.1 Ενσωμάτωση Ανανεώσιμων Πηγών Ενέργειας (ΑΠΕ) στα Ηλεκτρικά Δίκτυα	15
1.2 Προκλήσεις για την Ενσωμάτωση των ΑΠΕ στα Ηλεκτρικά Δίκτυα	17
1.2.1 Απρόβλεπτη φύση των ΑΠΕ	17
1.2.2 Δίκτυα διανομής	17
1.2.3 Δίκτυα διανομής χαμηλής τάσης	17
1.2.4 Μικροδίκτυα	18
1.3 Πολυενεργειακό σύστημα	18
1.4 Αντλία θερμότητας	19
1.5 Αντικείμενο διπλωματικής	20
ΚΕΦΑΛΑΙΟ 2: ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ	20
ΚΕΦΑΛΑΙΟ 3: ΜΕΘΟΔΟΛΟΓΙΑ	24
3.1 Σύνδεση ηλεκτρικού με το δίκτυο τηλεθέρμανσης	24
3.1.1 Ελεγκτής τάσης	25
3.1.2 Ελεγκτής θερμότητας	25

<b>ΚΕΦΑΛΑΙΟ 4: ΑΝΑΛΥΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΕΡΓΑΛΕΙΟΥ</b>	<b>28</b>
4.1 Εγκατάσταση	28
4.1.1 Κατέβασμα αρχείου	28
4.1.2 Anaconda Navigator	29
4.1.3 Δημιουργία καινούργιου περιβάλλοντος και εγκατάσταση της Spyder	35
4.1.4 Εγκατάσταση συγκεκριμένων εκδόσεων βιβλιοθηκών	36
4.2 Εκτέλεση κώδικα	39
4.3 Τροποποίηση αρχείου κώδικα	41
4.3.1 Ηλεκτρικό δίκτυο	41
4.3.2 Θερμικό Δίκτυο	51
4.3.3 Ελεγκτής θερμότητας	59
<b>ΚΕΦΑΛΑΙΟ 5: ΜΕΛΕΤΗ ΠΕΡΙΠΤΩΣΗΣ</b>	<b>63</b>
5.1 Ηλεκτρικό δίκτυο	63
Πίνακας 1: Ισχύς φορτίων	64
Πίνακας 2: Ισχύς Φωτοβολταϊκών	64
Πίνακας 3: Μήκη γραμμών	65
5.2 Θερμικό δίκτυο	66
Πίνακας 4: Βαλβίδες	66
Πίνακας 5: Καταναλωτές	67
Πίνακας 6: Μήκος σωληνώσεων παροχής	67
Πίνακας 7: Μήκος σωληνώσεων επιστροφής	68
5.3 Εγκατάσταση μετατροπής της ηλεκτρικής ενέργειας σε θερμική	69
<b>ΚΕΦΑΛΑΙΟ 6: ΑΠΟΤΕΛΕΣΜΑΤΑ</b>	<b>70</b>
6.1 Αποτελέσματα μελέτης	70
<b>ΚΕΦΑΛΑΙΟ 7: ΣΥΜΠΕΡΑΣΜΑΤΑ</b>	<b>86</b>
7.1 Συμπεράσματα της Μελέτης	86
7.1.1 Αποτελεσματικότητα της Αντλίας Θερμότητας	86
7.1.2 Έλεγχος Τάσης και Θερμικός Έλεγχος	87
7.1.3 Συμβολή στη Σταθερότητα του Δικτύου	87
7.1.4 Βελτιστοποίηση Χρήσης Ανανεώσιμων Πηγών Ενέργειας	87
7.2 Μελλοντική Εργασία	87

ΠΑΡΑΡΤΗΜΑ – ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ	89
ΒΙΒΛΙΟΓΡΑΦΙΑ	90
ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ - ΑΡΚΤΙΚΟΛΕΞΑ - ΑΚΡΩΝΥΜΙΑ	92

## Κατάλογος Γραφημάτων

---

Γράφημα 1: Η τάση του ζυγού 18 σε p.u. (σενάριο 1) .....	71
Γράφημα 2: Η τάση του ζυγού 22 σε p.u. (σενάριο 2) .....	71
Γράφημα 3: Η τάση του ζυγού 10 σε p.u. (σενάριο 3) .....	72
Γράφημα 4: Η τάση στον ζυγό 3 σε p.u. (σενάριο 1) .....	72
Γράφημα 5: Η τάση στον ζυγό 3 σε p.u. (σενάριο 2) .....	73
Γράφημα 6: Η τάση στον ζυγό 3 σε p.u. (σενάριο 3) .....	73
Γράφημα 7: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 1).....	74
Γράφημα 8: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 2).....	74
Γράφημα 9: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 3).....	74
Γράφημα 10: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 1).....	75
Γράφημα 11: Η φόρτωση την γραμμής των ζυγών 22-23 (σενάριο 2).....	75
Γράφημα 12: Η φόρτωση της γραμμής των ζυγών 10-11 (σενάριο 3).....	76
Γράφημα 13: Η φόρτωση της γραμμής των ζυγών 14-18 (σενάριο 1).....	77
Γράφημα 14: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 1).....	77
Γράφημα 15: Η φόρτωση της γραμμής των ζυγών 21-22 (σενάριο 2).....	77
Γράφημα 16: Η φόρτωση της γραμμής των ζυγών 22-23 (σενάριο 2).....	78
Γράφημα 17: Η φόρτωση της γραμμής των ζυγών 9-10 (σενάριο 3).....	78
Γράφημα 18: Η φόρτωση της γραμμής των ζυγών 10-11 (σενάριο 3).....	79
Γράφημα 19: Μέση τιμή θερμοκρασίας της δεξαμενής (σενάριο 1,2,3).....	79
Γράφημα 20: Κατανάλωση ισχύος της αντλίας θερμότητας (σενάριο 1,2,3) .....	80
Γράφημα 21: Η τάση του ζυγού 18 σε p.u. (σενάριο 4) .....	81
Γράφημα 22: Η τάση του ζυγού 18 σε p.u. (σενάριο 5) .....	81
Γράφημα 23: Η τάση του ζυγού 3 σε p.u. (σενάριο 4) .....	82
Γράφημα 24: Η τάση του ζυγού 3 σε p.u. (σενάριο 5) .....	82
Γράφημα 25: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 4).....	83
Γράφημα 26: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 5).....	83
Γράφημα 27: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 4,5).....	84
Γράφημα 28: Η φόρτωση της γραμμής των ζυγών 14-18 (σενάριο 4,5).....	84
Γράφημα 29: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 4,5).....	85



## Κατάλογος Εικόνων

---

Εικόνα 1: Μέση ετήσια προσθήκη ισχύος ανανεώσιμων πηγών ενέργειας και εγκατεστημένη ισχύς, ιστορικά και πρόβλεψη [1].....	16
Εικόνα 2: Συνολική παραγωγή ηλεκτρικής ενέργειας από ΑΠΕ, φωτοβολταϊκά και υδροηλεκτρικά κατά την περίοδο 2000-2026.[4] .....	16
Εικόνα 3: Η αρχιτεκτονική ενός πολύ-ενεργειακού συστήματος[10] .....	19
Εικόνα 4: Αντλία θερμότητας[17] .....	25
Εικόνα 5: Ρύθμιση ελέγχου PID για τη ρύθμιση της κατανάλωσης ισχύος της αντλίας θερμότητας[17].....	26
Εικόνα 6: Τρόποι λειτουργίας του ελεγκτή θερμότητας[17] .....	27
Εικόνα 7: Διαδρομή εύρεσης του zip αρχείου στην ιστοσελίδα του GitHub .....	29
Εικόνα 8: Λήψη του εργαλείου Anaconda Navigator .....	30
Εικόνα 9: Επιλογές αρχείου διαφόρων λογισμικών .....	30
Εικόνα 10: Εγκατάσταση βήμα 1 .....	31
Εικόνα 11: Εγκατάσταση βήμα 2 .....	31
Εικόνα 12: Εγκατάσταση βήμα 3 .....	32
Εικόνα 13: Εγκατάσταση βήμα 4 .....	32
Εικόνα 14: Εγκατάσταση βήμα 5 .....	33
Εικόνα 15: Εγκατάσταση βήμα 6 .....	33
Εικόνα 16: Εγκατάσταση βήμα 7 .....	34
Εικόνα 17: Εγκατάσταση βήμα 8 .....	34
Εικόνα 18: Αρχική οθόνη της εφαρμογής.....	35
Εικόνα 19: Δημιουργία καινούργιου περιβάλλοντος.....	35
Εικόνα 20: Εγκατάσταση της εφαρμογής Spyder .....	36
Εικόνα 21: Άνοιγμα του τερματικού.....	37
Εικόνα 22: Εγκατάσταση συγκεκριμένων βιβλιοθηκών βήμα 1.....	37
Εικόνα 23: Εγκατάσταση συγκεκριμένων βιβλιοθηκών βήμα 2.....	38
Εικόνα 24: Άνοιγμα της Εφαρμογής Spyder.....	38
Εικόνα 25: Αρχή εκτέλεσης κώδικα με τον ελεγκτή τάσης ενεργοποιημένο .....	39
Εικόνα 26: Τέλος εκτέλεσης κώδικα με τον ελεγκτή τάσης ενεργοποιημένο .....	39
Εικόνα 27: Αρχή εκτέλεσης κώδικα με τον ελεγκτή τάσης απενεργοποιημένο .....	40
Εικόνα 28: Τέλος εκτέλεσης κώδικα με τον ελεγκτή τάσης απενεργοποιημένο .....	40
Εικόνα 29: Φάκελος cosim_pandaripes_pandapower.....	41
Εικόνα 30: Διαδρομή υποφακέλου power.....	41
Εικόνα 31: Δεδομένα για τους Ζυγούς .....	42
Εικόνα 32: Δεδομένα για φορτία και την αντλία θερμότητας.....	43
Εικόνα 33: Δεδομένα φωτοβολταϊκών.....	44
Εικόνα 34: Δεδομένα για την εξωτερική πηγή ενέργειας .....	45
Εικόνα 35: Δεδομένα για τις γραμμές.....	45

Εικόνα 36: CSV αρχείο για τα φορτία .....	47
Εικόνα 37: CSV αρχείο για τα την παραγωγή των φωτοβολταϊκών .....	48
Εικόνα 38: Διαδρομή υποφακέλου el_network.....	49
Εικόνα 39: Κομμάτι του κώδικα προσομοίωσης - αρχικοποίηση και δήλωση των οντοτήτων φορτίων και φωτοβολταϊκών.....	49
Εικόνα 40: Σύνδεση οντοτήτων.....	50
Εικόνα 41: Ορισμός λεξικού PLOT_DICT στον κώδικα ανάλυσης των αποτελεσμάτων.....	50
Εικόνα 42: Συνάρτηση σύγκρισης αποτελεσμάτων.....	51
Εικόνα 43: Διαδρομή υποφακέλου dh_network.....	51
Εικόνα 44: Κλάση DHNetwork.....	52
Εικόνα 45: Δημιουργία κόμβων .....	53
Εικόνα 46: Δημιουργία σωλήνων παροχής.....	54
Εικόνα 47: Δημιουργία σωλήνων επιστροφής.....	54
Εικόνα 48: Δημιουργία καταναλωτών .....	54
Εικόνα 49: Δημιουργία της παράκαμψης bypass .....	55
Εικόνα 50: Δημιουργία βαλβίδων ελέγχου ροής.....	55
Εικόνα 51: Τροποποιήσεις μεταβλητών .....	56
Εικόνα 52: Συνάρτηση update.....	56
Εικόνα 53: Ενημέρωση μεταβλητών .....	57
Εικόνα 54: Κλάση DHNetworkSimulator.....	57
Εικόνα 55: Κομμάτι του κώδικα προσομοίωσης – αρχικοποίηση θερμικών καταναλωτών .....	58
Εικόνα 56: Συνάρτηση connectEntities .....	58
Εικόνα 57: Συνάρτηση connectDataCollector.....	59
Εικόνα 58: Διαδρομή υποφακέλου heat.....	59
Εικόνα 59: Διαδρομή υποφακέλου flex_heat_controller.....	60
Εικόνα 60: Κλάση SimpleFlexHeatController.....	60
Εικόνα 61: Συνάρτηση _do_state_based_control.....	61
Εικόνα 62: Ενημέρωση μεταβλητών εισόδου και εξόδου του ελεγκτή.....	61
Εικόνα 63: Κλάση SimpleFlexHeatControllerSimulator.....	61
Εικόνα 64: Κομμάτι κώδικα προσομοίωσης - Ενημέρωση καταναλωτών .....	62
Εικόνα 65: Ηλεκτρικό Δίκτυο [8].....	63
Εικόνα 66: Δίκτυο τηλεθέρμανσης.....	66

## **Κατάλογος Πινάκων**

---

Πίνακας 1: Ισχύς φορτίων .....	64
Πίνακας 2: Ισχύς Φωτοβολταϊκών.....	64
Πίνακας 3: Μήκη γραμμών .....	65
Πίνακας 4: Βαλβίδες .....	66
Πίνακας 5: Καταναλωτές .....	67
Πίνακας 6: Μήκος σωληνώσεων παροχής.....	67
Πίνακας 7: Μήκος σωληνώσεων επιστροφής.....	68



# Κεφάλαιο 1: Εισαγωγή

---

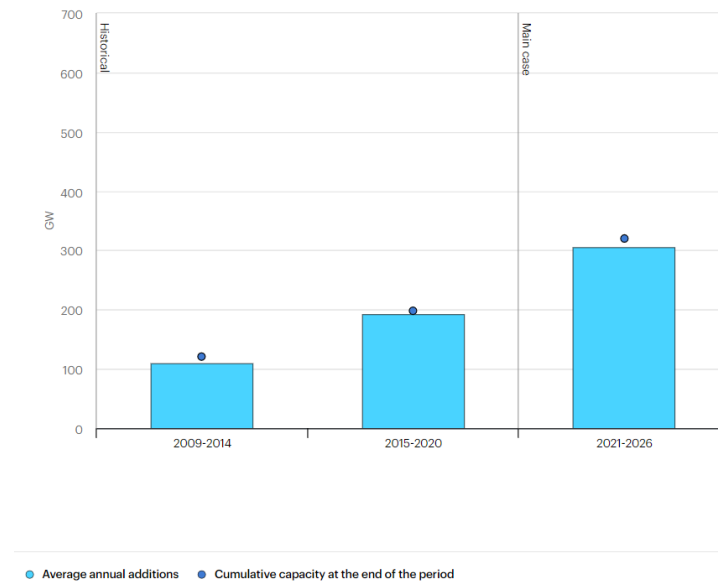
Η Ευρωπαϊκή Ένωση, στο πλαίσιο της Ευρωπαϊκής Πράσινης Συμφωνίας, έχει θέσει ως στόχο να καταστεί κλιματικά ουδέτερη έως το 2050. Για να επιτευχθεί αυτός ο στόχος, είναι απαραίτητη η ενεργειακή μετάβαση, που περιλαμβάνει τη μείωση των εκπομπών διοξειδίου του άνθρακα και την αύξηση της χρήσης ανανεώσιμων πηγών ενέργειας σε όλα τα επίπεδα της παραγωγής και κατανάλωσης ενέργειας. Όμως η αυξανόμενη διείσδυση των Ανανεώσιμων Πηγών Ενέργειας (ΑΠΕ) στα ηλεκτρικά δίκτυα παγκοσμίως, όπως τα φωτοβολταϊκά (Photovoltaic – PV), έχει αναδείξει σημαντικές προκλήσεις για τη σταθερότητα, την ασφάλεια και την αξιοπιστία των συστημάτων ηλεκτρικής ενέργειας. Η ασυνεχής και ασταθής φύση της παραγωγής ενέργειας από ΑΠΕ μπορεί να προκαλέσει προβλήματα όπως υπερτάσεις, συχνές διακυμάνσεις στην ισχύ, και άλλες αστάθειες που απειλούν την εύρυθμη λειτουργία των ηλεκτρικών δικτύων. Για την αντιμετώπιση αυτών των προβλημάτων, απαιτούνται λύσεις που να εξισορροπούν την προσφορά και τη ζήτηση ενέργειας σε πραγματικό χρόνο, διασφαλίζοντας έτσι την εύρυθμη λειτουργία του συστήματος. Σε αυτό το πλαίσιο, τα πολυενεργειακά συστήματα (Multi Energy System-MES), που συνδυάζουν διαφορετικά ενεργειακά συστήματα όπως το ηλεκτρικό, το θερμικό και το φυσικό αέριο, προβάλλουν ως μια λύση για την αντιμετώπιση των σύγχρονων ενεργειακών προκλήσεων.

Τα δίκτυα τηλεθέρμανσης, τα οποία αποτελούν σημαντικό μέρος των MES, μπορούν να διαδραματίσουν σημαντικό ρόλο στην ενσωμάτωση των ΑΠΕ και στην παροχή επικουρικών υπηρεσιών στα ηλεκτρικά δίκτυα. Μέσω τεχνολογιών όπως οι αντλίες θερμότητας, και οι θερμικές δεξαμενές, τα δίκτυα τηλεθέρμανσης μπορούν να προσφέρουν ευελιξία στην αποθήκευση και διαχείριση της ενέργειας, εξομαλύνοντας τις διακυμάνσεις που προκαλούνται από την αστάθεια των ΑΠΕ, μειώνοντας ταυτόχρονα την εξάρτηση από τα συμβατικά ορυκτά καύσιμα και προωθώντας τη βιώσιμη ανάπτυξη.

## 1.1 Ενσωμάτωση Ανανεώσιμων Πηγών Ενέργειας (ΑΠΕ) στα Ηλεκτρικά Δίκτυα

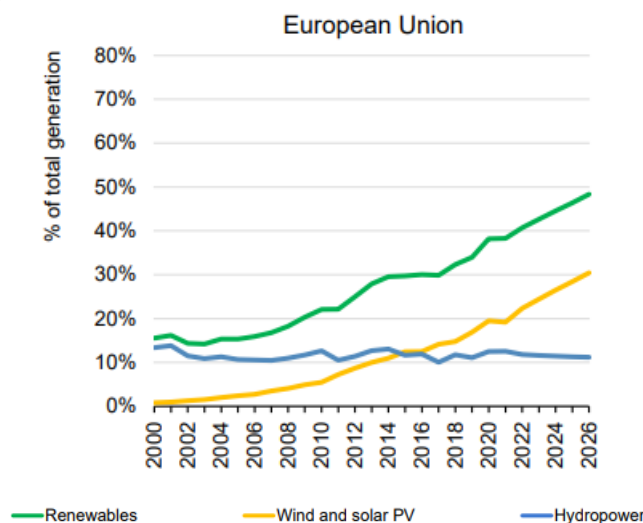
Η αυξημένη ενσωμάτωση των ΑΠΕ στα ηλεκτρικά δίκτυα αποτελεί έναν από τους βασικούς πυλώνες για την ενεργειακή μετάβαση προς πιο βιώσιμες μορφές ενέργειας. Τα τελευταία χρόνια, οι ανανεώσιμες πηγές όπως η ηλιακή και αιολική έχουν γνωρίσει τεράστια αύξηση στην παγκόσμια εγκατεστημένη ισχύ, καθιστώντας τις κεντρικές στις στρατηγικές καταπολέμησης της κλιματικής αλλαγής. Η παγκόσμια προσθήκη ΑΠΕ αναμένεται να είναι κατά μέσο όρο περίπου 305 GW ετησίως μεταξύ 2021 και 2026, 58% υψηλότερα από την προηγούμενη πενταετία όπως φαίνεται στην Εικόνα 1. Συνολικά μόνο τα φωτοβολταϊκά αποτελούν το 60% της παγκόσμιας εγκατεστημένης

ισχύς και αναμένεται στα επόμενα χρόνια να διπλασιαστεί παρόλο την αύξηση των τιμών για την παραγωγή τους. Αυτή η ταχεία αύξηση αποδεικνύει τη δέσμευση πολλών χωρών για τη μείωση των εκπομπών αερίων του θερμοκηπίου μέσω της αντικατάστασης ορυκτών καυσίμων με καθαρές μορφές ενέργειας.[1]



Εικόνα 1: Μέση ετήσια προσθήκη ισχύος ανανεώσιμων πηγών ενέργειας και εγκατεστημένη ισχύς, ιστορικά και πρόβλεψη [1]

Συγκεκριμένα η Ευρωπαϊκή Ένωση (ΕΕ) αποτελεί μία από τις πρωτοπόρες περιοχές στον κόσμο για την προώθηση των ΑΠΕ, έχοντας θέσει φιλόδοξους στόχους για την αύξηση της διείσδυσης της καθαρής ενέργειας. Στο πλαίσιο της Πράσινης Συμφωνίας [2] και της στρατηγικής <<Fit for 55>> [3], η ΕΕ στοχεύει να μειώσει τις καθαρές εκπομπές αερίων του θερμοκηπίου κατά τουλάχιστον 55% μέχρι το 2030, σε σύγκριση με τα επίπεδα του 1990. Για να επιτύχει αυτούς τους στόχους, η ΕΕ έχει δεσμευτεί να αυξήσει το μερίδιο των ΑΠΕ στην ενεργειακή της κατανάλωση. Σύμφωνα με τον κανονισμό για την ενέργεια από ανανεώσιμες πηγές, κάθε κράτος μέλος έχει υιοθετήσει εθνικούς στόχους, και συνολικά, η ΕΕ στοχεύει στο 42.5% της ενέργειάς της να προέρχεται από ΑΠΕ μέχρι το 2030 (Εικόνα 2).



Εικόνα 2: Συνολική παραγωγή ηλεκτρικής ενέργειας από ΑΠΕ, φωτοβολταϊκά και υδροηλεκτρικά κατά την περίοδο 2000-2026.[4]

## **1.2 Προκλήσεις για την Ενσωμάτωση των ΑΠΕ στα Ηλεκτρικά Δίκτυα**

Ωστόσο, η προσθήκη περισσότερων μονάδων ΑΠΕ στα υφιστάμενα δίκτυα καθιστά την αξιοπιστία και τη διαχείριση του συστήματος πιο δύσκολη. Ένα από τα κύρια προβλήματα που αντιμετωπίζουν τα ηλεκτρικά δίκτυα με την ενσωμάτωση των ΑΠΕ είναι η απρόβλεπτη και μεταβαλλόμενη φύση της παραγωγής τους. Σε αντίθεση με τις παραδοσιακές μορφές παραγωγής ενέργειας (όπως ο άνθρακας ή το φυσικό αέριο), η ηλιακή και η αιολική ενέργεια εξαρτώνται από φυσικούς παράγοντες που δεν είναι πάντα σταθεροί ή προβλέψιμοι. Αυτό οδηγεί σε διακυμάνσεις στην παραγωγή ενέργειας, οι οποίες μπορεί να επηρεάσουν την ισορροπία μεταξύ παραγωγής και ζήτησης στο δίκτυο.

### **1.2.1 Απρόβλεπτη φύση των ΑΠΕ**

Για παράδειγμα, η ηλιακή παραγωγή μπορεί να μειωθεί απότομα κατά τη διάρκεια μιας συννεφιασμένης ημέρας ή να μηδενιστεί τη νύχτα, ενώ η αιολική ενέργεια μπορεί να ποικίλλει δραματικά ανάλογα με την ταχύτητα του ανέμου. Αυτές οι διακυμάνσεις καθιστούν δύσκολη τη διατήρηση της σταθερότητας του συστήματος και της ποιότητας της παροχής ενέργειας. Για να εξισορροπηθούν οι διαφορές μεταξύ της παραγωγής και της ζήτησης ηλεκτρικής ενέργειας, μπορεί να χρειαστεί η μόνιμη λειτουργία εφεδρικών γεννητριών, η ενσωμάτωση μπαταριών ή η αποσύνδεση μονάδων ΑΠΕ όταν η παραγωγή υπερβαίνει τη ζήτηση. Και στις τρεις περιπτώσεις όμως υπάρχει πιθανότητα οικονομικής ζημίας ενώ οι τεχνολογίες αποθήκευσης είναι ακόμα ακριβές και η ευρεία τους εφαρμογή παραμένει πρόκληση [5], [6].

### **1.2.2 Δίκτυα διανομής**

Στα δίκτυα διανομής παρατηρούνται προβλήματα με την υπέρβαση των ορίων τάσης, καθώς οι ώρες υψηλής ζήτησης ενδέχεται να μην συμπίπτουν με τις ώρες αιχμής της παραγωγής από ΑΠΕ. Η διαχείριση της τάσης είναι απαραίτητη σε ένα ηλεκτρικό δίκτυο και δεν πρέπει να παραβλέπεται, καθώς τα όρια τάσης διαφέρουν ανάλογα με το επίπεδο ισχύος που έχουν καθοριστεί βάσει ευρωπαϊκών προτύπων [7]. Η ικανότητα του δικτύου να λειτουργεί με ευελιξία είναι καθοριστική για τη διατήρηση της ισορροπίας μεταξύ προσφοράς και ζήτησης ηλεκτρικής ενέργειας, με απώτερο σκοπό την ενσωμάτωση ΑΠΕ μεγάλης κλίμακας.

### **1.2.3 Δίκτυα διανομής χαμηλής τάσης**

Στα δίκτυα διανομής χαμηλής τάσης, οι υπερτάσεις αποτελούν το κύριο πρόβλημα που προκαλείται από τη μαζική διείσδυση μονάδων ΑΠΕ. Αυτό οφείλεται στο γεγονός ότι η διοχέτευση ενεργού ισχύος σε σημεία του δικτύου δημιουργεί αντίστροφη ροή, η οποία, σε συνδυασμό με την ωμική φύση του ηλεκτρικού δικτύου, αυξάνει την τάση. Σε δίκτυα μέσης τάσης, αυτό το ζήτημα μπορεί να αντιμετωπιστεί με τη διαχείριση της άεργου ισχύος. Ωστόσο, στα δίκτυα χαμηλής τάσης, μια τέτοια προσέγγιση θα προκαλούσε υψηλές απώλειες, καθώς ο λόγος της ωμικής αντίστασης προς την αυτεπαγωγή είναι υψηλός. Ο πλέον αποδοτικός τρόπος για την αντιμετώπιση των υπερτάσεων σε αυτά τα δίκτυα είναι η μείωση της ενεργού ισχύος από τις μονάδες ΑΠΕ [8].

### 1.2.4 Μικροδίκτυα

Ένα επιπλέον πρόβλημα που εμφανίζεται σε μικροδίκτυα (microgrids) είναι οι μικρές διακυμάνσεις της συχνότητας, που οφείλονται στις μεταβολές της παραγωγής ενέργειας από μονάδες ΑΠΕ. Απότομες αλλαγές στις καιρικές συνθήκες ή ακραία καιρικά φαινόμενα, για παράδειγμα καταιγίδες, μπορούν να μειώσουν την παραγωγή των φωτοβολταϊκών σχεδόν στο μηδέν. Η εγκατάσταση μπαταριών, είτε σε τοπικό επίπεδο είτε διανεμημένα μέσα στο σύστημα, συμβάλλει στην εξομάλυνση αυτών των διακυμάνσεων, ώστε οι καταναλωτές να μην αντιλαμβάνονται αυτές τις διαταραχές. [9]

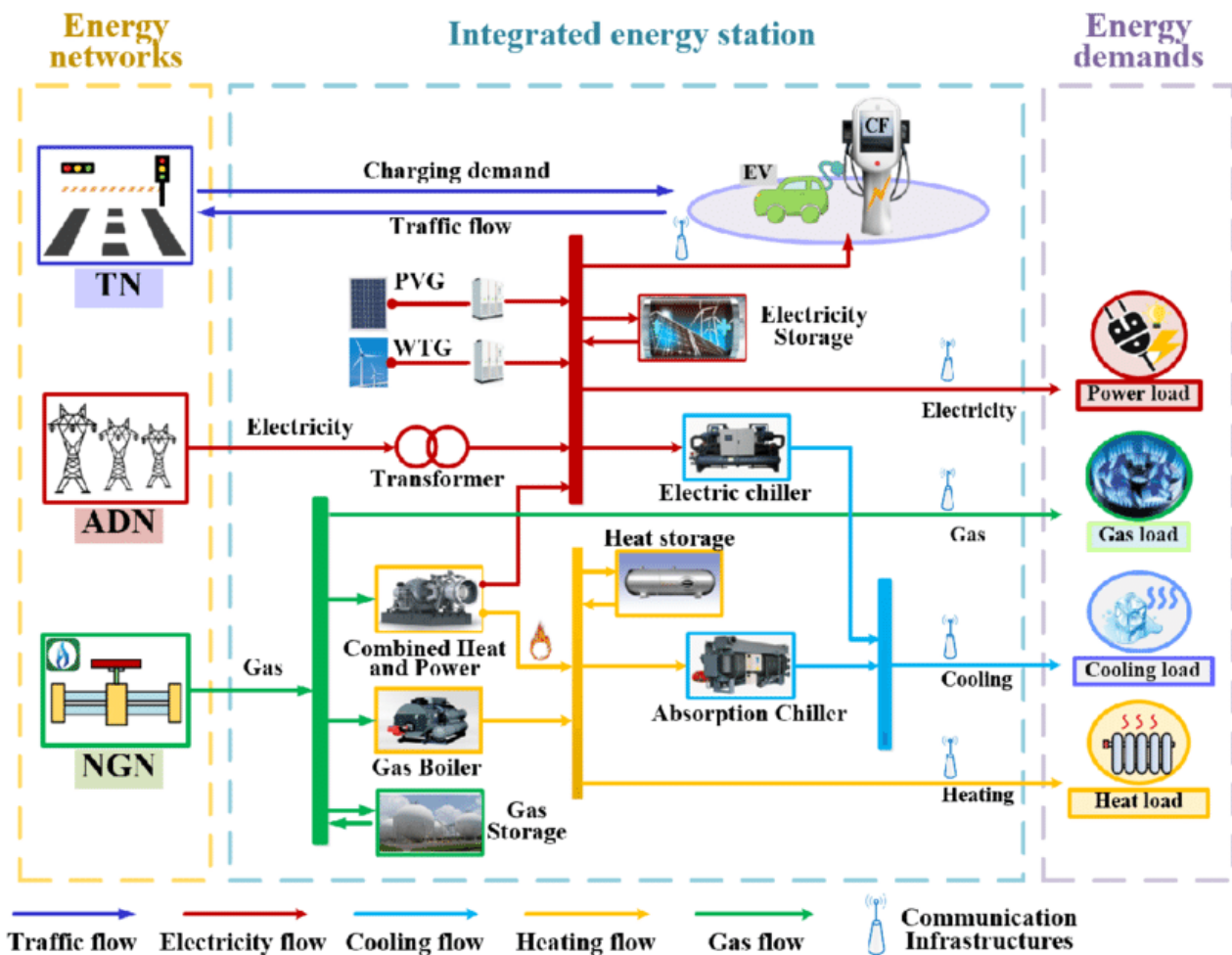
## 1.3 Πολυενεργειακό σύστημα

Ένα ενεργειακό σύστημα (Multi Energy System-MES) αναφέρεται σε οποιοδήποτε τεχνολογικό σύστημα που στοχεύει στη μεταφορά, μετατροπή ή αποθήκευση ενέργειας. Υπάρχουν διάφοροι τύποι συστημάτων παροχής ενέργειας, όπως το ηλεκτρικό, το θερμικό και το σύστημα φυσικού αερίου, τα οποία συνήθως λειτουργούν ανεξάρτητα το ένα από το άλλο. Συνήθως, κάθε ένα από αυτά τα συστήματα εξετάζεται ξεχωριστά ή με περιορισμένη συνέργεια με τα υπόλοιπα. Ωστόσο, η ανάλυσή τους ως ενιαίο σύστημα μπορεί να προσφέρει σημαντικά οφέλη. Η πλήρης κατανόηση όλων των ροών ισχύος επιτρέπει την ασφαλέστερη διαχείριση της κάθε παροχής, ενώ παράλληλα λαμβάνονται υπόψη και μειώνονται οι ενεργειακές απώλειες, τα κόστη και οι περιβαλλοντικοί ρύποι [9]. Από την άλλη πλευρά, οι ξεχωριστές αναλύσεις καθιστούν πιο δύσκολη την επίτευξη βέλτιστου σχεδιασμού.

Οι αλληλεπιδράσεις μεταξύ των ενεργειακών συστημάτων πάντα υπήρχαν και αυξάνονται συνεχώς. Για παράδειγμα, σύνδεση του ηλεκτρικού δικτύου με τα δίκτυα θερμότητας, ψύξης ή και φυσικού αερίου είναι από τις πιο διαδεδομένες περιπτώσεις. Η διασύνδεση επιτυγχάνεται συχνά μέσω τεχνολογιών όπως η συμπαραγωγή ηλεκτρισμού και θερμότητας (Combined Heat and Power-CHP), οι ηλεκτρικές αντλίες θερμότητας (Electrical Heat Pumps-EHPs), τα συστήματα κλιματισμού και άλλες συσκευών. Παρομοίως, οι αλληλεπιδράσεις μεταξύ ηλεκτρικής ενέργειας, των καυσίμων και των μεταφορών προβλέπονται να αυξηθούν ή ήδη βρίσκονται σε εξέλιξη, όπως με τη χρήση ηλεκτρικών οχημάτων (Electrical Vehicles-EVs), βιοκαυσίμων και μεταφορών που βασίζονται στις κυψέλες καυσίμου υδρογόνου. Ο στόχος αυτών των προσπαθειών είναι η δημιουργία ενός πιο καθαρού και ενεργειακά αποδοτικού συστήματος, όπου διαφορετικές μορφές ενέργειας συνυπάρχουν και συνεργάζονται. Ένα τέτοιο σύστημα αναφέρεται ως πολυενεργειακό [9].

Σε ένα πολυενεργειακό σύστημα, η διαχείριση των παροχών αποτελεί αρμοδιότητα των έξυπνων δικτύων. Η λειτουργία ενός έξυπνου δικτύου βασίζεται στην ανάλυση δεδομένων από μετρητές, με στόχο την αποτελεσματικότερη διαχείριση, την παροχή εργαλείων και υπηρεσιών στους χρήστες, ώστε να προσαρμόζουν τη συμπεριφορά τους ανάλογα με τον ρόλο τους στο σύστημα, καθώς και τη βελτιστοποίηση της λειτουργίας του συστήματος μέσω των διαθέσιμων πόρων και υποδομών. Για να εναρμονιστεί με τη λειτουργία των πολυενεργειακών συστημάτων, το έξυπνο δίκτυο επεκτείνεται πέρα από τον ηλεκτρικό τομέα, συμπεριλαμβάνοντας και άλλους τύπους δικτύων.





Εικόνα 3: Η αρχιτεκτονική ενός πολύ-ενεργειακού συστήματος[10]

Για να εξασφαλιστεί η μέγιστη απόδοση ενός πολυενεργειακού συστήματος, δεν αρκεί μόνο η βέλτιστη λειτουργία των επιμέρους δικτύων. Απαιτείται επίσης η αποδοτική μετατροπή της ενέργειας, ώστε να ελαχιστοποιούνται οι απώλειες στα σημεία σύνδεσης. Παράλληλα, ενισχύεται η διανεμημένη παραγωγή, ενώ προωθείται η συγκεντρωμένη εγκατάσταση πόρων σε στρατηγικά σημεία για καλύτερη διαχείριση της απόκρισης στη ζήτηση. Με αυτόν τον τρόπο, η ενέργεια από ΑΠΕ μπορεί να καλύψει άμεσα τη ζήτηση, τόσο στο κύριο σύστημα όσο και στα γειτονικά.

## 1.4 Αντλία θερμότητας

Μεταξύ των διαφόρων τύπων MES η συνδυασμένη λειτουργία των δικτύων διανομής χαμηλής τάσης και των δικτύων τηλεθέρμανσης είναι πιο διαδεδομένη. Ανάμεσα στις διάφορες τεχνολογίες που μπορούν να χρησιμοποιηθούν, οι αντλίες θερμότητας ξεχωρίζουν. Πράγματι, οι αντλίες θερμότητας διακρίνονται για την υψηλή απόδοση στη μετατροπή από πρωτογενή σε τελική ενέργεια, γνωστή ως συντελεστής απόδοσης (COP), καθιστώντας τις ιδανική λύση για την αποδοτικότερη αξιοποίηση της ενέργειας που παράγεται από τις κατανεμημένες μονάδες ανανεώσιμων πηγών [11].

Οι αντλίες θερμότητας, με τον υψηλό συντελεστή απόδοσης που διαθέτουν, αξιοποιούν καλύτερα τις ΑΠΕ. Αυτό οφείλεται στο γεγονός ότι λειτουργούν ως φορτία στο ηλεκτρικό δίκτυο, αυξάνοντας

τη ζήτηση, κάτι ωφέλιμο ειδικά σε περιόδους υπερπαραγωγής από ΑΠΕ. Με αυτόν τον τρόπο, για να αποφευχθεί η ανάγκη μείωσης της ενεργού ισχύος που παράγεται από τους παραγωγούς μπορούν να εγκατασταθούν αντλίες θερμότητας. Παράλληλα, αυτές συμβάλλουν στην παροχή θερμότητας στο δίκτυο τηλεθέρμανσης για την κάλυψη των τοπικών θερμικών αναγκών.

## **1.5 Αντικείμενο διπλωματικής**

Η παρούσα διπλωματική εργασία επικεντρώνεται στη μοντελοποίηση ενός πολυενεργειακού συστήματος, το οποίο συνδυάζει το ηλεκτρικό και το θερμικό δίκτυο, και διερευνά πώς το δίκτυο τηλεθέρμανσης μπορεί να συνδράμει στην ενίσχυση του ηλεκτρικού δικτύου, μεγιστοποιώντας παράλληλα την απορρόφηση ανανεώσιμων πηγών ενέργειας (ΑΠΕ) και ελαχιστοποιώντας τις σχετικές περικοπές. Η υλοποίηση βασίζεται σε ένα ήδη υπάρχον λεπτομερές μοντέλο και πραγματοποιείται με τη χρήση της γλώσσας προγραμματισμού Python, αξιοποιώντας δύο εξειδικευμένες βιβλιοθήκες, την pandarower και την pandaripes. Μέσω αυτών των εργαλείων, δημιουργούνται και προσομοιώνονται το ηλεκτρικό και το θερμικό δίκτυο, μια θερμική μπαταρία - δεξαμενή και οι αντίστοιχοι ελεγκτές. Η ηλεκτρική ισχύς μετατρέπεται σε θερμική και διοχετεύεται στο δίκτυο τηλεθέρμανσης, με τη διασύνδεση να επιτυγχάνεται μέσω της μεταφοράς ισχύος από το ηλεκτρικό στο θερμικό δίκτυο, χωρίς αντίστροφη ροή. Η επιτυχής διασύνδεση επιβεβαιώνεται μέσω της ρύθμισης της τάσης, καθώς το προτεινόμενο μοντέλο επικεντρώνεται στον έλεγχο της τάσης που επιτυγχάνεται με τη δημιουργία πολυενεργειακών συστημάτων. Επιπλέον, η μελέτη εστιάζει ιδιαίτερα στους ελεγκτές: ο ελεγκτής τάσης από την πλευρά του ηλεκτρικού δικτύου και ο ελεγκτής θερμότητας από το θερμικό δίκτυο. Οι ελεγκτές αυτοί αποτελούν τον πιο κρίσιμο παράγοντα της μελέτης, καθώς καθορίζουν το σημείο ρύθμισης της κατανάλωσης ισχύος της αντλίας θερμότητας και τον τρόπο με τον οποίο θα συμβάλει στη συνολική λειτουργία του συστήματος.

## ***Κεφάλαιο 2: Βιβλιογραφική Ανασκόπηση***

---

Η ιδέα της σύνδεσης ενός ηλεκτρικού δικτύου με ένα δίκτυο τηλεθέρμανσης δεν είναι καινούργια. Από τα τέλη του 2000 και αρχές της δεκαετίας 2010 άρχισε να χρησιμοποιείται ο όρος MES όταν οι ερευνητές άρχισαν να συνειδητοποιούν τη σημασία της σύνδεσης διαφορετικών μορφών ενέργειας σε ένα ενιαίο σύστημα για την αύξηση της αποδοτικότητας και της αξιοπιστίας. Σημαντικές μελέτες και συνέδρια έχουν επικεντρωθεί στην μελέτη τους ανά τα χρόνια.

Από τις πρώτες μελέτες που έγιναν πάνω στην συμπαραγωγή ηλεκτρικής και θερμικής ενέργειας είναι αυτήν του [12] το 2007. Παρουσιάζει μια προσέγγιση για τη βελτιστοποίηση της ροής ισχύος σε συστήματα που περιλαμβάνουν πολλαπλές ενεργειακές υποδομές, όπως η ηλεκτρική ενέργεια, το φυσικό αέριο και η τηλεθέρμανση. Επικεντρώνεται στη μοντελοποίηση της ροής ισχύος σε σταθερή κατάσταση, τη μεταφορά και τη μετατροπή πολλαπλών ενεργειακών φορέων με τη χρήση μιας νέας έννοιας των ενεργειακών κόμβων όπου οι είσοδοι και οι έξοδοι του συστήματος συνδέονται μέσω ενός πίνακα σύζευξης, ο οποίος βασίζεται στα χαρακτηριστικά των συσκευών μετατροπής και αποθήκευσης ενέργειας. Με αυτό το μοντέλο, διατυπώνονται τα προβλήματα συνδυασμένης οικονομικής διαχείρισης και βέλτιστης ροής ισχύος, τα οποία καλύπτουν τη μεταφορά και τη μετατροπή ενέργειας. Ωστόσο, αυτή η μεθοδολογία μοντελοποίησης αγνοεί τα χαρακτηριστικά του δικτύου, γεγονός που οδηγεί σε ανακριβή αποτελέσματα.

Από τις αρχές της δεκαετίας του 2010, η τηλεθέρμανση έχει αναδειχθεί ως σημαντικός τομέας έρευνας, όπως καταδεικνύεται στο [13]. Το άρθρο αυτό εξετάζει τον ρόλο της τηλεθέρμανσης σε μελλοντικά συστήματα ανανεώσιμης ενέργειας, υπογραμμίζοντας τη σημασία της στην προώθηση της βιώσιμης ενεργειακής μετάβασης. Η μελέτη αναλύει διάφορα μοντέλα και σενάρια ενεργειακής πολιτικής που ενσωματώνουν την τηλεθέρμανση με υψηλά ποσοστά χρήσης ΑΠΕ, περιλαμβάνοντας την αξιοποίηση τεχνολογιών θερμικής αποθήκευσης και την ανάκτηση αποβλήτων θερμότητας από βιομηχανικές διεργασίες. Τα οικονομικά και περιβαλλοντικά οφέλη της που προκύπτουν συνδυάζοντας ηλεκτρικά και θερμικά δίκτυα ενέργειας, όπως η μείωση των εκπομπών CO<sub>2</sub> και η αύξηση της ενεργειακής απόδοσης, καθώς και οι προκλήσεις που σχετίζονται με την υποδομή και τις τεχνολογικές απαιτήσεις, διερευνώνται λεπτομερώς. Για την επίτευξη αυτών, οι συγγραφείς χρησιμοποιούν μεθόδους ανάλυσης ενεργειακών συστημάτων και προσομοιώσεις για την αξιολόγηση της αλληλεπίδρασης μεταξύ της τηλεθέρμανσης και διαφόρων ενεργειακών τεχνολογιών, όπως οι αντλίες θερμότητας, οι λέβητες βιομάζας και τα φωτοβολταϊκά συστήματα. Τα αποτελέσματα υποδεικνύουν ότι η ενσωμάτωση θερμικής αποθήκευσης στα ηλεκτρικά δίκτυα μπορεί να ενισχύσει τη σταθερότητα και να μειώσει τις ενεργειακές απώλειες. Παρόλα αυτά, η μελέτη επικεντρώνεται κυρίως σε ένα μοντέλο που αφορά τη Δανία, αφήνοντας ανοιχτό το πεδίο για περαιτέρω ανάλυση σε διαφορετικά γεωγραφικά και πολιτικά πλαίσια.

Συνεχίζοντας στα μέσα της δεκαετίας του 2010 το άρθρο [9] παρέχει μια συνολική επισκόπηση των πολυενεργειακών συστημάτων, των βασικών εννοιών τους, και των μοντέλων αξιολόγησης που χρησιμοποιούνται για την ανάλυσή τους. Το άρθρο περιγράφει διάφορα μοντέλα και μεθόδους που χρησιμοποιούνται για την αξιολόγηση των MES, όπως μοντέλα βελτιστοποίησης, προσομοιώσεις, στοχαστικά και διάφορα μοντέλα δικτύου. Αυτά επιτρέπουν στους ερευνητές και τους μηχανικούς να αναλύσουν την απόδοση, το κόστος, τις περιβαλλοντικές επιπτώσεις και τις τεχνικές απαιτήσεις των πολυενεργειακών συστημάτων σε διάφορα σενάρια. Όμως η πολυπλοκότητα του σχεδιασμού και της λειτουργίας των MES, καθώς και οι ανάγκες για δεδομένα και υπολογιστική ισχύ, αποτελούν σημαντικές προκλήσεις. Υπάρχουν επίσης οικονομικές και ρυθμιστικές προκλήσεις που απαιτούν συνεργασία μεταξύ των φορέων και σαφείς κατευθυντήριες γραμμές για την εφαρμογή των MES.

Λίγα χρόνια αργότερα, ένα νέο δεδομένο εισήχθη στη μοντελοποίηση του θερμικού δικτύου σε συνεργασία με το ηλεκτρικό δίκτυο στο [14]. Συγκεκριμένα, η μεταβλητή ροή μάζας και η μεταβλητή θερμοκρασίας αποτέλεσαν βασικά στοιχεία στο πείραμα που διεξήχθη στο νησί Μπάρι της Ουαλίας, με στόχο τη δημιουργία ενός αυτόνομου συστήματος, χωρίς διασύνδεση με εξωτερικά δίκτυα. Το σύστημα στηρίχθηκε σε μονάδες σύνδεσης όπως οι αντλίες θερμότητας και οι ηλεκτρικοί λέβητες. Δεδομένου ότι η έρευνα πραγματοποιήθηκε σε νησί, η έννοια της νησιώτικης (islanded) λειτουργίας θεωρήθηκε κεντρική. Δύο μέθοδοι αναπτύχθηκαν για τη συνδυασμένη ανάλυση της απόδοσης των δικτύων ηλεκτρικής ενέργειας και θερμότητας ως ενιαίο

σύνολο. Η πρώτη μέθοδος ήταν η αποσυνθετική προσέγγιση, ενώ η δεύτερη ήταν η ολοκληρωτική τεχνική ηλεκτρικού-υδραυλικού-θερμικού υπολογισμού, με τη δεύτερη να απαιτεί λιγότερες επαναλήψεις λόγω της αυξημένης πολυπλοκότητας των δικτύων στην αποσυνθετική μέθοδο. Παρόλα αυτά, υπάρχουν ορισμένοι περιορισμοί, όπως το γεγονός ότι η ανάλυση βασίζεται σε σταθερές συνθήκες (steady-state conditions), οι οποίες δεν λαμβάνουν υπόψη τις δυναμικές μεταβολές στη ζήτηση ενέργειας ή τις καιρικές συνθήκες. Αυτό μπορεί να επηρεάσει σημαντικά την ακρίβεια του μοντέλου σε πραγματικές συνθήκες, όπου η ζήτηση ενέργειας παρουσιάζει διακυμάνσεις. Επιπλέον, η χρήση απλοποιημένων μοντέλων περιορίζει την ακρίβεια των προβλέψεων και την εφαρμογή σε μεγάλης κλίμακας συστήματα.

Προς το τέλος της δεκαετίας, το άρθρο [11] διερευνά το πρόβλημα βελτιστοποίησης του σχεδιασμού για αντλίες θερμότητας μεγάλης κλίμακας στα δίκτυα θέρμανσης και ηλεκτρικής ενέργειας, με σκοπό τη μείωση του συνολικού κόστους και τη μεγιστοποίηση των ωφελειών για όλους τους εμπλεκόμενους φορείς. Παρουσιάζεται μια μέθοδος για τον καθορισμό των βέλτιστων τοποθεσιών και μεγεθών των αντλιών θερμότητας, με στόχο την αποδοτική τους λειτουργία με το χαμηλότερο δυνατό κόστος, λαμβάνοντας υπόψη παράγοντες όπως η διαθεσιμότητα πηγών θερμότητας (π.χ. υπόγεια ύδατα, θαλασσινό νερό) και η εγγύτητα στους καταναλωτές θερμότητας. Η μελέτη προτείνει μια προσέγγιση τριών επιπέδων βελτιστοποίησης: αρχικά, πραγματοποιείται η βελτιστοποίηση του δικτύου τηλεθέρμανσης, στη συνέχεια του ηλεκτρικού δικτύου, και τέλος, προτείνεται μια συνδυασμένη προσέγγιση κόστους για την εξισορρόπηση και των δύο συστημάτων. Η ανάλυση έδειξε ότι ο βέλτιστος σχεδιασμός εξαρτάται σε μεγάλο βαθμό από τη διαθεσιμότητα των θερμικών πόρων και την υφιστάμενη ηλεκτρική υποδομή της περιοχής, με το συνολικό κόστος να μειώνεται όταν η αντλία τοποθετείται σε τοποθεσίες που προσφέρουν μεγαλύτερη απόδοση και υψηλότερη ενεργειακή απόδοση. Όμως, το προτεινόμενο μοντέλο βασίζεται σε στατικές συνθήκες και επικεντρώνεται στη Δανία, συγκεκριμένα σε μια αστική περιοχή της Κοπεγχάγης, για την ανάλυση μιας μόνο αντλίας, χωρίς να λαμβάνει υπόψη τις δυναμικές μεταβολές, όπως οι αλλαγές στη ζήτηση ενέργειας. Επιπλέον, το μοντέλο περιλαμβάνει απλοποιήσεις στα μαθηματικά υποδείγματα και δεν εξετάζει πλήρως τις επιπτώσεις της αποθήκευσης ενέργειας.

Σύμφωνα με τις παραπάνω έρευνες [9] έως [13], η πολυπλοκότητα που προκύπτει από τη μη ισορροπημένη λειτουργία του ηλεκτρικού δικτύου δεν έχει αντιμετωπιστεί επαρκώς στην πλειονότητα της βιβλιογραφίας. Το ζήτημα αυτό αντιμετωπίζεται εν μέρη στο άρθρο [15] που εξετάζει τη μοντελοποίηση των τοπικών αγορών ενέργειας για συστήματα MES σε ασύμμετρα δίκτυα χαμηλής τάσης. Ειδικότερα, επικεντρώνεται σε πώς μπορούν να οργανωθούν και να λειτουργήσουν τοπικές ενεργειακές κοινότητες σε τέτοια δίκτυα, λαμβάνοντας υπόψη την αλληλεπίδραση μεταξύ διαφορετικών τύπων ενέργειας (π.χ., ηλεκτρισμός, θερμότητα, αέριο). Η έρευνα εξετάζει ένα τριφασικό μοντέλο που ενσωματώνει τις ανισορροπίες που προκαλούνται από τη σύνδεση μονοφασικών φορτίων και των φορτίων ΑΠΕ έναντι του παραδοσιακής μονοφασικής σύνδεσης. Παρότι οι βέλτιστες λύσεις και για τα δύο μοντέλα ήταν παρόμοιες, το μονοφασικό μοντέλο αποδείχθηκε καταλληλότερο για προβλήματα σχεδιασμού που περιλαμβάνουν εκτεταμένα σενάρια, ενώ το τριφασικό ήταν πιο αποτελεσματικό στην επίλυση καθημερινών λειτουργικών ζητημάτων, προσφέροντας μια πιο ακριβή ανάλυση στην τοπική ανταλλαγή ενέργειας και στη συμπεριφορά της τάσης. Παρά τις σημαντικές συνεισφορές του, το άρθρο θα μπορούσε να επεκταθεί περαιτέρω, εξετάζοντας την ενσωμάτωση πρόσθετων ενεργειακών φορέων, τη δυναμική ανάλυση του δικτύου, καθώς και την εφαρμογή του μοντέλου σε μεγαλύτερη κλίμακα. Επίσης, η εισαγωγή εμπορικών και ρυθμιστικών παραμέτρων θα μπορούσε να ενισχύσει την αξιοπιστία του μοντέλου και την εφαρμογή του σε πραγματικές αγορές ενέργειας.

Η ενσωμάτωση συστημάτων ελέγχου τόσο στο ηλεκτρικό δίκτυο όσο και στο δίκτυο τηλεθέρμανσης έχει αρχίσει να εξετάζεται πρόσφατα. Συγκεκριμένα στο [16], η αντλία θερμότητάς ελέγχεται ώστε να ανταποκριθεί στα προγράμματα διαχείρισης της ζήτησης από την πλευρά του θερμικού δικτύου. Για τη βελτιστοποίηση της λειτουργίας, χρησιμοποιείται ένα μοντέλο προβλεπτικού ελέγχου (Model Predictive Control - MPC), το οποίο λαμβάνει σήματα από τον Διαχειριστή Συστήματος Διανομής (Distribution System Operator - DSO), για να μειώσει ή να αυξήσει την κατανάλωση ενέργειας. Το MPC επιτρέπει τη διαχείριση της λειτουργίας της αντλίας θερμότητας μέσω διαδικασιών όπως η αποκοπή αιχμής, όπου η αντλία θερμότητας διακόπτει τη λειτουργία της κατά τις ώρες αιχμής και η αποθηκευμένη θερμότητα καλύπτει τις ανάγκες των κτιρίων. Κατά τις περιόδους χαμηλής ζήτησης, η κατανάλωση της αντλίας αυξάνεται για τη φόρτιση της θερμικής δεξαμενής. Παρά τη σημαντική συμβολή του άρθρου στην ευελιξία των ενεργειακών δικτύων μέσω της χρήσης αντλιών θερμότητας και θερμικής αποθήκευσης, υπάρχουν ορισμένοι περιορισμοί. Το σύστημα ελέγχου βασίζεται σε προβλέψεις του Διαχειριστή Συστήματος Διανομής, γεγονός που απαιτεί ακριβή εκτίμηση της ζήτησης και των συνθηκών του δικτύου. Η έρευνα επισημαίνει ότι το σύστημα πρέπει να λαμβάνει έγκαιρες ειδοποιήσεις για τις αιχμές κατανάλωσης, ώστε να μπορεί να προσαρμόσει τη φόρτιση της θερμικής αποθήκευσης. Αυτό συνεπάγεται μια χρονική υστέρηση μεταξύ της ειδοποίησης και της εφαρμογής της δράσης, γεγονός που μπορεί να περιορίσει την αποδοτικότητα σε περιπτώσεις όπου οι αιχμές κατανάλωσης δεν μπορούν να προβλεφθούν με ακρίβεια ή εμφανίζονται ξαφνικά.

## Κεφάλαιο 3: Μεθοδολογία

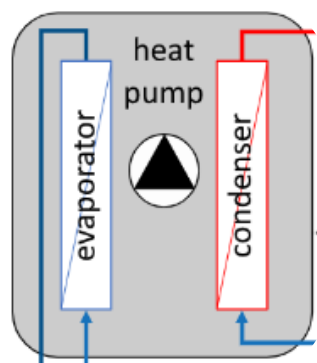
---

### 3.1 Σύνδεση ηλεκτρικού με το δίκτυο τηλεθέρμανσης

Η σύνδεση και ανάλυση του ηλεκτρικού δικτιού με το δίκτυο τηλεθέρμανσης υλοποιήθηκε με την βοήθεια ενός εργαλείου το EriGrid και συγκεκριμένα το μοντέλο Benchmark. Το EriGrid 2.0 (European Research Infrastructure supporting Smart Grid Systems, Technology, Development, Validation, and Rollout) είναι ένα ευρωπαϊκό ερευνητικό έργο που επικεντρώνεται στην έρευνα και την ανάπτυξη έξυπνων συστημάτων δικτύων. Με βάση τα αποτελέσματα του πρώτου EriGrid-1 αναπτύχθηκε το δεύτερο με σκοπό να διευρύνει τις ερευνητικές υπηρεσίες και τα εργαλεία των ευρωπαϊκών ερευνητικών υποδομών για την αξιολόγηση των έξυπνων ενεργειακών δικτύων, με κύρια έμφαση στο ηλεκτρικό δίκτυο. Το Benchmark Multi-Energy Network μοντέλο είναι ένα από τα αποτελέσματα του έργου EriGrid. Αυτό το μοντέλο παρέχει ένα πρότυπο για την προσομοίωση και την ανάλυση πολυενεργειακών δικτύων, δηλαδή δικτύων που ενσωματώνουν πολλαπλές μορφές ενέργειας (ηλεκτρική, θερμική, αέρια κ.λπ.) σε μία ενιαία υποδομή.

Η υλοποίηση της σύνδεσης των δύο δικτύων γίνεται με την βοήθεια της rython και συγκεκριμένα δυο βιβλιοθήκες της, την pandarower για την αναπαράσταση του ηλεκτρικού και την pandaripes για του θερμικού δικτύου. Ο κώδικας απαρτίζεται από τέσσερις βασικές κλάσεις. Το ηλεκτρικό δίκτυο, το θερμικό δίκτυο, την εγκατάσταση μετατροπής της ηλεκτρικής ενέργειας σε θερμική δηλαδή την αντλία θερμότητας και τον έλεγχο. Το ηλεκτρικό δίκτυο αποτελείται από έναν κόμβο εξωτερικής τροφοδοσίας(άπειρος ζυγός), ζυγοί, γραμμές, φορτία και γεννήτριες-φωτοβολταϊκά. Το θερμικό δίκτυο αποτελείται από έναν κόμβο θερμικής εξωτερικής τροφοδοσίας(άπειρος ζυγός) αντίστοιχα, εναλλάκτη θερμότητας- καταναλωτές, παράκαμψη (bypass), σωλήνες, κόμβοι. Η εγκατάσταση μετατροπής της ηλεκτρικής ενέργειας σε θερμική περιλαμβάνει την αντλία θερμότητας, τον εξατμιστή (evaporator), τον συμπυκνωτή (condenser), την δεξαμενή και αντλίες. Τέλος η κλάση του ελέγχου έχει τον ελεγκτή θερμότητας (Flex Heat Controller) και τον ελεγκτή τάσης (Voltage Controller) όπου η λειτουργία των ελεγκτών θα αναλυθεί παρακάτω.

Η αντλία θερμότητας αποτελείται από τον συμπυκνωτή και τον εξατμιστή όπως φαίνεται στην Εικόνα 4. Η πηγή για τον εξατμιστή είναι η γραμμή επιστροφής του θερμικού δικτύου. Η έξοδος του συμπυκνωτή τροφοδοτείται στη δεξαμενή θερμικής αποθήκευσης.



Εικόνα 4: Αντλία θερμότητας[17]

Η ηλεκτρική ενέργεια που παράγεται από τα φωτοβολταϊκά συστήματα που είναι συνδεδεμένα στο δίκτυο μετατρέπεται σε θερμική ενέργεια μέσω μιας αντλίας θερμότητας και αποθηκεύεται σε μια δεξαμενή ζεστού νερού. Αυτή η δεξαμενή λειτουργεί ως αποκεντρωμένη πηγή θερμότητας, υποστηρίζοντας το τοπικό δίκτυο τηλεθέρμανσης. Παράλληλα, η εγκατάσταση αυτή, που μετατρέπει την ενέργεια σε θερμότητα, συμβάλλει στη βελτίωση της σταθερότητας του ηλεκτρικού δικτύου. Η λειτουργία ολόκληρου του συστήματος ελέγχεται από δύο ελεγκτές.

### 3.1.1 Ελεγκτής τάσης

Ο ελεγκτής τάσης, συνδέεται στον ζυγό της επιλογή μας, επιβλέπει την τάση στο ηλεκτρικό δίκτυο και καθορίζει την απαιτούμενη ισχύ για την αντλία θερμότητας με σκοπό να διατηρείται η τάση εντός των επιτρεπτών ορίων. Σε περιόδους υψηλής παραγωγής από φωτοβολταϊκά και χαμηλής κατανάλωσης, μπορεί να προκληθεί υπέρταση και συμφόρηση στο ηλεκτρικό δίκτυο. Σε αυτή την περίπτωση, ο ελεγκτής τάσης θα ζητήσει να αυξηθεί η κατανάλωση ισχύος από την αντλία θερμότητας, αν αυτό είναι εφικτό. Αντίθετα, σε περιόδους υψηλού φορτίου, όταν υπάρχει υπόταση, θα ζητήσει να μειωθεί η ισχύς της αντλίας θερμότητας ή ακόμη και να διακοπεί η λειτουργία της.

### 3.1.2 Ελεγκτής θερμότητας

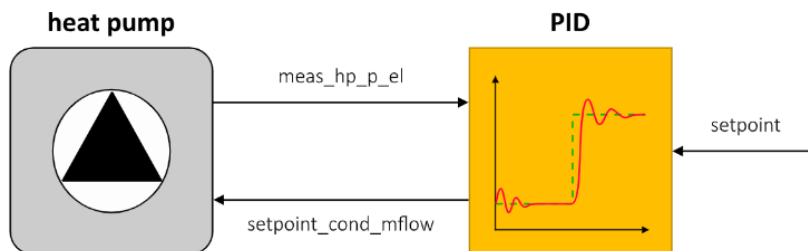
Ο ελεγκτής θερμότητας καθορίζει αν η παροχή θερμότητας θα καλυφθεί αποκλειστικά από το εξωτερικό δίκτυο ή αν εγκατάσταση μετατροπής της ηλεκτρικής ενέργειας σε θερμική θα συνεισφέρει με την απελευθέρωση θερμότητας από την δεξαμενή. Η θερμοκρασία της δεξαμενής αποθήκευσης παρακολουθείται συνεχώς και διατηρείται εντός των επιτρεπτών ορίων. Εάν χρειαστεί, η αντλία θερμότητας ενεργοποιείται για να φορτίσει τη δεξαμενή, λαμβάνοντας υπόψη το σημείο ρύθμισης κατανάλωσης ισχύος που ορίζει ο ελεγκτής τάσης (δηλαδή, η κατανάλωση ισχύος δεν υπερβαίνει το καθορισμένο όριο, αλλά μπορεί να είναι μικρότερη). Ο ελεγκτής θερμότητας επηρεάζει την λειτουργία της εγκατάστασης μετατροπής της ηλεκτρικής ενέργειας σε θερμική με το να ελέγχει τις αντλίες των δύο ξεχωριστών θερμοϋδραυλικών βρόγχων.

Ο 1ος βρόχος. Ζεστό νερό αντλείται από την δεξαμενή αποθήκευσης και τροφοδοτεί την γραμμή τροφοδοσίας του θερμικού δικτύου με την βοήθεια μιας αντλίας. Παράλληλα ίση ποσότητα κρύου νερού αντλείται από την γραμμή επιστροφής του δικτύου θέρμανσης για να ξαναγεμίσει η δεξαμενή. Η λειτουργία της υδραυλικής αντλίας ρυθμίζεται ανάλογα με τα επίπεδα θερμοκρασίας στη δεξαμενή αποθήκευσης.

Ο 2ος βρόχος. Κρύο νερό αντλείται από το κάτω μέρος της δεξαμενής αποθήκευσης, θερμαίνεται από την αντλία θερμότητας, συγκεκριμένα από τον συμπυκνωτή, και τοποθετείται στην κορυφή

της δεξαμενής με τη βοήθεια μιας άλλης υδραυλικής αντλίας. Ο ρυθμός ροής μάζας του νερού που περνάει μέσα από τον συμπυκνωτή, της αντλίας θερμότητας, καθορίζει την ηλεκτρική κατανάλωση της αντλίας με στόχο την θέρμανση του νερού σε συγκεκριμένη θερμοκρασία. Η λειτουργία της αντίστοιχης αντλίας διέπεται από έναν ελεγκτή PID, βλέπε Εικόνα 5. Το σημείο ρύθμισης για αυτόν τον ελεγκτή PID καθορίζεται από τη μηχανή κατάσταση, της οποίας ο ορισμός ουσιαστικά υλοποιεί έναν ελεγκτή υστέρησης για τη φόρτιση της δεξαμενής. Κατά αυτόν τον τρόπο επιτυγχάνεται ο έλεγχος κατανάλωσης ηλεκτρικής ισχύος ενεργοποιώντας την υδραυλική αντλία ανάλογα με τα επίπεδα θερμοκρασίας στη δεξαμενή αποθήκευσης και του αιτήματος κατανάλωσης του ελεγκτή τάσης.

Ένας PID ελεγκτής είναι ένας τύπος συστήματος ελέγχου που χρησιμοποιείται ευρέως στη βιομηχανία και σε άλλες εφαρμογές για την επίτευξη ακριβούς και σταθερού ελέγχου διαφόρων διεργασιών. Το PID προέρχεται από τα αρχικά των λέξεων Proportional (Αναλογικός), Integral (Ολοκληρωτικός) και Derivative (Παραγωγή), τα οποία αναφέρονται στα τρία βασικά συστατικά που τον απαρτίζουν. Ο ελεγκτής συνδυάζει και τις τρεις συνιστώσες για να διατηρήσει την ελεγχόμενη μεταβλητή όσο το δυνατόν πιο κοντά στην επιθυμητή τιμή. Ο ελεγκτής λαμβάνει συνεχώς την τιμή του σφάλματος (τη διαφορά μεταξύ της πραγματικής και της επιθυμητής τιμής), υπολογίζει την αντίδραση βάσει των P, I, D παραμέτρων και ρυθμίζει τη διαδικασία. [18]



Εικόνα 5: Ρύθμιση ελέγχου PID για τη ρύθμιση της κατανάλωσης ισχύος της αντλίας θερμότητας[17]

Ο ελεγκτής θερμότητας προσφέρει έξι τρόπους λειτουργίας:

1ος: Το θερμικό δίκτυο προσφέρει μόνο

2ος: Το θερμικό δίκτυο προσφέρει και η αντλία θερμότητας γεμίζει την δεξαμενή

3ος : Η δεξαμενή ζεστού νερού προσφέρει μόνο

4ος : Η δεξαμενή ζεστού νερού προσφέρει με ενεργή την αντλία θερμότητας

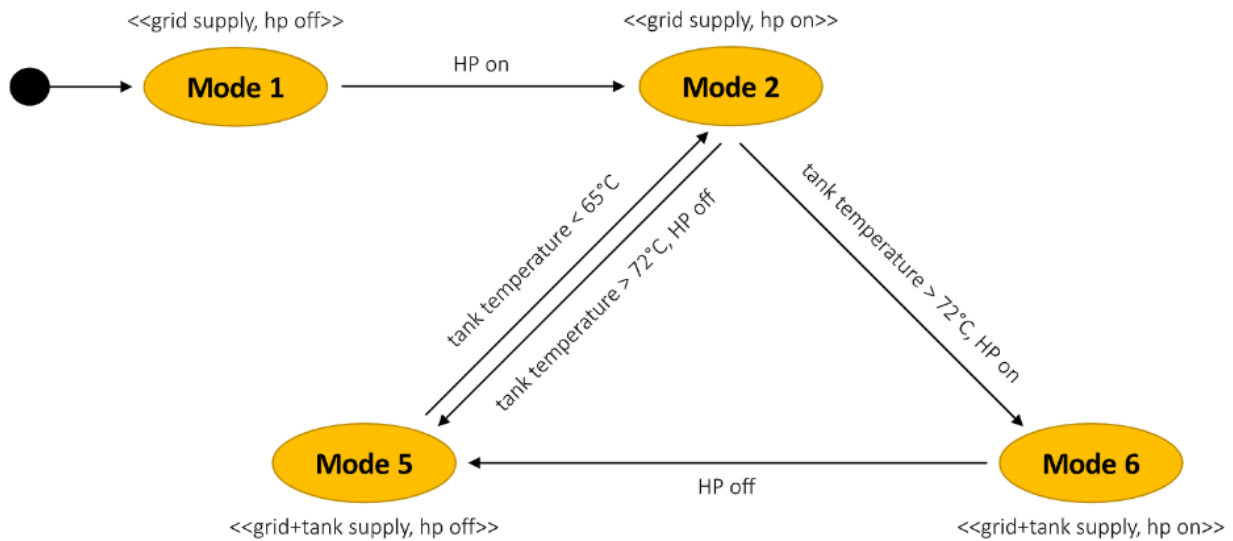
5ος: Το θερμικό δίκτυο προσφέρει και βοηθάει και η δεξαμενή θερμότητας

6ος : Το θερμικό δίκτυο προσφέρει με την υποστήριξη της δεξαμενής και ενεργή την αντλία θερμότητας

Στις λειτουργίες 1,3,5 η αντλία του 2ου βρόγχου είναι απενεργοποιημένη άσχετα την ρύθμιση του ελεγκτή τάση και η έξοδος του PID ελεγκτή είναι ρυθμισμένη στο μηδέν με αποτέλεσμα μηδενική ροή μάζας. Στις λειτουργίες 2,4,6 το σημείο ρύθμισης της κατανάλωσης ισχύος του ελεγκτή τάσης χρησιμοποιείται ως σημείο ρύθμισης για τον ελεγκτή PID για τον έλεγχο της 1ου βρόγχου. Υπάρχουν όμως κάποιοι περιορισμοί λόγο της περιορισμένης χωρητικότητας της αντλίας θερμότητας και της δεξαμενής, ο ελεγκτής τάσης δεν λαμβάνει υπόψη τον 3ο και 4ο τρόπο καθώς



το θερμικό σύστημα έμπαινε σε λειτουργία νησιδοποίησης όπως φαίνεται στην παρακάτω Εικόνα 6.



Εικόνα 6: Τρόποι λειτουργίας του ελεγκτή θερμότητας[17]

Το σύστημα και οι ελεγκτές έχουν διατηρηθεί όσο το δυνατόν πιο απλοί, ώστε η εστίαση να είναι στην εφαρμογή χωρίς να αυξηθεί άσκοπα την πολυπλοκότητα. Παρ' όλα αυτά, τα θερμικά και ηλεκτρικά συστήματα είναι στενά συνδεδεμένα, με το ένα να επηρεάζει το άλλο και αντίστροφα.

- Ο ελεγκτής τάσης καθορίζει το σημείο ρύθμισης της κατανάλωσης ισχύος της αντλίας θερμότητας,
- Το σημείο ρύθμισης της κατανάλωσης ισχύος της αντλίας θερμότητας επηρεάζει το ρυθμό φόρτισης της δεξαμενής,
- Η θερμική ζήτηση σε συνδυασμό με τον ρυθμό φόρτισης/εκφόρτισης της δεξαμενής καθορίζει τη θερμοκρασία της δεξαμενής,
- Η εκφόρτιση της δεξαμενής επηρεάζει τη θερμοκρασία της γραμμής επιστροφής και στη συνέχεια την απόδοση της αντλίας θερμότητας (καθώς η γραμμή επιστροφής χρησιμεύει ως πηγή για την αντλία θερμότητας),
- Η θερμοκρασία της δεξαμενής και η απόδοση της αντλίας θερμότητας καθορίζουν την ενεργοποίηση της αντλίας θερμότητας από τον ελεγκτή θερμότητας, και
- Η πραγματική κατανάλωση ισχύος της αντλίας θερμότητας (η οποία μπορεί να είναι μικρότερη από το σημείο ρύθμισης) επηρεάζει άμεσα τις αποφάσεις του ελεγκτή τάσης.

Αυτή η αλληλεξάρτηση μεταξύ των δύο υποσυστημάτων, της οποίας η δυναμική μπορεί να κατανοηθεί πλήρως μόνο μέσω της ταυτόχρονης αξιολόγησης και των δύο. Ως εκ τούτου, αυτή η φαινομενικά απλή διάταξη λειτουργεί ως εξαιρετικό σημείο αναφοράς για εφαρμογές πολυενεργειακών συστημάτων.

# Κεφάλαιο 4: Αναλυτική περιγραφή εργαλείου

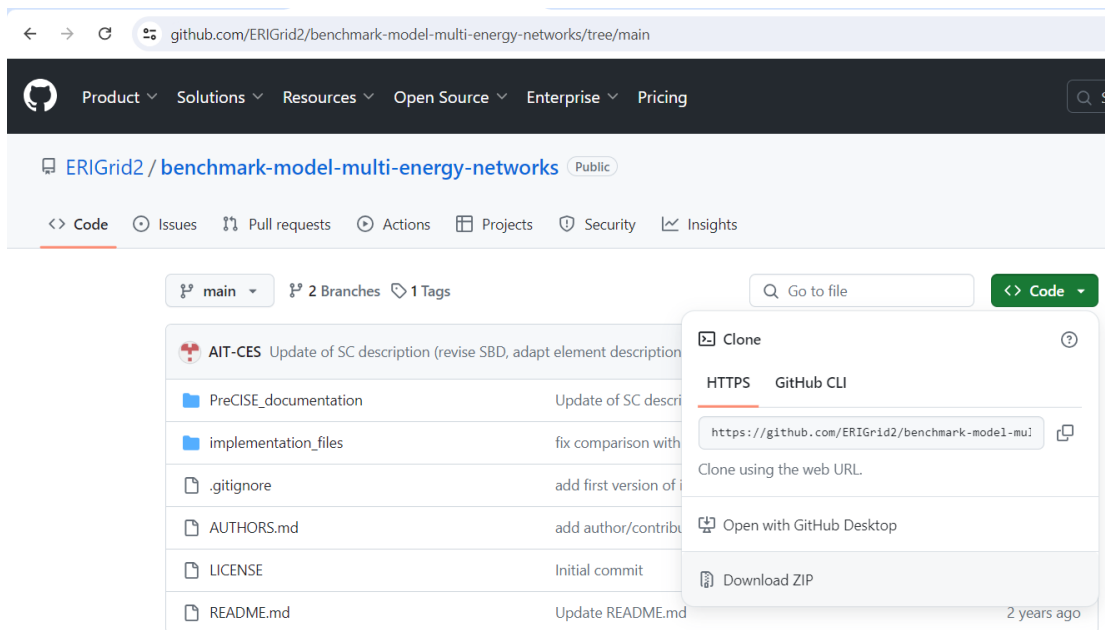
---

Το Anaconda Navigator είναι ένα γραφικό περιβάλλον διαχείρισης για την πλατφόρμα Anaconda, η οποία παρέχει εργαλεία για την επιστήμη δεδομένων και την ανάλυση μεγάλων δεδομένων [19]. Μέσω του Navigator, οι χρήστες μπορούν εύκολα να διαχειριστούν περιβάλλοντα, να εγκαταστήσουν βιβλιοθήκες και να εκτελέσουν εφαρμογές όπως το Spyder, ένα δημοφιλές ολοκληρωμένο περιβάλλον ανάπτυξης για Python [20]. Οι βιβλιοθήκες randaPOWER [21] και randaPipes [22] είναι ειδικές βιβλιοθήκες της Python, σχεδιασμένες για την ανάλυση και προσομοίωση ενεργειακών συστημάτων και δικτύων. Το randaPOWER επικεντρώνεται στα ηλεκτρικά δίκτυα και προσφέρει εργαλεία για τη μοντελοποίηση, ανάλυση και βελτιστοποίηση ηλεκτρικών συστημάτων, ενώ το randaPipes είναι πιο κατάλληλο για την ανάλυση ροής σε δίκτυα σωληνώσεων, όπως δίκτυα φυσικού αερίου ή θερμικών δικτύων. Και οι δύο βιβλιοθήκες είναι ισχυρά εργαλεία που χρησιμοποιούνται για τη μοντελοποίηση πολύπλοκων ενεργειακών συστημάτων, παρέχοντας εύκολες στη χρήση μεθόδους ανάλυσης και προσομοίωσης.

## 4.1 Εγκατάσταση

### 4.1.1 Κατέβασμα αρχείου

Αρχικά, θα κατεβάσουμε το αρχείο ZIP του EriGrid, το οποίο βρίσκεται στον σύνδεσμο [17]. Για την λήψη, επιλέγουμε το "<>Code" και στη συνέχεια επιλέγουμε "Download ZIP"(Εικόνα 7).



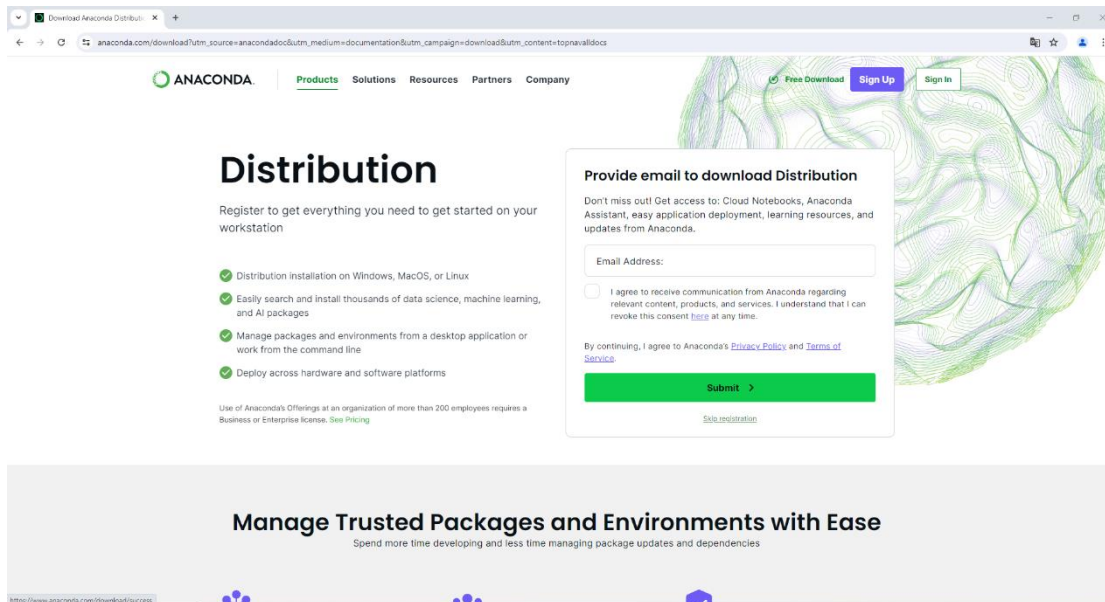
Εικόνα 7: Διαδρομή εύρεσης του zip αρχείου στην ιστοσελίδα του GitHub

Ο φάκελος περιέχει μια αναλυτική περιγραφή για το εργαλείο και τον κώδικα που θα χρησιμοποιήσουμε. Ανοίγοντας τον φάκελο `implementation files`, επιλέγουμε τον φάκελο `cosim_randaripes_randarower` και τον εξαγάγουμε. Αυτό υποδεικνύει ότι η υλοποίηση του κώδικα βασίζεται αποκλειστικά σε δύο βιβλιοθήκες της Python, χωρίς να απαιτείται η εγκατάσταση επιπλέον εφαρμογών, πέρα από το Anaconda Navigator.

Η επιλογή του συγκεκριμένου φακέλου έγινε διότι, παρότι ο άλλος φάκελος υλοποιεί την ίδια λειτουργία χρησιμοποιώντας τη βιβλιοθήκη `randarower` σε συνδυασμό με τη `Modelica DisHeatLib`, η τελευταία προϋποθέτει την εγκατάσταση του εργαλείου `Dymola` και ενός C++ compiler. Η βιβλιοθήκη `Modelica DisHeatLib` είναι ανοικτού κώδικα και περιλαμβάνει βασικά μοντέλα για δίκτυα τηλεθέρμανσης και συστήματα ελέγχου, προσφέροντας διασύνδεση με προαιρετικό δίκτυο ηλεκτρικής ενέργειας. Ωστόσο, η εκτέλεση του κώδικα μέσω αυτής απαιτεί επιπλέον εργαλεία όπως το `Dymola`, καθώς και άδεια χρήσης για την εξαγωγή αποτελεσμάτων σε μορφή `Functional Mock-up Unit (FMU)`. Επιπλέον, λειτουργεί μόνο σε περιβάλλον Windows.

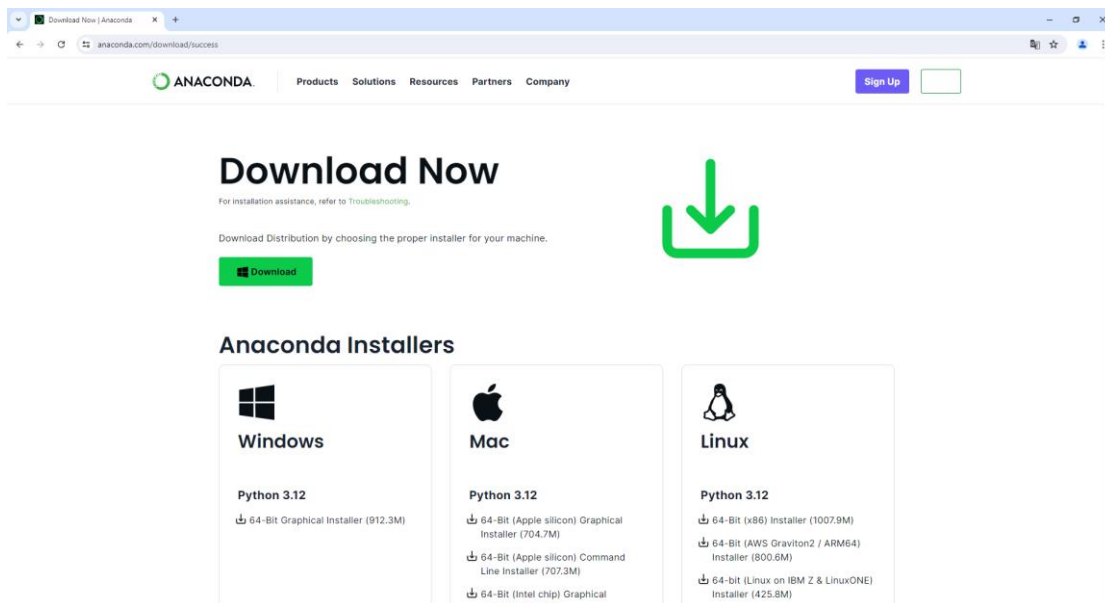
#### 4.1.2 Anaconda Navigator

Στη συνέχεια, θα προχωρήσουμε στη λήψη του Anaconda Navigator, ενός χρήσιμου εργαλείου που είναι φιλικό προς τους νέους χρήστες και επιτρέπει την εκτέλεση πολλαπλών έργων με διαφορετικές εκδόσεις της Python και των αντίστοιχων βιβλιοθηκών-πακέτων. Για να το κατεβάσουμε, μεταβαίνουμε στον σύνδεσμο [23] και επιλέγουμε `Skip registration` που βρίσκεται κάτω από την επιλογή `Submit`. (Εικόνα 8)



Εικόνα 8: Λήψη του εργαλείου Anaconda Navigator

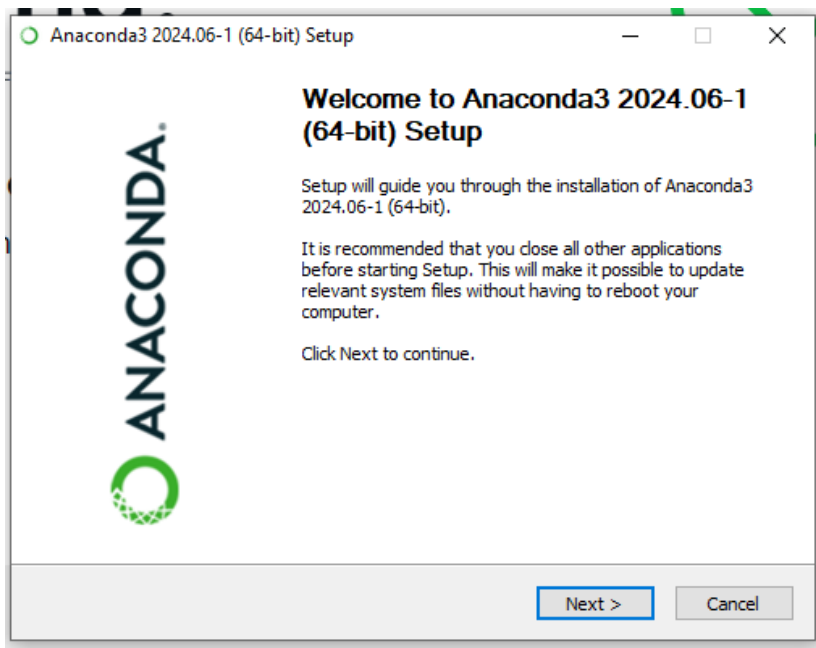
Στην συνέχεια επιλέγουμε ανάλογα το λειτουργικό σύστημα του υπολογιστή μας το κατάλληλο. (Εικόνα 9)



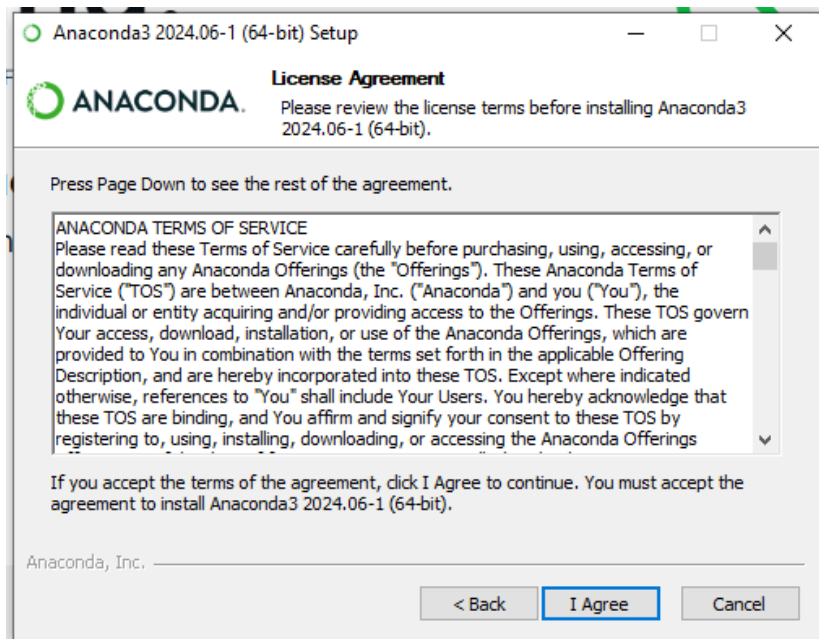
Εικόνα 9: Επιλογές αρχείου διάφορων λογισμικών

Μετά την ολοκλήρωση της λήψης, εκτελούμε το αρχείο .exe για να ξεκινήσει η διαδικασία εγκατάστασης. Η εγκατάσταση θα προχωρήσει ακολουθώντας τις οδηγίες που παρέχονται στις εικόνες παρακάτω.

Ξεκινώντας την εγκατάσταση επιλέγουμε Next και I Agree. (Εικόνα 10) και (Εικόνα 11)

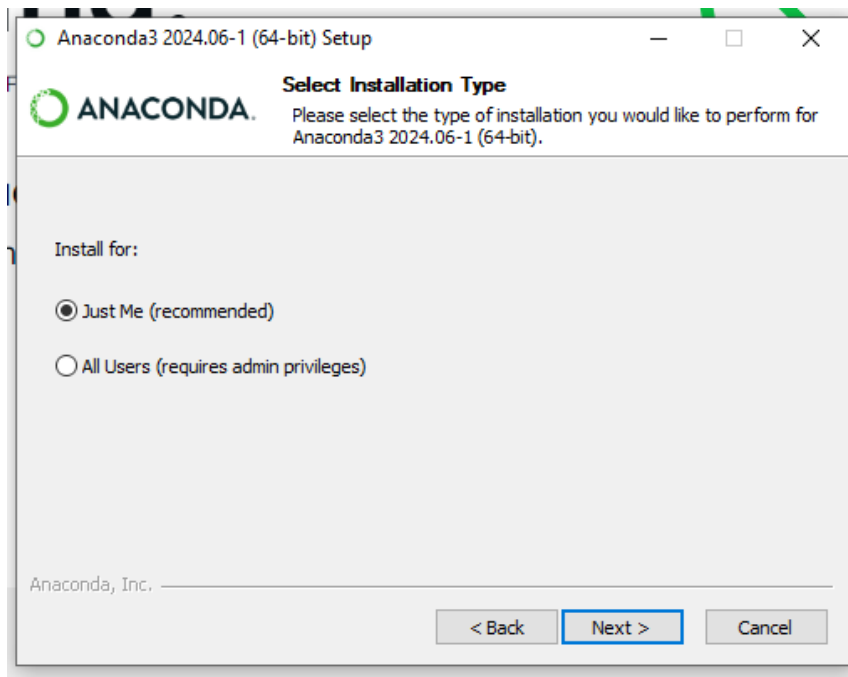


Εικόνα 10: Εγκατάσταση βήμα 1



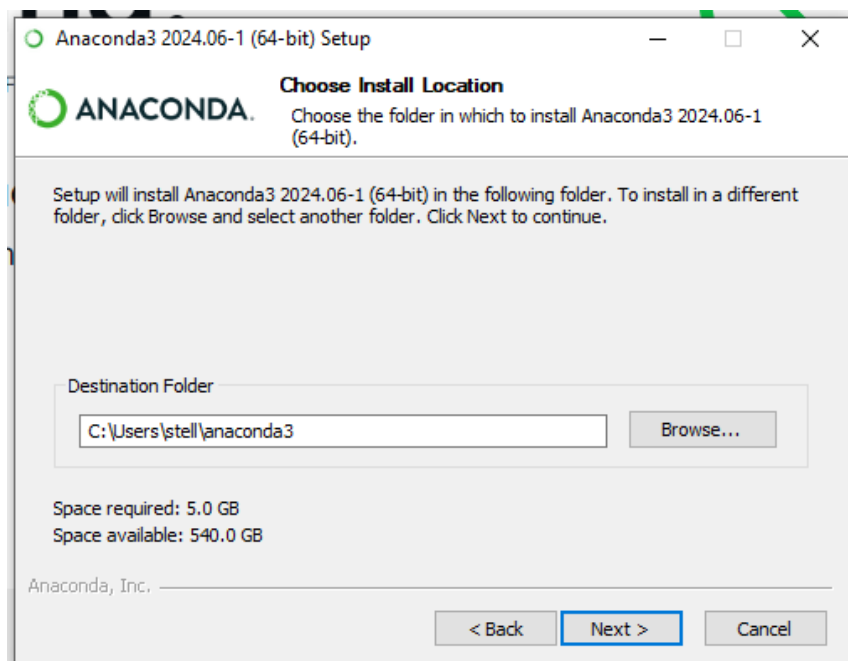
Εικόνα 11: Εγκατάσταση βήμα 2

Έπειτα στο επόμενο βήμα είναι δικιά σας επιλογή άμα θέτε να κατεβάσετε το Anaconda Navigator σε όλους τους χρήστες του υπολογιστή σας ή μόνο τοπικά σε αυτόν που χρησιμοποιείτε. (Εικόνα 12)



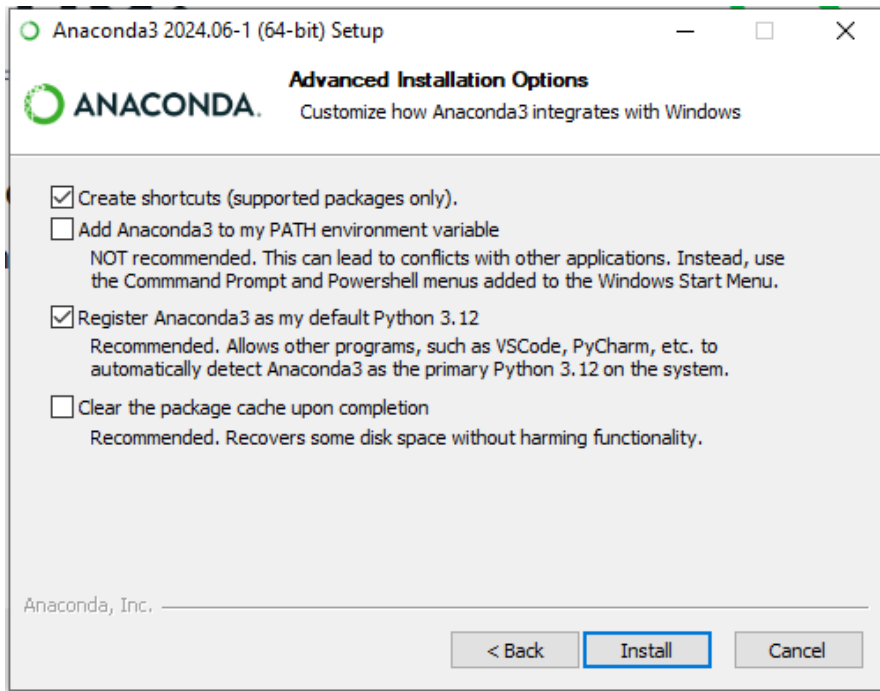
Εικόνα 12: Εγκατάσταση βήμα 3

Μετά την επιλογή της εντολής Next, δίνεται η δυνατότητα επιλογής του φακέλου εγκατάστασης. Ωστόσο, προτιμότερο θα ήταν να διατηρηθεί η προεπιλεγμένη τοποθεσία εγκατάστασης. (Εικόνα 13)



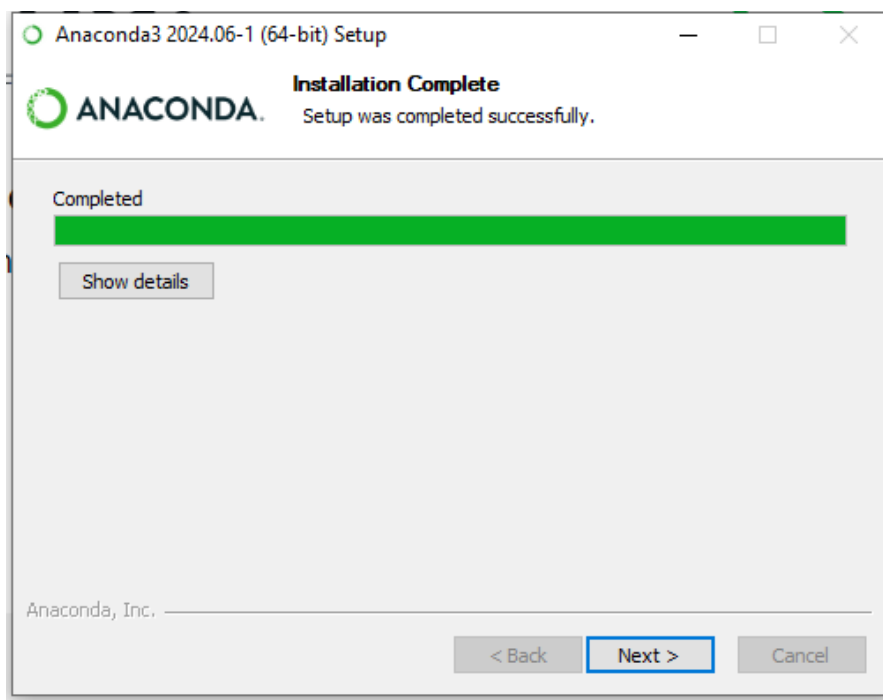
Εικόνα 13: Εγκατάσταση βήμα 4

Στην συνέχεια επιλέγουμε το πρώτο και τρίτο κουτάκι και Install. (Εικόνα 14)

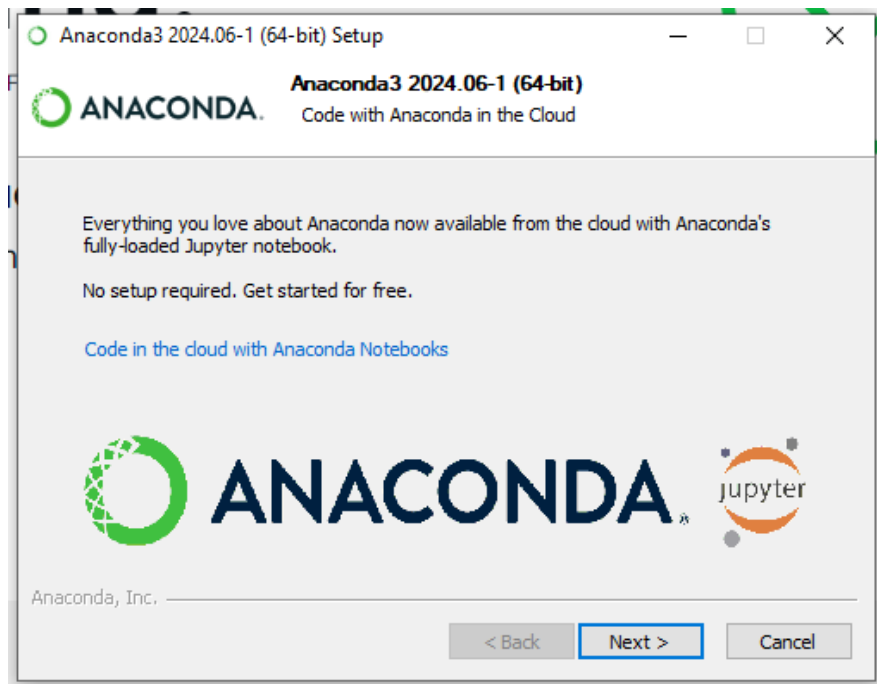


Εικόνα 14: Εγκατάσταση βήμα 5

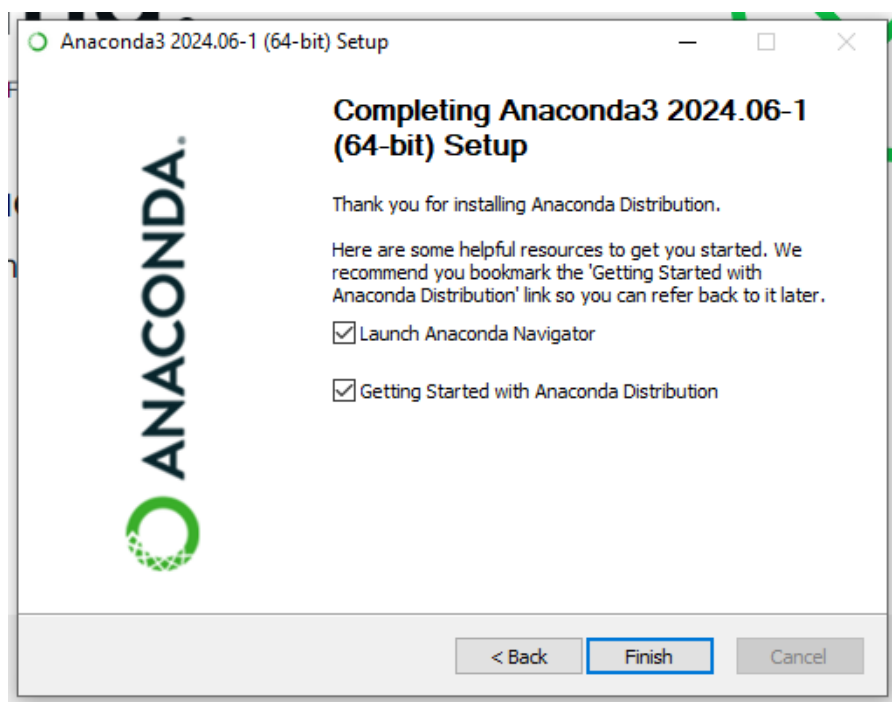
Έπειτα προχωράμε σύμφωνα με την (Εικόνα 15) και (Εικόνα 16) .



Εικόνα 15: Εγκατάσταση βήμα 6



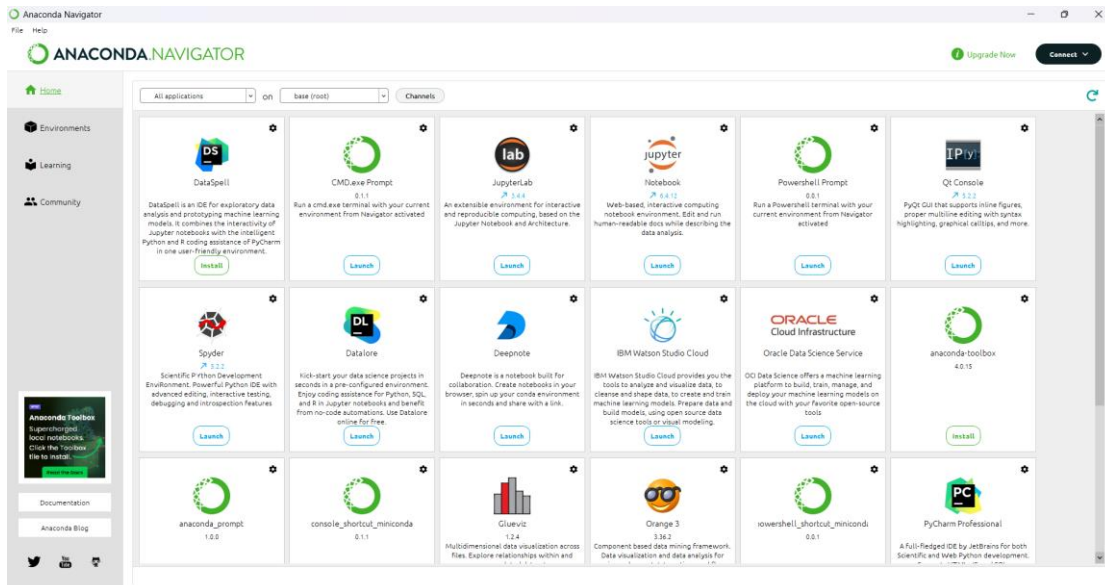
Εικόνα 16: Εγκατάσταση βήμα 7



Εικόνα 17: Εγκατάσταση βήμα 8

Με την επιλογή της εντολής Finish (Εικόνα 17), η εγκατάσταση θα ολοκληρωθεί και η εφαρμογή θα ανοίξει αυτόματα. (Εικόνα 18) Στη συνέχεια, θα προχωρήσουμε στο επόμενο βήμα της διαδικασίας.



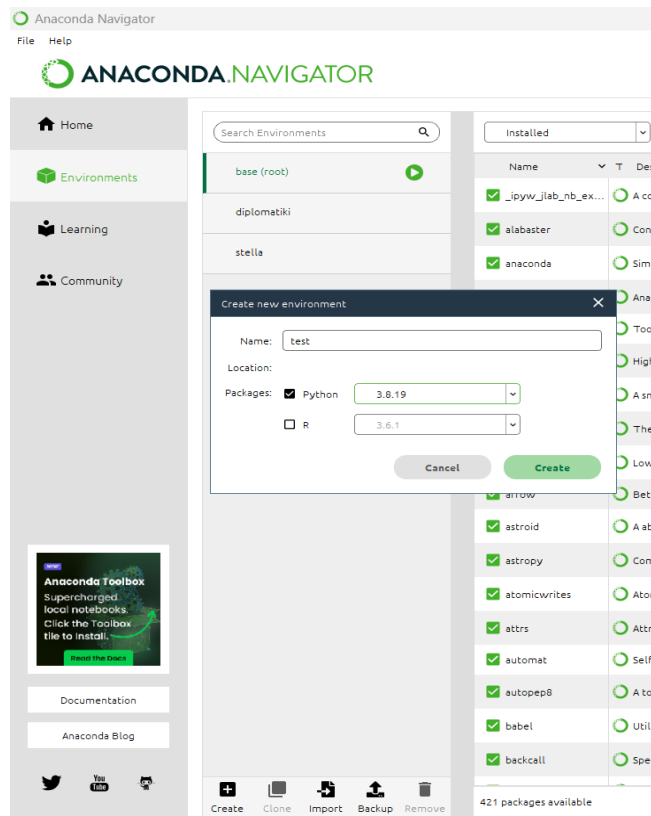


Εικόνα 18: Αρχική οθόνη της εφαρμογής

### 4.1.3 Δημιουργία καινούργιου περιβάλλοντος και εγκατάσταση της Spyder

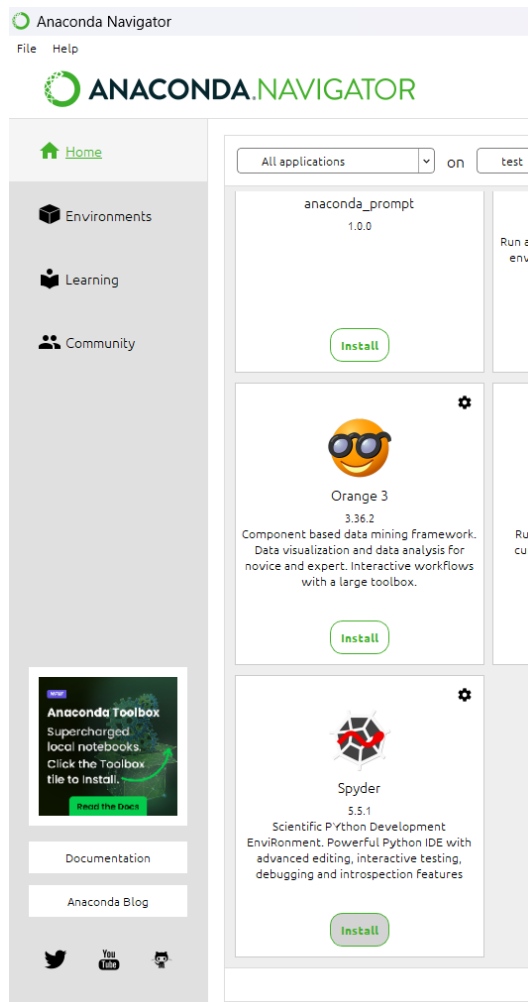
Έπειτα θα δημιουργήσουμε ένα καινούργιο περιβάλλον όπου εκεί θα εγκαταστήσουμε συγκεκριμένες εκδοχές από βιβλιοθήκες που απαιτούνται για την εκτέλεση της προσομοίωσης.

Πηγαίνοντας αριστερά επιλέγουμε το Environments από το αριστερό μενού. Στην κάτω αριστερή γωνία, κάνουμε κλικ στο Create, ονομάζουμε το περιβάλλον όπως επιθυμούμε και επιλέγουμε το πακέτο Python, την έκδοση 3.8.19. Τέλος, επιλέγουμε Create για να ολοκληρώσουμε τη διαδικασία δημιουργίας του περιβάλλοντος. (Εικόνα 19)



Εικόνα 19: Δημιουργία καινούργιου περιβάλλοντος

Στην συνέχεια έχοντας επιλέξει το περιβάλλον που μόλις δημιουργήσαμε πηγαίνουμε, από το αριστερό μενού, στο Home για να εγκαταστήσουμε την εφαρμογή που θα χρειαστούμε για την εκτέλεση του παράδειγμα μας. Κάτω αριστερά βρίσκεται η Spyder και κάνουμε install. (Εικόνα 20) Επιλέγουμε την Spyder καθώς είναι ένα εξαιρετικό εργαλείο για όσους εργάζονται με την Ρυθση σε επιστημονικούς υπολογισμούς και ανάλυση δεδομένων, προσφέροντας μια ισχυρή και εύχρηστη πλατφόρμα ανάπτυξης.

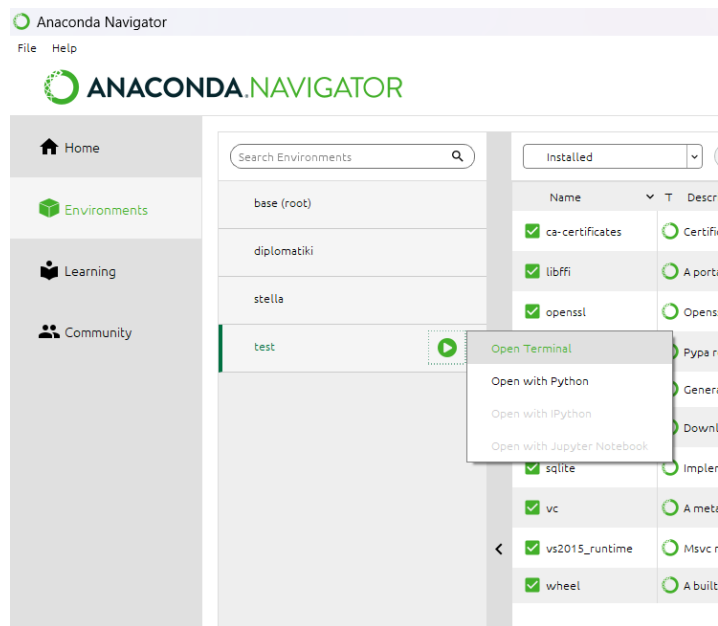


Εικόνα 20: Εγκατάσταση της εφαρμογής Spyder

#### 4.1.4 Εγκατάσταση συγκεκριμένων εκδόσεων βιβλιοθηκών

Έπειτα θα προχωρήσουμε στην εγκατάσταση των συγκεκριμένων εκδόσεων των βιβλιοθηκών που απαιτούνται.

Επιλέγουμε το καινούργιο environment που έχουμε δημιουργήσει και περιμένουμε λίγα δευτερόλεπτα να φορτώσει. Πατάμε το κουμπί που βρίσκεται στα δεξιά του και επιλέγουμε Open Terminal. (Εικόνα 21)



Εικόνα 21: Άνοιγμα του τερματικού

Χρησιμοποιώντας τις εντολές «cd» και «dir» οι οποίες προσπελάσουν τους υπάρχοντες φακέλους και παρουσιάζουν τα περιεχόμενα αυτών (Εικόνα 22), αντίστοιχα ανοίγουμε τον με φάκελο «cosim\_pandapipes\_pandapower» που περιέχει το αρχείο «requirements.txt». Το αρχείο «requirements.txt» περιλαμβάνει τις βιβλιοθήκες που απαιτούνται για την εκτέλεση του προτεινόμενου προγράμματος MES της EriGrid.

```

C:\WINDOWS\system32\cmd. x + v
(test) C:\Users\stell>cd desktop
(test) C:\Users\stell\Desktop>cd cosim_pandapipes_pandapower
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>pip install -r requirements.txt

```

Εικόνα 22: Εγκατάσταση συγκεκριμένων βιβλιοθηκών βήμα 1

Για την επίτευξη αυτού εκτελείται η εντολή `pip install -r requirements.txt` και στο αρχείο «README» του ίδιου φακέλου. Η εκτέλεση αυτής της εντολής θα ξεκινήσει τη διαδικασία εγκατάστασης των απαιτούμενων βιβλιοθηκών στις συγκεκριμένες εκδόσεις τους, οι οποίες περιγράφονται στο αρχείο «requirements.txt». Η διαδικασία εγκατάστασης ολοκληρώνεται όταν επανεμφανιστεί το command line με την προηγούμενη κατεύθυνση που είχαμε ορίσει στην Εικόνα 23.

```

C:\WINDOWS\system32\cmd. x + v
Using cached matplotlib-3.7.5-cp38-cp38-win_amd64.whl (7.5 MB)
Using cached contourpy-1.1.1-cp38-cp38-win_amd64.whl (477 kB)
Using cached cyclers-0.12.1-py3-none-any.whl (8.3 kB)
Using cached decorator-4.4.2-py2.py3-none-any.whl (9.2 kB)
Downloading fonttools-4.53.0-cp38-cp38-win_amd64.whl (1.5 MB)
1.5/1.5 MB 2.3 MB/s eta 0:00:00
Downloading importlib_resources-6.4.0-py3-none-any.whl (38 kB)
Using cached kiwisolver-1.4.5-cp38-cp38-win_amd64.whl (56 kB)
Using cached llvmlite-0.39.1-cp38-cp38-win_amd64.whl (23.2 MB)
Using cached numexpr-2.8.6-cp38-cp38-win_amd64.whl (94 kB)
Downloading packaging-24.1-py3-none-any.whl (53 kB)
54.0/54.0 kB 99.9 kB/s eta 0:00:00
Downloading pillow-10.4.0-cp38-cp38-win_amd64.whl (2.6 MB)
2.6/2.6 MB 1.6 MB/s eta 0:00:00
Using cached pyparsing-3.1.2-py3-none-any.whl (103 kB)
Using cached python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Using cached pytz-2024.1-py2.py3-none-any.whl (505 kB)
Downloading importlib_metadata-8.0.0-py3-none-any.whl (24 kB)
Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Downloading zipp-3.19.2-py3-none-any.whl (9.0 kB)
Installing collected packages: simpy, simple-pid, pytz, docopt, dataclasses, zipp, six, simpy.io, pyparsing, pillow, packaging, numpy, llvmlite, kiwisolver, fonttools, decorator, cyclers, scipy, python-dateutil, numexpr, networkx, mosaik-api, importlib-resources, importlib-metadata, contourpy, tables, pandas, numba, mosaik, matplotlib, pandapower, pandapipes
Successfully installed contourpy-1.1.1 cyclers-0.12.1 dataclasses-0.6 decorator-4.4.2 docopt-0.6.2 fonttools-4.53.0 importlib-metadata-8.0.0 importlib-resources-6.4.0 kiwisolver-1.4.5 llvmlite-0.39.1 matplotlib-3.7.5 mosaik-2.6.1 mosaik-api-2.4.2 networkx-2.5.1 numba-0.56.0 numexpr-2.8.6 numpy-1.22.4 packaging-24.1 pandapipes-0.6.0 pandapower-2.8.0 pandas-1.4.3 pillow-10.4.0 pyparsing-3.1.2 python-dateutil-2.9.0.post0 pytz-2024.1 scipy-1.8.1 simple-pid-1.0.1 simpy-3.0.13 simpy.io-0.2.3 six-1.16.0 tables-3.7.0 zipp-3.19.2
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>

```

Εικόνα 23: Εγκατάσταση συγκεκριμένων βιβλιοθηκών βήμα 2

Πριν συνεχίσουμε με την υλοποίηση του παράδειγμα που κατεβάσαμε προτείνεται να επανεκκινήσουμε το Anaconda Navigator

Για πρώτη φορά θα τρέξουμε την προσομοίωση στην Spyder. Πηγαίνουμε στο Home και πατάμε Launch την Spyder. Μόλις ανοίξει πατάμε File -> Open -> επιλέγουμε το αρχείο benchmark\_multi\_energy\_sim.py και πατάμε Run.

Άμα έχουμε κάνει σωστά τα βήματα θα τρέξει κανονικά και θα μοιάζει όπως η Εικόνα 24.

```

Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\stell\Desktop\cosim_pandapipes_pandapower\benchmark_multi_energy_sim.py
benchmark_multi_energy_sim.py
1 | Copyright (c) 2021 by ERI-Grid 2.0. All rights reserved.
2 | # Use of this source code is governed by a BSD-style license that can be found in the LICENSE file.
3 | ...
4 | This MOSAIK co-simulation setup implements the ERI-Grid 2.0 multi-energy benchmark.
5 | ...
6 |
7 | # Define default for simulation start time, step size and end (1 MOSAIK time-step = 1 second).
8 | START_TIME = '2019-02-01 00:00:00'
9 | STEP_SIZE = 60 * 1
10 | END = 7 * 24 * 60 * 60
11 |
12 | # Specify profiles for generation and demand.
13 | HEAT_DEMAND_LOAD_PROFILES = 'resources/heat/heat_demand_load_profiles.csv'
14 | POWER_DEMAND_LOAD_PROFILES = 'resources/power/power_demand_load_profiles.csv'
15 | PV_GENERATION_PROFILES = 'resources/power/pv_generation_profiles.csv'
16 |
17 | # MOSAIK simulator configuration.
18 | SIM_CONFIG = {
19 |     'DNNetworkSim': {
20 |         'python': 'simulators:DNNetworkSimulator'
21 |     },
22 |     'ENNetworkSim': {
23 |         'python': 'simulators:ElectricNetworkSimulator'
24 |     },
25 |     'StorageTankSim': {
26 |         'python': 'simulators:StratifiedWaterStorageTankSimulator'
27 |     },
28 |     'HeatPumpSim': {
29 |         'python': 'simulators:ConstantTcond#PSimulator'
30 |     },
31 |     'HeatExchangerSim': {
32 |         'python': 'simulators:HEXConsumerSimulator'
33 |     },
34 |     'FlexHeatCtrlSim': {
35 |         'python': 'simulators:SimpleFlexHeatControllersSimulator'
36 |     },
37 |     'VoltageCtrlSim': {
38 |         'python': 'simulators:VoltageControlSimulator'
39 |     },
40 |     'TimeSeriesSim': {

```

```

Console I/O X
Starting "TimeSeriesSim" as "TimeSeriesSim-1" ...
Starting "StorageTankSim" as "StorageTankSim-0" ...
Starting "HeatPumpSim" as "HeatPumpSim-0" ...
Starting "FlexHeatCtrlSim" as "FlexHeatCtrlSim-0" ...
Starting "VoltageCtrlSim" as "VoltageCtrlSim-0" ...
Starting "CollectorSim" as "CollectorSim-0" ...
Starting simulation.
Controller state changed from 1 to 2
0.11%progress: 0.14%progress: 0.16%progress: 0.20%
Progress: 1.68%progress: 0.22%progress: 0.25%progress: 0.29%progress:
0.34%progress: 0.38%progress: 0.43%progress: 0.45%progress:
0.46%progress: 0.54%progress: 0.58%progress: 0.61%progress:
0.65%progress: 0.69%progress: 0.74%progress: 0.78%progress:
0.83%progress: 0.87%progress: 0.91%progress: 0.93%progress:
0.98%progress: 1.04%progress: 1.07%progress: 1.12%progress:
1.16%progress: 1.21%progress: 1.25%progress: 1.28%progress:
1.32%progress: 1.36%progress: 1.40%progress: 1.45%progress:
1.50%progress: 1.53%progress: 1.57%progress: 1.61%progress: 1.64%
Python Console History
conda: test (Python 3.8.10) | Completion: conda | LSP: Python | Line 1, Col 1 | ASCII | LF | RW | Mem 82%

```

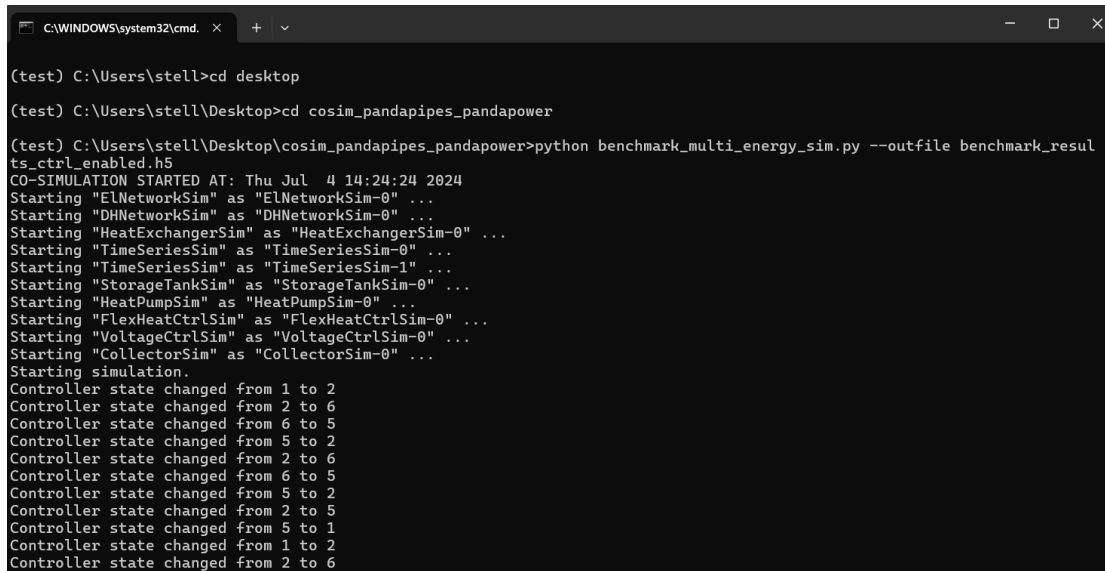
Εικόνα 24: Άνοιγμα της Εφαρμογής Spyder

Παρατηρώντας στις πρώτες γραμμές του κώδικα μπορούμε να δούμε με τι βήμα τρέχει. Το STEP\_SIZE = 60\*1 και END = 7\*24\*60\*60 σημαίνει ότι η προσομοίωση εκτελείται με βήμα ενός λεπτού για διάρκεια επτά ημερών. Σε αυτό το σημείο μπορούμε να αλλάξουμε το βήμα προσομοίωσης που επιθυμούμε.

Για τον παραπάνω λόγο και ανάλογα την δύναμη του υπολογιστή μας η ώρα που θα ολοκληρωθεί η προσομοίωση ποικίλει.

## 4.2 Εκτέλεση κώδικα

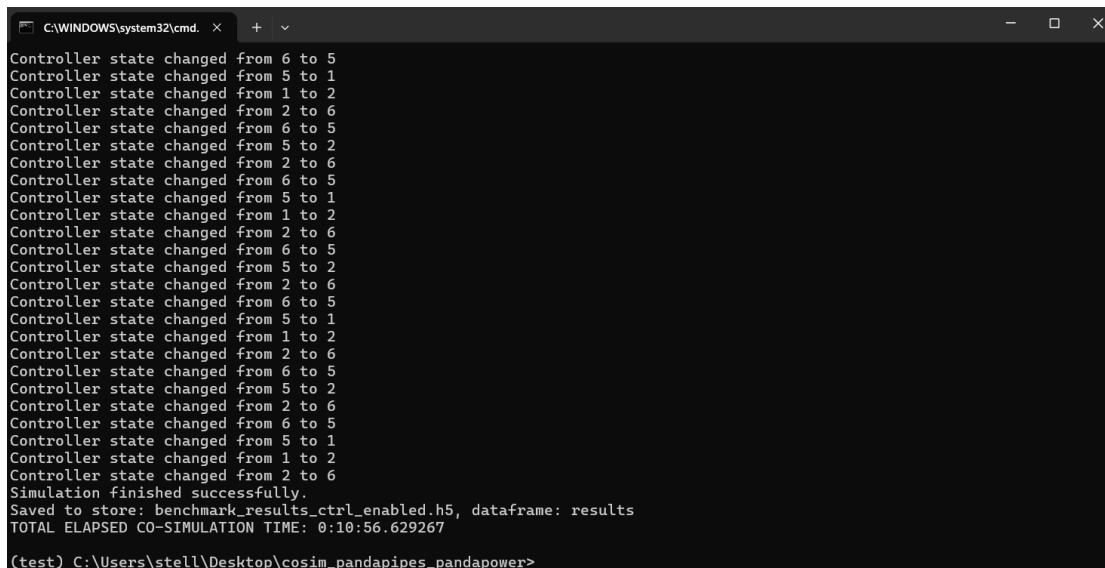
Πηγαίνουμε ξανά στο terminal γράφουμε με τις εντολές που δείξαμε παραπάνω, φτάνουμε στο φάκελο στον οποίο βρίσκεται το αρχείο `benchmark_multi_energy_sim.py` και γράφουμε στο terminal `python benchmark_multi_energy_sim.py --outfile benchmark_results_ctrl_enabled.h5` η οποία είναι η πρώτη προσομοίωση του παραδείγματος με ενεργοποιημένο τον ελεγκτή τάσης (voltage controller) όπως φαίνεται στην Εικόνα 25.



```
(test) C:\WINDOWS\system32\cmd. x + v
(test) C:\Users\stell>cd desktop
(test) C:\Users\stell\Desktop>cd cosim_pandapipes_pandapower
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>python benchmark_multi_energy_sim.py --outfile benchmark_results_ctrl_enabled.h5
CO-SIMULATION STARTED AT: Thu Jul 4 14:24:24 2024
Starting "ELNetworkSim" as "ELNetworkSim-0" ...
Starting "DHNetworkSim" as "DHNetworkSim-0" ...
Starting "HeatExchangerSim" as "HeatExchangerSim-0" ...
Starting "TimeSeriesSim" as "TimeSeriesSim-0" ...
Starting "TimeSeriesSim" as "TimeSeriesSim-1" ...
Starting "StorageTankSim" as "StorageTankSim-0" ...
Starting "HeatPumpSim" as "HeatPumpSim-0" ...
Starting "FlexHeatCtrlSim" as "FlexHeatCtrlSim-0" ...
Starting "VoltageCtrlSim" as "VoltageCtrlSim-0" ...
Starting "CollectorSim" as "CollectorSim-0" ...
Starting simulation.
Controller state changed from 1 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 1
Controller state changed from 1 to 2
Controller state changed from 2 to 6
```

Εικόνα 25: Αρχή εκτέλεσης κώδικα με τον ελεγκτή τάσης ενεργοποιημένο

Η προσομοίωση ξεκινά και στο τέλος της διαδικασίας, εμφανίζεται ο συνολικός χρόνος που διήρκεσε η προσομοίωση. (Εικόνα 26)



```
Controller state changed from 6 to 5
Controller state changed from 5 to 1
Controller state changed from 1 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 1
Controller state changed from 1 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 1
Controller state changed from 1 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 6
Controller state changed from 6 to 5
Controller state changed from 5 to 1
Controller state changed from 1 to 2
Controller state changed from 2 to 6
Simulation finished successfully.
Saved to store: benchmark_results_ctrl_enabled.h5, dataframe: results
TOTAL ELAPSED CO-SIMULATION TIME: 0:10:56.629267
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>
```

Εικόνα 26: Τέλος εκτέλεσης κώδικα με τον ελεγκτή τάσης ενεργοποιημένο

Συνεχίζοντας γράφουμε στο terminal `python benchmark_multi_energy_sim.py --outfile benchmark_results_ctrl_disabled.h5 --voltage-control-disable` η οποία είναι η δεύτερη προσομοίωση του παραδείγματος με απενεργοποιημένο τον ελεγκτή τάσης (voltage controller) όπως παρουσιάζεται στην Εικόνα 27 και Εικόνα 28.

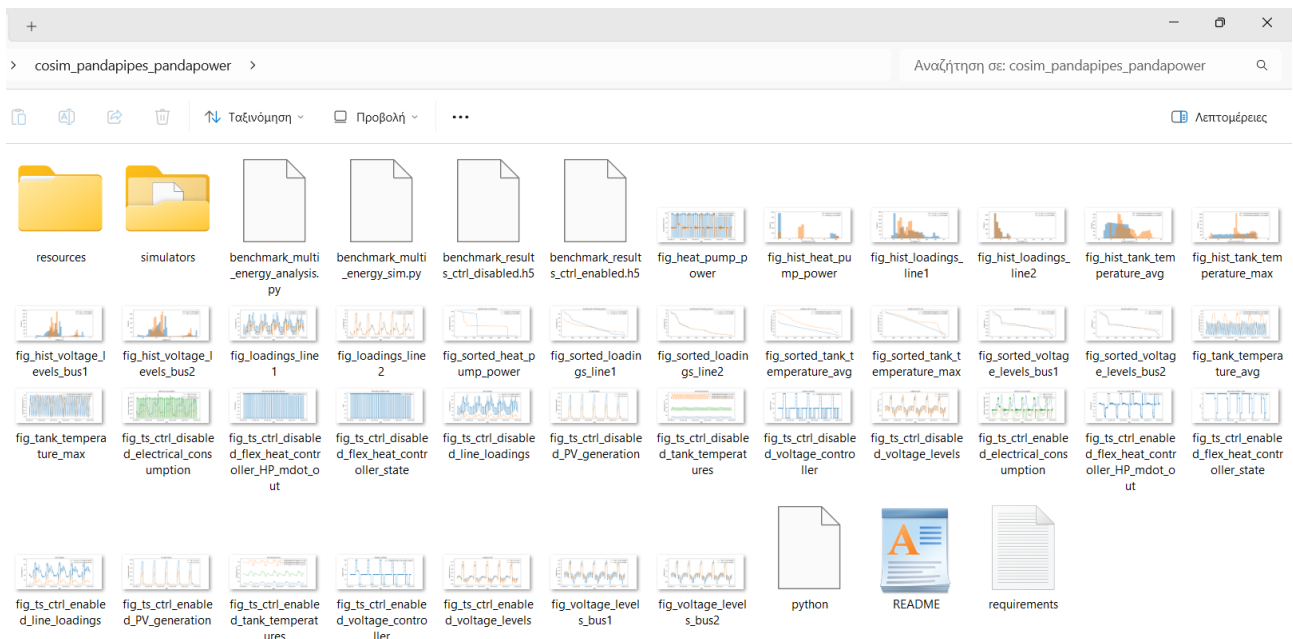
```
C:\WINDOWS\system32\cmd. x + v
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>python benchmark_multi_energy_sim.py --outfile benchmark_resul
ts_ctrl_disabled.h5 --voltage-control-disabled
CO-SIMULATION STARTED AT: Thu Jul 4 14:51:26 2024
Starting "ELNetworkSim" as "ELNetworkSim-0" ...
Starting "DHNetworkSim" as "DHNetworkSim-0" ...
Starting "HeatExchangerSim" as "HeatExchangerSim-0" ...
Starting "TimeSeriesSim" as "TimeSeriesSim-0" ...
Starting "TimeSeriesSim" as "TimeSeriesSim-1" ...
Starting "StorageTankSim" as "StorageTankSim-0" ...
Starting "HeatPumpSim" as "HeatPumpSim-0" ...
Starting "FlexHeatCtrlSim" as "FlexHeatCtrlSim-0" ...
Starting "VoltageCtrlSim" as "VoltageCtrlSim-0" ...
Starting "CollectorSim" as "CollectorSim-0" ...
Controller state changed from 1 to 2
Starting simulation.
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
```

Εικόνα 27: Αρχή εκτέλεσης κώδικα με τον ελεγκτή τάσης απενεργοποιημένο

```
C:\WINDOWS\system32\cmd. x + v
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Controller state changed from 2 to 5
Controller state changed from 5 to 2
Simulation finished successfully.
Saved to store: benchmark_results_ctrl_disabled.h5, dataframe: results
TOTAL ELAPSED CO-SIMULATION TIME: 0:10:58.586166
(test) C:\Users\stell\Desktop\cosim_pandapipes_pandapower>
```

Εικόνα 28: Τέλος εκτέλεσης κώδικα με τον ελεγκτή τάσης απενεργοποιημένο

Τέλος γράφουμε πάλι στο terminal `python benchmark_multi_energy_analysis.py` για να μας βγάλει τα γραφήματα από τις δύο προσημειώσεις. Δεν θα εκτυπώσει κάτι στο terminal όπως τα προηγούμενα αλλά τα γραφήματα θα αποθηκευτούν απευθείας στον φάκελο `cosim_pandapipes_pandapower`. (Εικόνα 29)



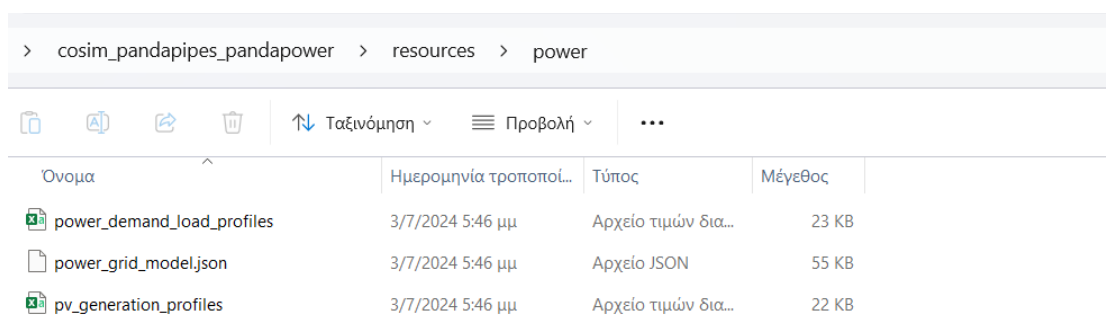
Εικόνα 29: Φάκελος `cosim_pandapipes_pandapower`

## 4.3 Τροποποίηση αρχείου κώδικα

Αφού βεβαιωθούμε ότι ο κώδικας εκτελείται σωστά και παράγει αποτελέσματα, θα προχωρήσουμε σε τροποποιήσεις του κώδικα ώστε να εκτελεστεί με τα επιθυμητά δεδομένα.

### 4.3.1 Ηλεκτρικό δίκτυο

Ξεκινώντας τις αλλαγές από το ηλεκτρικό δίκτυο στον φάκελο που βρισκόμαστε, `cosim_pandapipes_pandapower`, πηγαίνουμε στον υποφάκελο `resources` και στη συνέχεια στον φάκελο `power`. (Εικόνα 30)



Εικόνα 30: Διαδρομή υποφακέλου `power`

Ανοίγουμε το αρχείο `power_grid_model.json` με δεξί κλικ Άνοιγμα με -> WordPad για να μπορούμε να το επεξεργαστούμε.

Ένα αρχείο JSON (JavaScript Object Notation) είναι ένα ελαφρύ format ανταλλαγής δεδομένων που χρησιμοποιείται για την αποθήκευση και τη μεταφορά δεδομένων σε μια δομημένη μορφή. Είναι πολύ δημοφιλές λόγω της απλότητας και της ευελιξίας του, και χρησιμοποιείται ευρέως σε αρκετές

εφαρμογές και είναι συμβατό με πολλές γλώσσες προγραμματισμού, όπως στην περίπτωση μας στην Python.

Για να μπορέσουμε να τροποποιήσουμε τα δεδομένα σημαντικό θα είναι να καταλάβουμε τον κώδικα.

```
{
  "_module": "pandapower.auxiliary",
  "_class": "pandapowerNet",
  "_object": {
    "bus": {
      "_module": "pandas.core.frame",
      "_class": "DataFrame",
      "_object": "{\\"columns\\": [\\"name\\", \\"vn_kv\\", \\"type\\",
\\"zone\\", \\"in_service\\"], \\"index\\": [0, 1, 2], \\"data\\": [[\\"Bus_0
\\", 0.4, \\"b\\", null, true], [\\"Bus_1\\", 0.4, \\"b
\\", null, true], [\\"Bus_2\\", 0.4, \\"b\\", null, true]]}",
      "orient": "split",
      "dtype": {
        "name": "object",
        "vn_kv": "float64",
        "type": "object",
        "zone": "object",
        "in_service": "bool"
      }
    }
  }
},
```

Εικόνα 31: Δεδομένα για τους Ζυγούς

Όπως φαίνεται στην Εικόνα 31 ξεκινώντας από τις πρώτες γραμμές του κώδικα, διαπιστώνουμε ότι πρόκειται για μια κλάση της βιβλιοθήκης pandapower που περιγράφει ένα ηλεκτρικό δίκτυο. Το πρώτο αντικείμενο, `_object`, περιέχει τις οντότητες του pandapowerNet και περιλαμβάνει όλες τις κατηγορίες που θα αναλυθούν στη συνέχεια, όπως bus, load, line, κ.τ.λ.

Συνεχίζοντας το `"module": "pandas.core.frame"`, το οποίο αναφέρεται στην Pandas μια βιβλιοθήκη της Python, και συγκεκριμένα στο DataFrame. Τα δεδομένα των ζυγών αποθηκεύονται σε ένα DataFrame. DataFrame είναι μια δημοφιλής δομή δεδομένων της Pandas για την αποθήκευση και διαχείριση δεδομένων με τη μορφή πινάκων.

Το δεύτερο `_object` αποτελεί το κύριο τμήμα που μας ενδιαφέρει, καθώς περιέχει τις στήλες, τις γραμμές και τα δεδομένα του. Το πεδίο `"columns"` αντιστοιχεί στις στήλες `"name"`, `"vn_kv"`, `"type"`, `"in_service"`, οι οποίες περιγράφουν κάθε ζυγό. Το πεδίο `"index"` περιέχει τους αριθμούς των γραμμών. Στη συνέχεια, στο `"data"`, μπορούμε να τροποποιήσουμε τα δεδομένα για τους ζυγούς που επιθυμούμε. Το `"orient": "split"` δηλώνει τη διάταξη των δεδομένων, δηλαδή ότι οι στήλες, τα δεδομένα και οι γραμμές είναι αποθηκευμένα ξεχωριστά για ευκολία διαχείρισης. Αυτή η διάταξη δεν πρέπει να τροποποιηθεί. Τέλος, στη γραμμή `"dtype"` υπάρχει επεξήγηση για τον τύπο μεταβλητής που χρησιμοποιείται.

Με δεδομένα για τους ζυγούς είναι:

- `name`: το όνομα του ζυγού
- `vn_kv`: η τάση του σε κιλοβόλτ (kV)
- `type`: τι τύπος είναι δηλαδή b επειδή είναι ζυγός



- `in_service`: άμα είναι σε λειτουργία με `true/false`

Πολύ σημαντικό είναι ότι κάθε φορά που προσθέτουμε ή αφαιρούμε δεδομένα θα πρέπει να αλλάξουμε την τιμή στο `"index"` με πρώτο νούμερο το 0.

Ακριβώς από κάτω θα βρούμε το κομμάτι του κώδικα για τα φορτία μαζί και με την αντλία θερμότητας. (Εικόνα 32)

```

    "load": {
      "_module": "pandas.core.frame",
      "_class": "DataFrame",
      "_object": "{\\"columns\\": [\\"name\\", \\"bus\\", \\"p_mw\\",
        \\"q_mvar\\", \\"const_z_percent\\", \\"const_i_percent\\", \\"sn_mva\\",
        \\"scaling\\", \\"in_service\\", \\"type\\", \\"min_p_mw\\", \\"max_p_mw\\",
        \\"controllable\\"], \\"index\\": [0, 1, 2], \\"data\\": [[\\"Load 1
        \", 1, 0.1164, 0.029172589874744, 0.0, 0.0, 0.12, 1.0, true, \\"wye
        \", 0.0, 0.12, true], [\\"Heat Pump
        \", 1, 0.08, 0.0, 0.0, 0.0, 0.08, 1.0, true, \\"wye
        \", 0.08, 0.25, true], [\\"Load 2
        \", 2, 0.04365, 0.010939721203029, 0.0, 0.0, 0.045, 1.0, true, \\"wye
        \", 0.0, 0.045, true]]}",
      "orient": "split",
      "dtype": {
        "name": "object",
        "bus": "uint32",
        "p_mw": "float64",
        "q_mvar": "float64",
        "const_z_percent": "float64",
        "const_i_percent": "float64",
        "sn_mva": "float64",
        "scaling": "float64",
        "in_service": "bool",
        "type": "object",
        "min_p_mw": "float64",
        "max_p_mw": "float64",
        "controllable": "bool"
      }
    }
  },

```

Εικόνα 32: Δεδομένα για φορτία και την αντλία θερμότητας

Ότι εξηγήσαμε παραπάνω ισχύει και τα φορτία στο δομικό κομμάτι του κώδικα με την διαφορά εδώ να υπάρχουν κάποια επιπλέον δεδομένα τα οποία είναι:

- `bus`: Σε ποιον ζυγό είναι συνδεδεμένο το φορτίο
- `p_mw`: Η ενεργός ισχύς του σε MW.
- `q_mvar`: Η άεργος ισχύς του σε MVar.
- `const_z_percent`: Το ποσοστό σταθερής αντίστασης.
- `const_i_percent`: Το ποσοστό σταθερού ρεύματος.
- `sn_mva`: Η φαινόμενη ισχύς σε MVA.

- scaling: Ο συντελεστής κλίμακας, δηλαδή κατά πόσο της % της ονομαστικής τιμής της ενεργού ή άεργου ισχύς θα λειτουργήσει. Με 1.0 στο 100%, 0.5 στο 50% κ.τ.λ.
- type: Τύπος σύνδεσης (στην περίπτωση μας το wye για τριφασική σύνδεση τύπου αστέρα)
- min\_p\_mw: Η ελάχιστη ενεργός ισχύς σε MW.
- max\_p\_mw: Η μέγιστη ενεργός ισχύς σε MW
- controllable: Αν το φορτίο είναι ελεγχόμενο.

Συνεχίζοντας από κάτω βρίσκονται οι γεννήτριες ειδικού σκοπού (special generator) οι οποίες συνήθως περιλαμβάνουν ανανεώσιμες πηγές ενέργειας, όπως φωτοβολταϊκά ή μικρές ανεμογεννήτριες. Στην περίπτωση μας εδώ θα δηλώσουμε τις φωτοβολταϊκές μονάδες(Εικόνα 33). Με το καινούργιο δεδομένο που εισάγει εδώ είναι το current\_source: που μας δίνεται να επιλέξουμε αν η γεννήτρια λειτουργεί ως πηγή ρεύματος με true ή false.

```

"sgen": {
  "_module": "pandas.core.frame",
  "_class": "DataFrame",
  "_object": "{\\"columns\\": [\\"name\\", \\"bus\\", \\"p_mw\\",
\\"q_mvar\\", \\"sn_mva\\", \\"scaling\\", \\"in_service\\", \\"type\\",
\\"current_source\\"], \\"index\\": [0, 1], \\"data\\": [[\\"PV_1
\\", 1, 0.0, 0.0, null, 1.2, true, \\"PV\\", true], [\\"PV_2
\\", 2, 0.0, 0.0, null, 1.0, true, \\"PV\\", true]]}",
  "orient": "split",
  "dtype": {
    "name": "object",
    "bus": "int64",
    "p_mw": "float64",
    "q_mvar": "float64",
    "sn_mva": "float64",
    "scaling": "float64",
    "in_service": "bool",
    "type": "object",
    "current_source": "bool"
  }
},

```

Εικόνα 33: Δεδομένα φωτοβολταϊκών

Στην παρακάτω Εικόνα 34 υπάρχει το κομμάτι του κώδικα για το την εξωτερική πηγή ενέργειας (ext\_grid). Αυτή η εξωτερική πηγή συνδέεται στο δίκτυο μέσω ενός συγκεκριμένου ζυγού και παρέχει σταθερή τάση σε p.u. και το καινούργιο δεδομένο που είναι η γωνία φάσης(va\_degree).

Η γωνία φάσης μπορεί να είναι είτε θετική, σημαίνει ότι η τάση προηγείται του ρεύματος σε φορτία που είναι επαγωγικά, είτε αρνητική, η τάση υστερεί σε σχέση με το ρεύμα σε χωρητικά φορτία, είτε ίση με το μηδέν δηλαδή ότι η τάση και το ρεύμα είναι συγχρονισμένα (φτάνουν στη μέγιστη τιμή τους ταυτόχρονα).

```

"ext_grid": {
  "_module": "pandas.core.frame",
  "_class": "DataFrame",
  "_object": "{\\"columns\\": [\\"name\\", \\"bus\\", \\"vm_pu\\",
\\"va_degree\\", \\"in_service\\"], \\"index\\": [0], \\"data\\": [[\\"External
Grid\\", 0, 1.02, 0.0, true]]}",
  "orient": "split",
  "dtype": {
    "name": "object",
    "bus": "uint32",
    "vm_pu": "float64",
    "va_degree": "float64",
    "in_service": "bool"
  }
},

```

Εικόνα 34: Δεδομένα για την εξωτερική πηγή ενέργειας

Τέλος έχουμε τα δεδομένα για τις γραμμές που συνδέονται οι ζυγοί μεταξύ τους.(Εικόνα 35)

```

"line": {
  "_module": "pandas.core.frame",
  "_class": "DataFrame",
  "_object": "{\\"columns\\": [\\"name\\", \\"std_type\\", \\"from_bus
\\", \\"to_bus\\", \\"length_km\\", \\"r_ohm_per_km\\", \\"x_ohm_per_km\\",
\\"c_nf_per_km\\", \\"g_us_per_km\\", \\"max_i_ka\\", \\"df\\", \\"parallel\\",
\\"type\\", \\"in_service\\"], \\"index\\": [0, 1], \\"data\\": [[\\"LV_Line_0-1
\\", \\"94-AL1\\/\15-ST1A 0.4
\\", 0, 1, 0.3, 0.306, 0.29, 13.199999999999999, 0.0, 0.35, 1.0, 1, \\"o1
\\", true], [\\"LV_Line_1-2\\", \\"94-AL1\\/\15-ST1A 0.4
\\", 1, 2, 0.3, 0.306, 0.29, 13.199999999999999, 0.0, 0.35, 1.0, 1, \\"o1
\\", true]]}",
  "orient": "split",
  "dtype": {
    "name": "object",
    "std_type": "object",
    "from_bus": "uint32",
    "to_bus": "uint32",
    "length_km": "float64",
    "r_ohm_per_km": "float64",
    "x_ohm_per_km": "float64",
    "c_nf_per_km": "float64",
    "g_us_per_km": "float64",
    "max_i_ka": "float64",
    "df": "float64",
    "parallel": "uint32",
    "type": "object",
    "in_service": "bool"
  }
},

```

Εικόνα 35: Δεδομένα για τις γραμμές

Με καινούργια δεδομένα να έχουμε:

- `std_type`: Ο τυποποιημένος τύπος της γραμμής, που αναφέρεται σε έναν τύπο καλωδίου
- `r_ohm_per_km`: Η αντίσταση της σε ohm
- `x_ohm_per_km`: Η αυτεπαγωγή της σε ohm
- `c_nf_per_km`: Η χωρητικότητα της σε νανοφαράντ (nF)

- `g_us_per_km`: Η αγωγιμότητα της σε μικροσίμενς ( $\mu\text{S}$ )

όλα τα παραπάνω ανά χιλιόμετρο

- `max_i_ka`: Το μέγιστο επιτρεπτό ρεύμα της σε κιλοαμπέρ (kA)
- `df`: Ο συντελεστής σχεδιασμού (derating factor) της γραμμής δηλαδή το ποσοστό ρεύματος που μπορεί να αντέξει η γραμμή με 1.0 να μπορεί να αντέξει την τιμή του ονομαστικού ρεύματος της.
- `type`: Ο τύπος γραμμής π.χ., "ol" (overhead line) δηλαδή είναι μια εναέρια γραμμή.

Εντός του φακέλου `power`, παρατηρούμε την ύπαρξη δύο αρχείων τύπου CSV.

Ένα αρχείο CSV (Comma-Separated Values) είναι ένα απλό τύπου αρχείου που χρησιμοποιείται για την αποθήκευση δεδομένων σε μορφή πίνακα, όπως σε ένα λογιστικό φύλλο (π.χ. Excel ή Google Sheets). Τα αρχεία CSV είναι απλά αρχεία κειμένου όπου κάθε γραμμή αντιπροσωπεύει μια εγγραφή στον πίνακα και οι τιμές κάθε στήλης διαχωρίζονται με κόμμα (,). Μια εγγραφή σε έναν πίνακα δεδομένων, όπως σε ένα αρχείο CSV, αναφέρεται σε ένα σύνολο δεδομένων που αντιπροσωπεύει μια πλήρη μονάδα πληροφορίας. Συνήθως, κάθε εγγραφή αποτελεί μια ολόκληρη γραμμή στον πίνακα.

Τα αρχεία αυτά περιέχουν χρονοσειρές, ανά ένα τέταρτο, κατανάλωσης των φορτίων (Εικόνα 36) και παραγωγή των φωτοβολταϊκών αντίστοιχα (Εικόνα 37). Σε περίπτωση που έχουμε προσθέσει ή αφαιρέσει δεδομένα στο προηγούμενο αρχείο, θα πρέπει να προβούμε σε αντίστοιχες αλλαγές και στα αρχεία αυτά, διασφαλίζοντας ότι η αποθήκευση πραγματοποιείται ως αρχείο τύπου CSV (.csv).

	A	B	C	D
1	ts,Load_1,Load_2			
2	2019-02-01 00:00:00,0.0310,0.0170			
3	2019-02-01 00:15:00,0.0309,0.0167			
4	2019-02-01 00:30:00,0.0301,0.0175			
5	2019-02-01 00:45:00,0.0306,0.0190			
6	2019-02-01 01:00:00,0.0315,0.0193			
7	2019-02-01 01:15:00,0.0317,0.0199			
8	2019-02-01 01:30:00,0.0311,0.0192			
9	2019-02-01 01:45:00,0.0310,0.0179			
10	2019-02-01 02:00:00,0.0304,0.0161			
11	2019-02-01 02:15:00,0.0292,0.0141			
12	2019-02-01 02:30:00,0.0282,0.0130			
13	2019-02-01 02:45:00,0.0276,0.0119			
14	2019-02-01 03:00:00,0.0267,0.0107			
15	2019-02-01 03:15:00,0.0249,0.0097			
16	2019-02-01 03:30:00,0.0240,0.0080			
17	2019-02-01 03:45:00,0.0231,0.0075			
18	2019-02-01 04:00:00,0.0220,0.0071			
19	2019-02-01 04:15:00,0.0206,0.0068			
20	2019-02-01 04:30:00,0.0182,0.0065			
21	2019-02-01 04:45:00,0.0163,0.0062			
22	2019-02-01 05:00:00,0.0144,0.0058			
23	2019-02-01 05:15:00,0.0125,0.0052			
24	2019-02-01 05:30:00,0.0116,0.0048			
25	2019-02-01 05:45:00,0.0102,0.0057			
26	2019-02-01 06:00:00,0.0108,0.0059			
27	2019-02-01 06:15:00,0.0110,0.0077			
28	2019-02-01 06:30:00,0.0133,0.0099			
29	2019-02-01 06:45:00,0.0151,0.0122			

Εικόνα 36: CSV αρχείο για τα φορτία

	A	B	C	D
1	ts,PV_1,PV_2			
2	2019-02-01 00:00:00,0.000,0.000			
3	2019-02-01 00:15:00,0.000,0.000			
4	2019-02-01 00:30:00,0.000,0.000			
5	2019-02-01 00:45:00,0.000,0.000			
6	2019-02-01 01:00:00,0.000,0.000			
7	2019-02-01 01:15:00,0.000,0.000			
8	2019-02-01 01:30:00,0.000,0.000			
9	2019-02-01 01:45:00,0.000,0.000			
10	2019-02-01 02:00:00,0.000,0.000			
11	2019-02-01 02:15:00,0.000,0.000			
12	2019-02-01 02:30:00,0.000,0.000			
13	2019-02-01 02:45:00,0.000,0.000			
14	2019-02-01 03:00:00,0.000,0.000			
15	2019-02-01 03:15:00,0.000,0.000			
16	2019-02-01 03:30:00,0.000,0.000			
17	2019-02-01 03:45:00,0.000,0.000			
18	2019-02-01 04:00:00,0.000,0.000			
19	2019-02-01 04:15:00,0.000,0.000			
20	2019-02-01 04:30:00,0.000,0.000			
21	2019-02-01 04:45:00,0.000,0.000			
22	2019-02-01 05:00:00,0.000,0.000			
23	2019-02-01 05:15:00,0.000,0.000			
24	2019-02-01 05:30:00,0.000,0.000			
25	2019-02-01 05:45:00,0.000,0.000			
26	2019-02-01 06:00:00,0.000,0.000			
27	2019-02-01 06:15:00,0.000,0.000			
28	2019-02-01 06:30:00,0.000,0.000			
29	2019-02-01 06:45:00,0.000,0.000			

Εικόνα 37: CSV αρχείο για τα την παραγωγή των φωτοβολταϊκών

Σύμφωνα με την Εικόνα 38 παρατηρούμε στον φάκελο `simulators` και έπειτα στον υποφάκελο `el_network` την ύπαρξη αρχείων κώδικα Python. Ωστόσο, δεν απαιτείται κάποια τροποποίηση σε αυτά τα αρχεία.

> cosim\_pandapipes\_pandapower > simulators > el\_network >

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
__pycache__	4/7/2024 1:48 μμ	Φάκελος αρχείων	
__init__.py	3/7/2024 5:46 μμ	Αρχείο PY	1 KB
mosaik_wrapper.py	3/7/2024 5:46 μμ	Αρχείο PY	9 KB
simulator.py	3/7/2024 5:46 μμ	Αρχείο PY	15 KB

Εικόνα 38: Διαδρομή υποφακέλου el\_network

Αλλαγές θα χρειαστούν όμως στον κυρίως κώδικα στα αρχεία <<benchmark\_multi\_energy\_sim.py>> και <<benchmark\_multi\_energy\_analysis.py>>.

Ανοίγουμε όπως δείξαμε παραπάνω την Spyder και ξεκινάμε με τον πρώτο αρχείο κώδικα. Θα επιστρέψουμε αρκετές φορές σε αυτά τα δυο αρχεία για να κάνουμε αλλαγές.

```

172     )
173
174     # Add electrical network components to collection of entities.
175     grid = entities['el_network'].children
176     entities.update( {element.eid: element for element in grid if element.type in 'Load'} )
177     entities.update( {element.eid: element for element in grid if element.type in 'Sgen'} )
178     entities.update( {element.eid: element for element in grid if element.type in 'Bus'} )
179     entities.update( {element.eid: element for element in grid if element.type in 'Line'} )
180
181     # Time series player for the power consumption profile of load 1.
182     entities['consumer_Load1'] = simulators['load_gen_profiles'].TimeSeriesPlayer(
183         t_start = START_TIME,
184         series = profiles['power_demand'].copy(),
185         fieldname = 'Load_1',
186         interp_method = 'pchip',
187     )
188
189     # Time series player for the power consumption profile of load 2.
190     entities['consumer_Load2'] = simulators['load_gen_profiles'].TimeSeriesPlayer(
191         t_start = START_TIME,
192         series = profiles['power_demand'].copy(),
193         fieldname = 'Load_2',
194         interp_method = 'pchip',
195     )
196
197     # Time series player for generation profile of PV 1.
198     entities['gen_pv1'] = simulators['load_gen_profiles'].TimeSeriesPlayer(
199         t_start = START_TIME,
200         series = profiles['pv_generation'].copy(),
201         fieldname = 'PV_1',
202         interp_method = 'pchip',
203     )
204
205     # Time series player for generation profile of PV 2.
206     entities['gen_pv2'] = simulators['load_gen_profiles'].TimeSeriesPlayer(
207         t_start = START_TIME,
208         series = profiles['pv_generation'].copy(),
209         fieldname = 'PV_2',
210         interp_method = 'pchip',
211     )

```

Εικόνα 39: Κομμάτι του κώδικα προσομοίωσης - αρχικοποίηση και δήλωση των οντοτήτων φορτίων και φωτοβολταϊκών

Στις σειρές 181-211 (Εικόνα 39) παρατηρούμε με ποιον τρόπο δημιουργούνται και αρχικοποιούνται οι οντότητες των φορτίων και των φωτοβολταϊκών, αντίστοιχα κάνουμε και εμείς.

Παρακάτω στην Εικόνα 40 μέσω της συνάρτησης `connectEntities` θα γίνει η σύνδεση των οντοτήτων που δηλώσαμε παραπάνω με τα φορτία και τα φωτοβολταϊκά. Η σύνδεση αυτή διασφαλίζει ότι τα δεδομένα κατανάλωσης ή παραγωγής που προέρχονται από τις χρονοσειρές θα τροφοδοτούν τις αντίστοιχες μονάδες του δικτύου κατά τη διάρκεια της προσομοίωσης.

```
benchmark_multi_energy_sim.py X benchmark_multi_energy_analysis.py X
298 def connectEntities(world, entities):
299     '''
300     Add connections between the simulator entities.
301     '''
302     from simulators.el_network.simulator import make_eid as grid_id
303
304     # Connect electrical consumption profiles to electrical loads.
305     world.connect(entities['consumer_Load1'], entities[grid_id('Load_1',0)], ('out', 'p_mw'))
306     world.connect(entities['consumer_Load2'], entities[grid_id('Load_2',0)], ('out', 'p_mw'))
307
308     # Connect PV profiles to static generators.
309     world.connect(entities['gen_pv1'], entities[grid_id('PV_1',0)], ('out', 'p_mw'))
310     world.connect(entities['gen_pv2'], entities[grid_id('PV_2',0)], ('out', 'p_mw'))
```

Εικόνα 40: Σύνδεση οντοτήτων

Το δεύτερο αρχείο κώδικα το `analysis`, ορίζει ένα λεξικό με το όνομα `PLOT_DICT`, το οποίο χρησιμοποιείται για την οργάνωση και την προετοιμασία δεδομένων που θα εμφανιστούν σε γραφήματα (plots) κατά την ανάλυση των αποτελεσμάτων της προσομοίωσης. Με τον όρο λεξικό στην Python εννοούμε μια δομή δεδομένων που αποτελείται από ζεύγη κλειδιών (keys) και τιμών (values). Σε κάθε κλειδί αντιστοιχεί μία τιμή, και μέσω αυτού του κλειδιού μπορούμε να αναζητούμε ή να ανακτούμε την τιμή που αντιστοιχεί σε αυτό.

Στις σειρές 58-86 της Εικόνα 41 βρίσκονται οι ονομασίες των κλειδιών πριν τις αγκύλες, μέσα στις πρώτες αγκύλες οι τίτλοι των γραφημάτων και έπειτα η λίστα με τις οντότητες και τα αντίστοιχα πεδία που πρόκειται να απεικονιστούν. Θα πρέπει και εδώ αντίστοιχα να ενημερώσουμε τις οποιεσδήποτε αλλαγές έχουμε κάνει.

```
benchmark_multi_energy_sim.py X benchmark_multi_energy_analysis.py X
58     'electrical consumption': [
59         'electrical consumption in MW',
60         [
61             'Load_1_0.p_mw',
62             'Load_2_0.p_mw',
63             'Heat Pump_0.p_mw',
64         ]
65     ],
66     'PV generation': [
67         'PV generation in MW',
68         [
69             'PV_1_0.p_mw',
70             'PV_2_0.p_mw',
71         ]
72     ],
73     'voltage levels': [
74         'voltage levels in p.u.',
75         [
76             'Bus_1_0.vm_pu',
77             'Bus_2_0.vm_pu',
78         ]
79     ],
80     'line loadings': [
81         'line loading in %',
82         [
83             'LV_Line_0-1_0.loading_percent',
84             'LV_Line_1-2_0.loading_percent',
85         ]
86     ],
```

Εικόνα 41: Ορισμός λεξικού `PLOT_DICT` στον κώδικα ανάλυσης των αποτελεσμάτων



Στις σειρές 243-273 αυτές οι γραμμές κώδικα χρησιμοποιούν τη συνάρτηση `plot_results_compare` για να συγκρίνουν αποτελέσματα από δύο διαφορετικά σενάρια (ένα με τον ελεγκτή τάσης απενεργοποιημένο - 'ctrl disabled', και ένα με ελεγκτή τάσης ενεργοποιημένο - 'ctrl enabled') για διάφορες παραμέτρους, όπως τα επίπεδα τάσης των ζυγών και η φόρτωση της κάθε γραμμής. Αντίστοιχά κάνουμε και εδώ τις αλλαγές αν θέλουμε να δούμε το διάγραμμα. (Εικόνα 42)

```

benchmark_multi_energy_sim.py X benchmark_multi_energy_analysis.py X
243 # Compare voltage levels for bus 1.
244 plot_results_compare(
245     'Bus_1_0', 'vm_pu', 'voltage in p.u.',
246     'ctrl disabled', dict_results_ctrl_disabled,
247     'ctrl enabled', dict_results_ctrl_enabled,
248     'voltage_levels_bus1', BINS_BUS_VOLTAGE, SHOW_PLOTS, FIG_TYPE
249 )
250
251 # Compare voltage levels for bus 2.
252 plot_results_compare(
253     'Bus_2_0', 'vm_pu', 'voltage in p.u.',
254     'ctrl disabled', dict_results_ctrl_disabled,
255     'ctrl enabled', dict_results_ctrl_enabled,
256     'voltage_levels_bus2', BINS_BUS_VOLTAGE, SHOW_PLOTS, FIG_TYPE
257 )
258
259 # Compare line loadings for line 1.
260 plot_results_compare(
261     'LV Line 0-1_0', 'loading_percent', 'line loading in %',
262     'ctrl disabled', dict_results_ctrl_disabled,
263     'ctrl enabled', dict_results_ctrl_enabled,
264     'loadings_line1', BINS_LINE_LOADING, SHOW_PLOTS, FIG_TYPE
265 )
266
267 # Compare line loadings for line 2.
268 plot_results_compare(
269     'LV Line 1-2_0', 'loading_percent', 'line loading in %',
270     'ctrl disabled', dict_results_ctrl_disabled,
271     'ctrl enabled', dict_results_ctrl_enabled,
272     'loadings_line2', BINS_LINE_LOADING, SHOW_PLOTS, FIG_TYPE
273 )

```

Εικόνα 42: Συνάρτηση σύγκρισης αποτελεσμάτων

### 4.3.2 Θερμικό Δίκτυο

Σε αντίθεση με το ηλεκτρικό δίκτυο, όπου όλα τα δεδομένα είναι γραμμένα εκτός, στο θερμικό δίκτυο οι όποιες αλλαγές ή παραμετροποιήσεις πραγματοποιούνται απευθείας εντός του κώδικα. Ανοίγουμε από τον φάκελο `simulators` -> `dh_network` το αρχείο `simulator.py` και `mosaik_wrapper.py` στην Spyder. (Εικόνα 43)

> cosim\_pandapipes\_pandapower > simulators > dh\_network >

Όνομα	Ημερομηνία τροποποι...	Τύπος	Μέγεθος
__pycache__	4/7/2024 1:48 μμ	Φάκελος αρχείων	
__init__.py	3/7/2024 5:46 μμ	Αρχείο PY	1 KB
mosaik_wrapper.py	3/7/2024 5:46 μμ	Αρχείο PY	6 KB
simulator.py	3/7/2024 5:46 μμ	Αρχείο PY	29 KB
valve_control.py	3/7/2024 5:46 μμ	Αρχείο PY	7 KB

Εικόνα 43: Διαδρομή υποφακέλου `dh_network`

Ο κώδικας ορίζει μια κλάση με την ονομασία DHNetwork. Η συγκεκριμένη κλάση προσομοιώνει ένα δίκτυο τηλεθέρμανσης (district heating network) και περιλαμβάνει παραμέτρους, μεγέθη, εισόδους και μεταβλητές που σχετίζονται με τη ροή θερμότητας και τις θερμοκρασίες σε ένα τέτοιο σύστημα όπως φαίνεται στην Εικόνα 44. Σε αυτό το σημείο, έχουμε τη δυνατότητα να τροποποιήσουμε τις τιμές των παραμέτρων, καθώς και να προσθέσουμε ή να αφαιρέσουμε καταναλωτές. Είναι σημαντικό να σημειωθεί ότι, λόγω της πολυπλοκότητας του κώδικα, κάθε αλλαγή που πραγματοποιούμε πρέπει να εντοπίζεται σε όλα τα σημεία όπου εμφανίζεται στον κώδικα, ώστε να γίνονται οι αντίστοιχες τροποποιήσεις σε κάθε περίπτωση.

```

22 @dataclass
23 class DHNetwork:
24     """
25     Pandapipes district heating network model.
26     """
27
28     # Parameters
29     T_amb: float = 8 # Ambient ground temperature [degC]
30     enable_logging: bool = True # enable power flow logging
31     T_supply_grid: float = 75 # Supply temperature of the external grid [degC]
32     P_grid_bar: float = 6 # Pressure of the external grid [bar]
33     P_hp_bar: float = 6 # Pressure of the heat pump + storage unit [bar]
34     tank_installed: bool = True # Enable hp + tank connection point
35     dynamic_temp_flow_enabled: bool = True # Enable external temperature flow sim incl. network inertia
36
37     # Magnitudes
38     CP_WATER: float = 4186 # Specific heat capacity of water [J/(kgK)]
39
40     # Input
41     Qdot_evap: float = 0 # Heat consumption of heat pump evaporator [kW]
42     Qdot_cons1: float = 500 # Heat consumption of consumer 1 [kW]
43     Qdot_cons2: float = 500 # Heat consumption of consumer 2 [kW]
44     T_tank_forward: float = 70 # Supply temp of storage unit [degC]
45
46     mdot_cons1_set: float = 4 # Mass flow at consumer 1 [kg/s]
47     mdot_cons2_set: float = 4 # Mass flow at consumer 2 [kg/s]
48     mdot_bypass_set: float = 0.5 # Mass flow through bypass (const.) [kg/s]
49     mdot_grid_set: float = 7.5 # Mass flow injected by grid [kg/s]
50     mdot_tank_in_set: float = 0 # Mass flow injected in the tank [kg/s]
51     mdot_tank_out_set: float = - mdot_tank_in_set # Mass flow supplied by the tank [kg/s]
52
53
54     # Variables
55     T_return_tank: float = 40 # Return temperature of the storage unit [degC]
56     T_evap_in: float = 40 # Return temperature towards the heat pump evaporator [degC]
57     T_return_grid: float = 40 # Return temperature of the external grid [degC]
58     T_supply_cons1: float = 70 # Supply temperature at consumer 1 [degC]
59     T_supply_cons2: float = 70 # Supply temperature at consumer 2 [degC]
60     T_return_cons1: float = 40 # Return temperature at consumer 1 [degC]
61     T_return_cons2: float = 40 # Return temperature at consumer 2 [degC]

```

Εικόνα 44: Κλάση DHNetwork

Αρχικά, θα προχωρήσουμε στη δημιουργία του θερμικού δικτύου. Η διαδικασία ξεκινά με τον ορισμό των κόμβων (junctions), αξιοποιώντας τη βιβλιοθήκη pandaripes και, συγκεκριμένα, τη συνάρτηση pp.create\_junction. (Εικόνα 45)

Τα βασικά χαρακτηριστικά που ορίζονται είναι:

- `rh_bar`: Η αρχική πίεση του κόμβου σε bars, καθορίζεται από τη μεταβλητή `rh_init`, που είναι ίση με `self.P_grid_bar` (πίεση του εξωτερικού δικτύου) που είναι δηλωμένη στις παραμέτρους.
- `tfluid_k`: Η αρχική θερμοκρασία του ρευστού σε Κέλβιν, που είναι ίση με  $273.15 + T\_supply\_grid$  (θερμοκρασία παροχής από βαθμούς Κελσίου σε Κέλβιν).
- `name`: Το όνομα του κόμβου για να μπορεί να αναγνωρίζεται στο δίκτυο.
- `geodata`: Οι συντεταγμένες του κάθε κόμβου στο χώρο.

```

417 def _create_junctions(self):
418     # create nodes (with initial pressure and temperature)
419     net = self.net
420     pn_init = self.P_grid_bar
421     tfluid_init = 273.15 + self.T_supply_grid
422     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n1s', geodata=(0, 1))
423     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n1r', geodata=(0, -2.1))
424     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n2s', geodata=(3, 1))
425     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n2r', geodata=(3, -2.1))
426     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n3s', geodata=(6, 1))
427     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n3s_tank', geodata=(6, 3)) # create hp+tank injection
428     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n3sv', geodata=(6, 1.4)) # create tank valve
429     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n3r', geodata=(6, -2.1))
430     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n3r_tank', geodata=(6, -4.1))
431     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n4s', geodata=(10, 1))
432     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n4r', geodata=(11, -2.1))
433     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n5sv', geodata=(10, 1.5))
434     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n5s', geodata=(10, 4))
435     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n5r', geodata=(11, 4))
436     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n6s', geodata=(15, 1))
437     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n6r', geodata=(16, -2.1))
438     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n7sv', geodata=(15, 1.5))
439     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n7s', geodata=(15, 4))
440     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n7r', geodata=(16, 4))
441     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n8s', geodata=(19, 1))
442     pp.create_junction(net, pn_bar=pn_init, tfluid_k=tfluid_init, name='n8r', geodata=(19, -2.1))

```

Εικόνα 45: Δημιουργία κόμβων

Στη συνέχεια, η κατασκευή του θερμικού δικτύου προχωρά με τη σύνδεση των κόμβων μέσω σωληνώσεων. Η σύνδεση επιτυγχάνεται με τη χρήση της συνάρτησης `pp.create_ripen_from_parameters`, η οποία επιτρέπει τη δημιουργία σωληνώσεων παροχής (Εικόνα 46) και επιστροφής (Εικόνα 47) με καθορισμένες παραμέτρους. Οι παράμετροι που απαιτούνται για τη διαδικασία περιλαμβάνουν:

- `from_junction`: Κόμβος έναρξης.
- `to_junction`: Κόμβος τερματισμού.
- `length_km`: Μήκος της σωλήνωσης σε χιλιόμετρα (km).
- `diameter_m`: Διάμετρος σωλήνωσης σε μέτρα (m).
- `k_mm`: Συντελεστής που μας δείχνει πόσο τραχύ είναι το εσωτερικό τμήμα σε mm (στην περίπτωση μας είναι 0.01 χιλιοστά, κάτι που υποδηλώνει έναν σχεδόν λείο σωλήνα, με χαμηλή αντίσταση στη ροή του ρευστού).
- `sections`: Ο αριθμός των τμημάτων που διαιρείται η σωλήνωση.
- `alpha_w_per_m2k`: Συντελεστής μεταφοράς θερμότητας από τον σωλήνα προς το περιβάλλον ( $1.5 \text{ W/m}^2 \cdot \text{K}$ ).
- `text_k`: Θερμοκρασία περιβάλλοντος σε Κέλβιν ( $273.15 + 8 = 281.15 \text{ K}$ ).
- `name`: Όνομα της σωλήνωσης

```

simulator.py x  mosaik_wrapper.py x
462 def _create_pipes(self):
463     net = self.net
464     j = self.junction
465
466     l01 = 0.5
467
468     # supply pipes
469     pp.create_pipe_from_parameters(net, from_junction=j.index('n1s'), to_junction=j.index('n2s'), length_km=l01,
470                                 diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
471                                 text_k=273.15+8, name='L1s')
472     pp.create_pipe_from_parameters(net, from_junction=j.index('n3sv'), to_junction=j.index('n3s'), length_km=0.01,
473                                 diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
474                                 text_k=273.15 + 8, name='L1s_tank') # create tank pipe connection
475     pp.create_pipe_from_parameters(net, from_junction=j.index('n3s'), to_junction=j.index('n4s'), length_km=0.5,
476                                 diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
477                                 text_k=273.15 + 8, name='L2s')
478     pp.create_pipe_from_parameters(net, from_junction=j.index('n4s'), to_junction=j.index('n5sv'), length_km=0.01,
479                                 diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
480                                 text_k=273.15 + 8, name='L3s')
481     pp.create_pipe_from_parameters(net, from_junction=j.index('n4s'), to_junction=j.index('n6s'), length_km=0.5,
482                                 diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
483                                 text_k=273.15 + 8, name='L4s')
484     pp.create_pipe_from_parameters(net, from_junction=j.index('n6s'), to_junction=j.index('n7sv'), length_km=0.01,
485                                 diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
486                                 text_k=273.15 + 8, name='L5s')
487     pp.create_pipe_from_parameters(net, from_junction=j.index('n6s'), to_junction=j.index('n8s'), length_km=0.01,
488                                 diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
489                                 text_k=273.15 + 8, name='L6s')

```

Εικόνα 46: Δημιουργία σωλήνων παροχής

```

simulator.py x  mosaik_wrapper.py x
491 # return pipes
492 pp.create_pipe_from_parameters(net, from_junction=j.index('n2r'), to_junction=j.index('n1r'), length_km=l01,
493                               diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
494                               text_k=273.15+8, name='L1r')
495 pp.create_pipe_from_parameters(net, from_junction=j.index('n3r'), to_junction=j.index('n3r_tank'), length_km=0.01,
496                               diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
497                               text_k=273.15 + 8, name='L1r_tank') # create tank pipe connection
498 pp.create_pipe_from_parameters(net, from_junction=j.index('n4r'), to_junction=j.index('n3r'), length_km=0.5,
499                               diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
500                               text_k=273.15 + 8, name='L2r')
501 pp.create_pipe_from_parameters(net, from_junction=j.index('n5r'), to_junction=j.index('n4r'), length_km=0.01,
502                               diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
503                               text_k=273.15 + 8, name='L3r')
504 pp.create_pipe_from_parameters(net, from_junction=j.index('n6r'), to_junction=j.index('n4r'), length_km=0.5,
505                               diameter_m=0.1, k_mm=0.01, sections=5, alpha_w_per_m2k=1.5,
506                               text_k=273.15 + 8, name='L4r')
507 pp.create_pipe_from_parameters(net, from_junction=j.index('n7r'), to_junction=j.index('n6r'), length_km=0.01,
508                               diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
509                               text_k=273.15 + 8, name='L5r')
510 pp.create_pipe_from_parameters(net, from_junction=j.index('n8r'), to_junction=j.index('n6r'), length_km=0.01,
511                               diameter_m=0.1, k_mm=0.01, sections=1, alpha_w_per_m2k=1.5,
512                               text_k=273.15 + 8, name='L6r')

```

Εικόνα 47: Δημιουργία σωλήνων επιστροφής

Στη συνέχεια, οι καταναλωτές θα δηλωθούν ως εναλλάκτες θερμότητας (heat exchangers) μαζί με τις αντίστοιχες βαλβίδες τους. Βάσει των παραμέτρων των δεδομένων εισόδου, υπολογίζονται αρχικά οι θερμικές απαιτήσεις των καταναλωτών (q\_hex). Ακολουθεί η δημιουργία των βαλβίδων (pp.create\_valve), οι οποίες τοποθετούνται πριν από τους καταναλωτές, ενσωματώνοντας ένα νέο στοιχείο, τον συντελεστή απωλειών (loss coefficient). Τέλος, δημιουργούνται οι εναλλάκτες-καταναλωτές με την βοήθεια της συνάρτησης pp.create\_heat\_exchanger, όπου παρατηρούμε ότι συνδέουν το κύκλωμα παροχής με το κύκλωμα επιστροφής. (Εικόνα 48)

```

simulator.py x  mosaik_wrapper.py x
522 def _create_substations(self):
523     net = self.net
524     j = self.junction
525     q_hex1 = self.Qdot_cons1 * 1000
526     q_hex2 = self.Qdot_cons2 * 1000
527
528     # create control valves
529     pp.create_valve(net, j.index('n5sv'), j.index('n5s'), diameter_m=0.1, opened=True, loss_coefficient=1000, name='sub_v1')
530     pp.create_valve(net, j.index('n7sv'), j.index('n7s'), diameter_m=0.1, opened=True, loss_coefficient=1000, name='sub_v2')
531
532     # create heat exchanger
533     pp.create_heat_exchanger(net, from_junction=j.index('n5s'), to_junction=j.index('n5r'), diameter_m=0.1,
534                             qext_w=q_hex1, name='hex1')
535     pp.create_heat_exchanger(net, from_junction=j.index('n7s'), to_junction=j.index('n7r'), diameter_m=0.1,
536                             qext_w=q_hex2, name='hex2')

```

Εικόνα 48: Δημιουργία καταναλωτών

Το επόμενο κομμάτι είναι η δημιουργία μιας παράκαμψης (bypass) που τοποθετείται στο τέλος του θερμικού δικτύου με την συνάρτηση `pp.create_valve`. (Εικόνα 49)

```
simulator.py X mosaic_wrapper.py X
567 def _create_bypass(self):
568     net = self.net
569     j = self.junction
570
571     # create bypass valve
572     pp.create_valve(net, j.index('n8s'), j.index('n8r'), diameter_m=0.1, opened=True, loss_coefficient=1000, name='bypass')
573
574     self.valve = net.valve['name'].tolist()
```

Εικόνα 49: Δημιουργία της παράκαμψης bypass

Το τελευταίο κομμάτι κώδικα όπως φαίνεται στην παρακάτω Εικόνα 50 δημιουργεί και ρυθμίζει βαλβίδες ελέγχου ροής (flow control valves) για διάφορα τμήματα του δικτύου θέρμανσης. Κάθε βαλβίδα ελέγχεται από μια κλάση `CtrlValve` που ρυθμίζει τη ροή σε διαφορετικά σημεία του δικτύου με παραμέτρους:

- `gid=v.index`: Επιλέγει την βαλβίδα από τη λίστα με βάση το όνομα της.
- `gain`: Καθορίζει το ποσό που αντιδρά η βαλβίδα στις αλλαγές της ροής(μεγάλο `gain` η βαλβίδα θα ανοίξει πολύ γρήγορα για να επαναφέρει τη ροή στην επιθυμητή τιμή, μικρό `gain` η βαλβίδα θα ανοίξει σταδιακά, κάνοντας μικρότερες αλλαγές, οπότε η ροή θα επανέλθει πιο αργά).
- `order`: Καθορίζει τη σειρά ενεργοποίησης των βαλβίδων ελέγχου. Η βαλβίδα με μικρότερο `order` ενεργοποιείται πρώτη.
- `tol`: Αυτή η παράμετρος καθορίζει την ανοχή (tolerance) για το πώς αντιδρά η βαλβίδα στις διακυμάνσεις.

Τελευταία σειρά αποθηκεύει τα ονόματα όλων των βαλβίδων ελέγχου που δημιουργήθηκαν σε μια λίστα, ώστε να μπορεί το πρόγραμμα να ανατρέξει και να ελέγξει αυτές τις βαλβίδες αργότερα.

```
576 def _create_flow_control(self):
577     net = self.net
578     v = self.valve
579     s = self.sink
580
581     # create supply flow control
582     CtrlValve(net=net, gid=v.index('tank_v1'), gain=-3000,
583              # data_source=data_source, profile_name='tank',
584              level=0, order=1, tol=0.25, name='tank_ctrl1')
585
586     CtrlValve(net=net, gid=v.index('grid_v1'), gain=-3000,
587              # data_source=data_source, profile_name='tank',
588              level=0, order=2, tol=0.25, name='grid_ctrl')
589
590     # create load flow control
591     CtrlValve(net=net, gid=v.index('bypass'), gain=-2000,
592              # data_source=data_source, profile_name='bypass',
593              level=1, order=1, tol=0.25, name='bypass_ctrl')
594     CtrlValve(net=net, gid=v.index('sub_v1'), gain=-100,
595              #data_source=data_source, profile_name='hex1',
596              level=1, order=2, tol=0.1, name='hex1_ctrl')
597     CtrlValve(net=net, gid=v.index('sub_v2'), gain=-100,
598              # data_source=data_source, profile_name='hex2',
599              level=1, order=3, tol=0.1, name='hex2_ctrl')
600
601     self.controller = ['tank_ctrl1', 'grid_ctrl', 'bypass_ctrl', 'hex1_ctrl', 'hex2_ctrl']
```

Εικόνα 50: Δημιουργία βαλβίδων ελέγχου ροής

Μετά την ολοκλήρωση της διαμόρφωσης του θερμικού δικτύου, απαιτούνται ορισμένες επιπλέον τροποποιήσεις στον κώδικα που εντοπίζονται στις πρώτες γραμμές. Το τμήμα αυτού του κώδικα

βρίσκεται μέσα στη συνάρτηση `step_single`, η οποία είναι υπεύθυνη για την ενημέρωση των μεταβλητών του συστήματος (θερμοκρασίες, πιέσεις και μαζικές ροές), αποθηκεύοντάς τες ως μεταβλητές κλάσης. Παράλληλα, είναι απαραίτητο να δηλώσουμε αντίστοιχα όλες τις αλλαγές που έχουν πραγματοποιηθεί, σύμφωνα με τις τροποποιήσεις που έγιναν στη δήλωση των μεταβλητών. (Εικόνα 51)

```

128 # Set output variables
129 self.T_return_tank = round(self.net.res_junction.at[j.index('n3r'), 't_k'] - 273.15, 2)
130 self.T_evap_in = round(self.net.res_junction.at[j.index('n3r'), 't_k'] - 273.15, 2)
131 self.T_return_grid = round(self.net.res_junction.at[j.index('n1r'), 't_k'] - 273.15, 2)
132 self.T_supply_cons1 = round(self.net.res_junction.at[j.index('n5s'), 't_k'] - 273.15, 2)
133 self.T_supply_cons2 = round(self.net.res_junction.at[j.index('n7s'), 't_k'] - 273.15, 2)
134 self.T_return_cons1 = round(self.net.res_junction.at[j.index('n5r'), 't_k'] - 273.15, 2)
135 self.T_return_cons2 = round(self.net.res_junction.at[j.index('n7r'), 't_k'] - 273.15, 2)
136 self.P_supply_cons1 = round(self.net.res_junction.at[j.index('n5s'), 'p_bar'], 2)
137 self.P_supply_cons2 = round(self.net.res_junction.at[j.index('n7s'), 'p_bar'], 2)
138 self.P_return_cons1 = round(self.net.res_junction.at[j.index('n5r'), 'p_bar'], 2)
139 self.P_return_cons2 = round(self.net.res_junction.at[j.index('n7r'), 'p_bar'], 2)
140
141 self.mdot_cons1 = round(self.net.res_valve.at[v.index('sub_v1'), 'mdot_from_kg_per_s'], 2)
142 self.mdot_cons2 = round(self.net.res_valve.at[v.index('sub_v2'), 'mdot_from_kg_per_s'], 2)
143 self.mdot_bypass = round(self.net.res_valve.at[v.index('bypass'), 'mdot_from_kg_per_s'], 2)
144 self.mdot_grid = round(self.net.res_valve.at[v.index('grid_v1'), 'mdot_from_kg_per_s'], 2)
145 self.mdot_tank_out = round(self.net.res_valve.at[v.index('tank_v1'), 'mdot_from_kg_per_s'], 2)
146 self.mdot_tank_in = - self.mdot_tank_out

```

Εικόνα 51: Τροποποιήσεις μεταβλητών

Το επόμενο τμήμα του κώδικα (Εικόνα 52) αφορά μια συνάρτηση με την ονομασία `update`, η οποία χρησιμοποιείται για την ενημέρωση των παραμέτρων ενός δικτύου, όπως οι ροές μάζας (mass flows), οι θερμοκρασίες και τα θερμικά φορτία (heat loads). Απαιτούνται αλλαγές στη μαζική ροή από το δίκτυο (`mdot_grid_set`), η οποία υπολογίζεται ως το άθροισμα των μαζικών ροών των καταναλωτών (`mdot_cons1_set`, `mdot_cons2_set`), της παράκαμψης (`mdot_bypass_set`), και της εκροής νερού από τη δεξαμενή (`mdot_tank_out_set`). Επιπλέον, απαιτούνται τροποποιήσεις στους ελεγκτές των καταναλωτών (`hex1_ctrl`, `hex2_ctrl`) στις γραμμές 384 και 385. Τέλος, στην ενημέρωση των θερμικών φορτίων (`update load`), ο εναλλάκτης θερμότητας (`hex1`) ενημερώνεται με την τιμή  $Q_{dot\_cons1} * 1000$ , όπου το  $Q_{dot\_cons1}$  αναφέρεται στην θερμική κατανάλωση του πρώτου καταναλωτή σε kW, όπως έχει δηλωθεί αρχικά στις παραμέτρους εισόδου. Ο πολλαπλασιασμός με το 1000 μετατρέπει την κατανάλωση από kW σε Watt.

```

369 def update(self):
370     hex = self.heat_exchanger
371     ctrl = self.controller
372     sink = self.sink
373     source = self.source
374     v = self.valve
375
376     self.mdot_tank_out_set = - self.mdot_tank_in_set
377     self.mdot_grid_set = self.mdot_cons1_set + self.mdot_cons2_set + self.mdot_bypass_set - self.mdot_tank_out_set
378
379     # Update grid mass flow
380     self.net.sink.at[sink.index('sink_grid'), 'mdot_kg_per_s'] = self.mdot_grid_set
381
382     # Update controller(s)
383     self.net.controller.at[ctrl.index('bypass_ctrl'), 'object'].set_mdot_setpoint(self.mdot_bypass_set)
384     self.net.controller.at[ctrl.index('hex1_ctrl'), 'object'].set_mdot_setpoint(self.mdot_cons1_set)
385     self.net.controller.at[ctrl.index('hex2_ctrl'), 'object'].set_mdot_setpoint(self.mdot_cons2_set)
386     self.net.controller.at[ctrl.index('grid_ctrl'), 'object'].set_mdot_setpoint(self.mdot_grid_set)
387
388     # Update tank
389     if self.tank_installed:
390         self.net.sink.at[sink.index('sink_tank'), 'mdot_kg_per_s'] = self.mdot_tank_out_set
391         self.net.ext_grid.at[source.index('supply_tank'), 't_k'] = self.T_tank_forward + 273.15
392         self.net.controller.at[ctrl.index('tank_ctrl1'), 'object'].set_mdot_setpoint(self.mdot_tank_out_set)
393
394     # Update load
395     self.net.heat_exchanger.at[hex.index('hex1'), 'qext_w'] = self.Qdot_cons1 * 1000
396     self.net.heat_exchanger.at[hex.index('hex2'), 'qext_w'] = self.Qdot_cons2 * 1000
397     self.net.heat_exchanger.at[hex.index('hp_evap'), 'qext_w'] = self.Qdot_evap * 1000

```

Εικόνα 52: Συνάρτηση `update`

Στη συνέχεια, μεταβαίνουμε στο διπλανό αρχείο, όπως φαίνεται στην παρακάτω Εικόνα 53 που έχουμε ήδη ανοίξει το `mosaik_wrapper.py` προκειμένου να κάνουμε και εκεί ότι αλλαγές έχουμε κάνει στο προηγούμενο αρχείο. Ξεκινώντας από τις γραμμές 22-47 κάνουμε τις απαραίτητες αλλαγές προσθέτοντας ή αφαιρώντας τις αντίστοιχες μεταβλητές στην είσοδο και στην έξοδο.

```

21 # Input
22 'T_tank_forward', # Supply temp of storage unit
23 'mdot_tank_in_set', # Mass flow injected by tank
24 'mdot_grid_set', # Mass flow injected by the grid
25 'mdot_cons1_set', # Mass flow at consumer 1
26 'mdot_cons2_set', # Mass flow at consumer 2
27 'Qdot_evap', # Heat consumption of heat pump evaporator
28 'Qdot_cons1', # Heat consumption of consumer 1
29 'Qdot_cons2', # Heat consumption of consumer 2
30 # Output
31 'T_supply_grid', # Supply temperature of the external grid
32 'T_return_grid', # Return temperature of the external grid
33 'T_return_tank', # Return temperature of the storage unit
34 'T_evap_in', # Return temperature towards the heat pump evaporator
35 'T_supply_cons1', # Supply temperature at consumer 1
36 'T_supply_cons2', # Supply temperature at consumer 2
37 'T_return_cons1', # Return temperature at consumer 1
38 'T_return_cons2', # Return temperature at consumer 2
39 'mdot_tank_in', # Mass flow injected by tank
40 'mdot_grid', # Mass flow injected by the grid
41 'mdot_cons1', # Mass flow at consumer 1
42 'mdot_cons2', # Mass flow at consumer 2
43 'mdot_bypass', # Mass flow at consumer 2
44 'P_supply_cons1', # Supply pressure at consumer 1
45 'P_supply_cons2', # Supply pressure at consumer 2
46 'P_return_cons1', # Return pressure at consumer 1
47 'P_return_cons2', # Return pressure at consumer 2

```

Εικόνα 53: Ενημέρωση μεταβλητών

Στις σειρές 54-70 βρίσκεται η τελευταία κλάση `DHNetworkSimulator` η οποία είναι υπεύθυνη για τη διαχείριση και τον συγχρονισμό πολλαπλών προσομοιώσεων δικτύων θέρμανσης, διατηρεί πληροφορίες για τις εισόδους στο `self.input_vars` και τις εξόδους `self.output_vars`, όπου εδώ θα γίνουν οι οποιοσδήποτε αλλαγές, και καθορίζει το μέγεθος του βήματος χρόνου για την προσομοίωση. (Εικόνα 54)

```

54 class DHNetworkSimulator(Simulator):
55
56     step_size = 10
57     eid_prefix = ''
58     last_time = 0
59
60     def __init__(self, META=META):
61         super().__init__(META)
62
63         # Per-entity dicts
64         self.eid_counters = {}
65         self.simulators: Dict[DHNetwork] = {}
66         self.entityparams = {}
67         self.output_vars = {'T_return_tank', 'T_evap_in', 'T_return_grid', 'T_supply_cons1', 'T_supply_cons2', 'T_return_cons1', 'T_return_cons2',
68                             'P_supply_cons1', 'P_supply_cons2', 'P_return_cons1', 'P_return_cons2',
69                             'mdot_tank_in', 'mdot_grid', 'mdot_cons1', 'mdot_cons2', 'mdot_bypass'}
70         self.input_vars = ('mdot_grid_set', 'T_tank_forward', 'mdot_tank_in_set', 'mdot_cons1_set', 'mdot_cons2_set', 'Qdot_evap', 'Qdot_cons1', 'Qdot_cons2')

```

Εικόνα 54: Κλάση `DHNetworkSimulator`

Αλλαγές θα χρειαστούν όμως στον κυρίως κώδικα μόνο στο αρχείο `benchmark_multi_energy_sim.py`

Στις γραμμές 221-250 της Εικόνα 55 παρατηρείται ο τρόπος με τον οποίο δημιουργούνται και αρχικοποιούνται οι οντότητες των θερμικών καταναλωτών, καθώς και οι οντότητες που αναπαράγουν τα δεδομένα ζήτησης θέρμανσης. Αντίστοιχα, εφαρμόζουμε τις απαραίτητες

τροποποιήσεις, διασφαλίζοντας ότι η διαδικασία είναι συνεπής με τις αλλαγές που έχουμε ήδη πραγματοποιήσει.

```
benchmark_multi_energy_sim.py X
221 # Heat exchanger 1.
222 entities['hex_consumer1'] = simulators['hex_consumer'].HEXConsumer(
223     T_return_target = 40,
224     P_heat = 500,
225     mdot_hex_in = 3.5,
226     mdot_hex_out = -3.5,
227 )
228
229 # Heat exchanger 2.
230 entities['hex_consumer2'] = simulators['hex_consumer'].HEXConsumer(
231     T_return_target = 40,
232     P_heat = 500,
233     mdot_hex_in = 3.5,
234     mdot_hex_out = -3.5,
235 )
236
237 # Time series player for heat demand of consumer 1.
238 entities['heat_profiles1'] = simulators['heat_profiles'].TimeSeriesPlayer(
239     t_start = START_TIME,
240     series = profiles['heat_demand'].copy(),
241     fieldname = 'consumer1',
242 )
243
244
245 # Time series player for heat demand of consumer 2.
246 entities['heat_profiles2'] = simulators['heat_profiles'].TimeSeriesPlayer(
247     t_start = START_TIME,
248     series = profiles['heat_demand'].copy(),
249     fieldname = 'consumer2',
250 )
```

Εικόνα 55: Κομμάτι του κώδικα προσομοίωσης – αρχικοποίηση θερμικών καταναλωτών

Παρακάτω ενημερώνουμε αντίστοιχα στην συνάρτηση connectEntities. Αυτό το τμήμα κώδικα καθορίζει τη ροή δεδομένων μεταξύ διαφόρων οντοτήτων της προσομοίωσης. Συγκεκριμένα, συνδέει τον έλεγχο της ροής μάζας, τη θερμοκρασία και τις θερμικές απαιτήσεις των καταναλωτών με το δίκτυο τηλεθέρμανσης, τους εναλλάκτες θερμότητας - καταναλωτές και τους ελεγκτές θερμότητας. (Εικόνα 56)

```
benchmark_multi_energy_sim.py* X
318 # District heating network.
319 world.connect(entities['flex_heat_ctrl'], entities['dh_network'], ('mdot_1_supply', 'mdot_grid_set'))
320 world.connect(entities['flex_heat_ctrl'], entities['dh_network'], ('mdot_3_supply', 'mdot_tank_in_set'))
321 world.connect(entities['hex_consumer1'], entities['dh_network'], ('mdot_hex_out', 'mdot_cons1_set'))
322 world.connect(entities['hex_consumer2'], entities['dh_network'], ('mdot_hex_out', 'mdot_cons2_set'))
323
324 # Heat demand consumer 1.
325 world.connect(entities['heat_profiles1'], entities['dh_network'], ('out', 'Qdot_cons1'))
326 world.connect(entities['heat_profiles1'], entities['hex_consumer1'], ('out', 'P_heat'))
327 world.connect(entities['hex_consumer1'], entities['flex_heat_ctrl'], ('mdot_hex_out', 'mdot_HEX1'))
328 world.connect(entities['dh_network'], entities['hex_consumer1'], ('T_supply_cons1', 'T_supply'),
329     time_shifted=True, initial_data={'T_supply_cons1': 70})
330
331 # Heat demand consumer 2.
332 world.connect(entities['heat_profiles2'], entities['dh_network'], ('out', 'Qdot_cons2'))
333 world.connect(entities['heat_profiles2'], entities['hex_consumer2'], ('out', 'P_heat'))
334 world.connect(entities['hex_consumer2'], entities['flex_heat_ctrl'], ('mdot_hex_out', 'mdot_HEX2'))
335 world.connect(entities['dh_network'], entities['hex_consumer2'], ('T_supply_cons2', 'T_supply'),
336     time_shifted=True, initial_data={'T_supply_cons2': 70})
```

Εικόνα 56: Συνάρτηση connectEntities

Τέλος η συνάρτηση connectDataCollector διαμορφώνει τις συνδέσεις μεταξύ του συλλέκτη δεδομένων και των διάφορων οντοτήτων στο σύστημα τηλεθέρμανσης. Ο συλλέκτης λαμβάνει



μετρήσεις όπως η ροή μάζας, οι θερμοκρασίες και η θερμική ενέργεια από τις οντότητες του δικτύου, για την ανάλυση των δεδομένων σε πραγματικό χρόνο και σε προσομοιώσεις. Σε αυτό το σημείο θα χρειαστεί να δηλώσουμε ότι μεταβλητές έχουμε προσθέσει πριν στον κώδικα για το θερμικό δίκτυο. (Εικόνα 57)

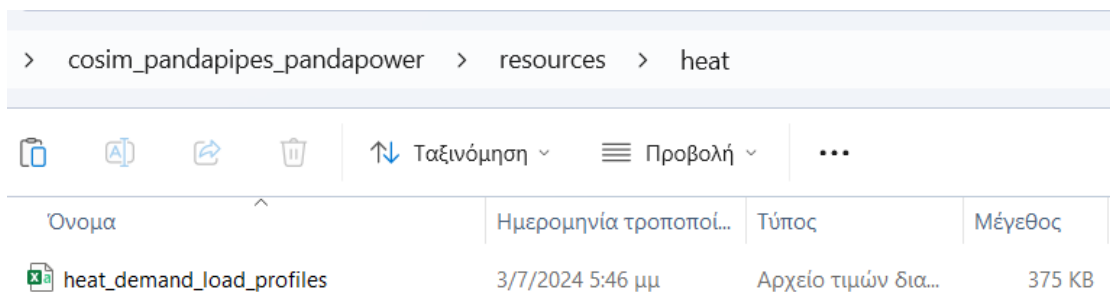
```

benchmark_multi_energy_sim.py* X
371 def connectDataCollector(world, entities):
372     """
373     Configure and connect the data collector.
374     """
375     collector_connections = {}
376
377     collector_connections['storage_tank'] = [
378         'T_cold', 'T_hot', 'T_avg',
379         'mdot_ch_in', 'mdot_dis_in', 'mdot_ch_out', 'mdot_dis_out',
380         'T_ch_in', 'T_dis_in'
381     ]
382
383     collector_connections['hex_consumer1'] = [
384         'P_heat', 'mdot_hex_in', 'mdot_hex_out',
385         'T_supply', 'T_return'
386     ]
387
388     collector_connections['hex_consumer2'] = [
389         'P_heat', 'mdot_hex_in', 'mdot_hex_out',
390         'T_supply', 'T_return'
391     ]
392
393     collector_connections['heat_pump'] = [
394         'T_cond_out', 'T_cond_in',
395         'T_evap_in', 'T_evap_out',
396         'mdot_cond_in', 'mdot_cond_out',
397         'mdot_evap_in', 'mdot_evap_out',
398         'Qdot_cond', 'Qdot_evap',
399         'W_effective', 'W_requested',
400         'W_max', 'W_evap_max', 'W_cond_max', 'W_rated',
401         'P_effective', 'P_requested',
402         'P_rated', 'eta_hp'
403     ]
404
405     collector_connections['dh_network'] = [
406         'T_tank_forward', 'T_supply_cons1', 'T_supply_cons2', 'T_return_cons1', 'T_return_cons2', 'T_return_tank', 'T_return_grid',
407         'mdot_cons1_set', 'mdot_cons2_set', 'mdot_grid_set', 'mdot_tank_in_set',
408         'mdot_cons1', 'mdot_cons2', 'mdot_grid', 'mdot_tank_in',
409         'Qdot_cons1', 'Qdot_cons2', 'Qdot_evap'
410     ]

```

Εικόνα 57: Συνάρτηση connectDataCollector

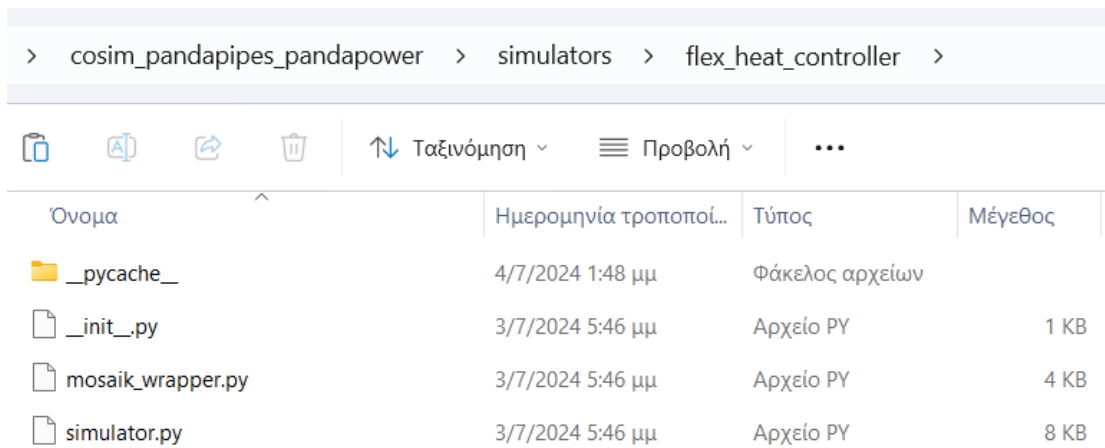
Για να κλείσουμε με το θερμικό δίκτυο θα χρειαστεί να ενημερώσουμε τα δεδομένα της ζήτησης των καταναλωτών σε περίπτωση που κάνουμε αλλαγές όπως δείξαμε παραπάνω στο csv αρχείο. (Εικόνα 58)



Εικόνα 58: Διαδρομή υποφακέλου heat

### 4.3.3 Ελεγκτής Θερμότητας

Ανοίγουμε από τον φάκελο simulators -> flex\_heat\_contoller το αρχείο simulator.py και mosaik\_wrapper.py στην Spyder. (Εικόνα 59)



Εικόνα 59: Διαδρομή υποφακέλου flex\_heat\_controller

Ο κώδικας ορίζει μια κλάση με την ονομασία SimpleFlexHeatController (Εικόνα 60). Η συγκεκριμένη κλάση προσομοιώνει έναν ελεγκτή θερμότητας ο οποίος επηρεάζει την λειτουργία της εγκατάστασης μετατροπής της ηλεκτρικής ενέργειας σε θερμική με διάφορες παραμέτρους και μεταβλητές για την ροή μάζας και την θερμοκρασία σε διάφορα σημεία στο σύστημα. Στις μεταβλητές μπορούμε να δηλώσουμε τις ροές μάζας των καταναλωτών με παρόμοιο τρόπο όπως αυτές των καταναλωτών 1 και 2 (mdot\_HEX1, mdot\_HEX2).

```

7  @dataclass
8  class SimpleFlexHeatController:
9      '''
10     Simple controller for the power-to-heat facility (heat pump & storage tank).
11     '''
12     # Parameters
13     T_tank_max: float = 72 # Maximum tank temperature - [degC]
14     T_tank_min: float = 65 # Minimum tank temperature - [degC]
15
16     # Variables
17     ## Input measures
18     mdot_HEX1: float = 0.0 # Mass flow requested by the consumer 1 HEX - [kg/s]
19     mdot_HEX2: float = 0.0 # Mass flow requested by the consumer 1 HEX - [kg/s]

```

Εικόνα 60: Κλάση SimpleFlexHeatController

Στην παρακάτω Εικόνα 61 θα χρειαστεί κάποιες αλλαγές στην συνάρτηση \_do\_state\_based\_control που καθορίζει τα έξι στάδια λειτουργίας του ελεγκτή θερμότητας. Οι αλλαγές θα περιοριστούν στις πρώτες γραμμές που θα χρειαστεί να προσθέσουμε τις μεταβλητές των καταναλωτών που δηλώσαμε παραπάνω.

```

simulator.py X mosaik_wrapper.py X
64 def _do_state_based_control(self):
65     self.mdot_2_supply = -(self.mdot_HEX1 + self.mdot_HEX2 + self.mdot_bypass)
66     self.mdot_1_return = self.mdot_HEX1 + self.mdot_HEX2 + self.mdot_bypass
67
68     if self.state == 1: # Mode 1: External grid supplies heat, hp and tank inactive
69         self.mdot_1_supply = - self.mdot_2_supply - self.MDOT_FORWARD_MIN
70         self.mdot_HP_out = 0
71
72     elif self.state == 2: # Mode 2: External grid supplies heat, HP charges tank
73         self.mdot_1_supply = - self.mdot_2_supply - self.MDOT_FORWARD_MIN
74         self.set_hp_mdout_out()
75
76     elif self.state == 3: # Mode 3: Discharge the tank, hp off
77         self.mdot_1_supply = self.MDOT_FORWARD_MIN
78         self.mdot_HP_out = 0
79
80     elif self.state == 4: # Mode 4: Discharge the tank, hp on
81         self.mdot_1_supply = self.MDOT_FORWARD_MIN
82         self.set_hp_mdout_out()
83
84     elif self.state == 5: # Mode 5: Tank supports (with fixed mass flow) the grid, hp off
85         self.mdot_1_supply = - self.mdot_2_supply - 2.0
86         self.mdot_HP_out = 0
87
88     elif self.state == 6: # Mode 6: Tank supports (with fixed mass flow) the grid, hp on
89         self.mdot_1_supply = - self.mdot_2_supply - 2.0
90         self.set_hp_mdout_out()

```

Εικόνα 61: Συνάρτηση `_do_state_based_control`

Στη συνέχεια, μεταβαίνουμε στο αρχείο `mosaik_wrapper.py` προκειμένου να δηλώσουμε τις αλλαγές που έχουμε πραγματοποιήσει, ώστε η προσομοίωση να εκτελεστεί σωστά με τα νέα στοιχεία. (Εικόνα 62)

```

simulator.py X mosaik_wrapper.py X
12 META = {
13     'models': {
14         'SimpleFlexHeatController': {
15             'public': True,
16             'params': ['voltage_control_enabled'],
17             'attrs': [
18                 # Input
19                 'mdot_HEX1', 'mdot_HEX2', 'T_tank_hot', 'T_hp_cond_in', 'T_hp_cond_out', 'T_hp_evap_in', 'T_hp_evap_out',
20                 'P_hp_el_setpoint', 'P_hp_effective',
21                 # Output
22                 'mdot_1_supply', 'mdot_2_supply', 'mdot_3_supply',
23                 'mdot_1_return', 'mdot_2_return', 'mdot_3_return',
24                 'mdot_HP_out', 'mdot_tank_in',
25                 'hp_on_request', 'hp_off_request', 'state'
26             ],
27         },
28     },
29 }

```

Εικόνα 62: Ενημέρωση μεταβλητών εισόδου και εξόδου του ελεγκτή

Στο επόμενο κομμάτι κώδικα υπάρχει η κλάση `SimpleFlexHeatControllerSimulator` που διαχειρίζεται και ελέγχει τη ροή θερμότητας σε ένα δίκτυο θέρμανσης μέσω της χρήσης του ελεγκτή `SimpleFlexHeatController`. Συγκεκριμένα θα χρειαστούν αλλαγές στις μεταβλητές εισόδου (`input_vars`). (Εικόνα 63)

```

simulator.py X mosaik_wrapper.py X
32 class SimpleFlexHeatControllerSimulator(Simulator):
33
34     step_size = 10
35     eid_prefix = ''
36     last_time = 0
37
38     def __init__(self, META=META):
39         super().__init__(META)
40
41         # Per-entity dicts
42         self.eid_counters = {}
43         self.simulators: Dict[SimpleFlexHeatController] = {}
44         self.entityparams = {}
45         self.output_vars = {'mdot_1_supply', 'mdot_2_supply', 'mdot_3_supply', 'mdot_1_return', 'mdot_2_return', 'mdot_3_return', 'Q_HP_set', 'mdot_HP_out', 'mdot_tank_in',
46                             'self.input_vars = {'mdot_HEX1', 'mdot_HEX2', 'T_tank_hot', 'T_hp_cond_in', 'T_hp_cond_out', 'T_hp_evap_in', 'T_hp_evap_out', 'P_hp_el_setpoint', 'P_hp_effective'}

```

Εικόνα 63: Κλάση `SimpleFlexHeatControllerSimulator`

Τέλος πηγαίνουμε ξανά στον κυρίως κώδικα της Εικόνα 64 και ελέγχουμε σε περίπτωση που προσθέσαμε παραπάνω καταναλωτές να ενημερώσουμε τις αντίστοιχες μεταβλητές.

```
benchmark_multi_energy_sim.py X
324 # Heat demand consumer 1.
325 world.connect(entities['heat_profiles1'], entities['dh_network'], ('out', 'Qdot_cons1'))
326 world.connect(entities['heat_profiles1'], entities['hex_consumer1'], ('out', 'P_heat'))
327 world.connect(entities['hex_consumer1'], entities['flex_heat_ctrl'], ('mdot_hex_out', 'mdot_HEX1'))
328 world.connect(entities['dh_network'], entities['hex_consumer1'], ('T_supply_cons1', 'T_supply'),
329               time_shifted=True, initial_data={'T_supply_cons1': 70})
330
331 # Heat demand consumer 1.
332 world.connect(entities['heat_profiles2'], entities['dh_network'], ('out', 'Qdot_cons2'))
333 world.connect(entities['heat_profiles2'], entities['hex_consumer2'], ('out', 'P_heat'))
334 world.connect(entities['hex_consumer2'], entities['flex_heat_ctrl'], ('mdot_hex_out', 'mdot_HEX2'))
335 world.connect(entities['dh_network'], entities['hex_consumer2'], ('T_supply_cons2', 'T_supply'),
336               time_shifted=True, initial_data={'T_supply_cons2': 70})
```

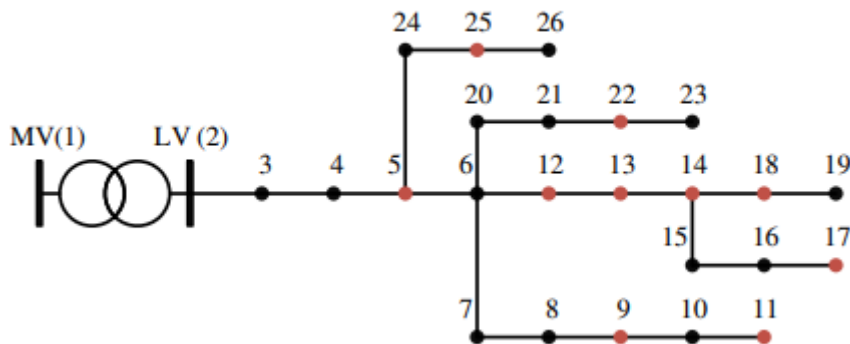
Εικόνα 64: Κομμάτι κώδικα προσομοίωσης - Ενημέρωση καταναλωτών

## Κεφάλαιο 5: Μελέτη περίπτωσης

Στο Κεφάλαιο 5, παρουσιάζεται η πρακτική εφαρμογή του πολυενεργειακού συστήματος που αναπτύχθηκε και προσομοιώθηκε κατά τη διάρκεια της έρευνας. Το κεφάλαιο εστιάζει στην ανάλυση ενός συνδυασμένου ηλεκτρικού και θερμικού δικτύου, με στόχο την αξιολόγηση της απόδοσης της εγκατάστασης και των ελεγκτών που σχεδιάστηκαν. Συγκεκριμένα, γίνεται λεπτομερής παρουσίαση των παραμέτρων του ηλεκτρικού δικτύου, περιλαμβανομένων των ζυγών, των γεννητριών και των φορτίων, καθώς και των στοιχείων του θερμικού δικτύου, όπως οι αντλίες θερμότητας, οι σωληνώσεις και οι δεξαμενές θερμικής αποθήκευσης.

### 5.1 Ηλεκτρικό δίκτυο

Το ηλεκτρικό δίκτυο που χρησιμοποιήθηκε στην παρούσα διπλωματική είναι αυτό της Εικόνα 65.



Εικόνα 65: Ηλεκτρικό Δίκτυο [8]

Το παραπάνω δίκτυο βρίσκεται στη χαμηλή τάση με σύνολο 26 ζυγούς. Με μαύρο απεικονίζονται οι ζυγοί στους οποίους είναι τοποθετημένα τα φορτία και με κόκκινο οι ζυγοί στους οποίους υπάρχουν πηγές. Ως πηγές χρησιμοποιούνται οι φωτοβολταϊκές μονάδες όπου η παραγωγή τους είναι ανεξάρτητες από τις συνθήκες του δικτύου. Όλα τα φορτία λειτουργούν με συντελεστή ισχύος  $pf = 0.95$  επαγωγικά. Αναλυτικά η ισχύς των φορτίων και των φωτοβολταϊκών κάθε ζυγού δίνονται στον Πίνακα 1 και Πίνακα 2. Τα μήκη των γραμμών μεταφοράς που συνδέουν τους ζυγούς δίνονται στον Πίνακα 3.

Για τις γραμμές αναλυτικότερα η σύνθετη αντίσταση γραμμής είναι  $0.6538 + i0.0769 \Omega/km$  στον κορμό ενώ για τους κλάδους του δικτύου είναι  $0.9393 + i0.0909 \Omega/km$ . Η μέγιστη ένταση ρεύματος ορίζεται στα  $0.35 kA$ .

Πίνακας 1: Ισχύς φορτίων

Όνομα	Ζυγό	Ενεργή Ισχύς(kW)
Load_3	1	5
Load_4	2	7.5
Load_6	4	12.5
Load_7	5	5
Load_8	6	7.5
Load_10	8	10
Load_15	13	5
Load_16	14	10
Load_19	17	7.5
Load_20	18	7.5
Load_21	19	7.5
Load_23	21	10
Load_24	22	2.5
Load_26	24	2.5

Πίνακας 2: Ισχύς Φωτοβολταϊκών

Όνομα	Ζυγό	Ενεργή Ισχύς(kW)
PV_5	3	15
PV_9	7	20
PV_11	9	15
PV_12	10	15
PV_13	11	15
PV_14	12	10

PV_17	15	25
PV_18	16	30
PV_22	20	25
PV_25	23	25

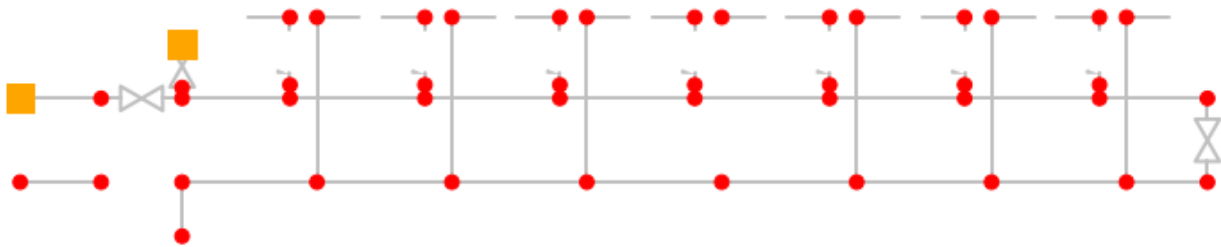
Πίνακας 3: Μήκη γραμμών

Όνομα	Από ζυγό	Προς ζυγό	Μήκος(m)
LV_Line_0-3	0	1	50
LV_Line_3-4	1	2	100
LV_Line_4-5	2	3	75
LV_Line_5-6	3	4	25
LV_Line_6-7	4	5	25
LV_Line_7-8	5	6	50
LV_Line_8-9	6	7	25
LV_Line_9-10	7	8	50
LV_Line_10-11	8	9	75
LV_Line_6-12	4	10	75
LV_Line_12-13	10	11	40
LV_Line_13-14	11	12	50
LV_Line_14-15	12	13	50
LV_Line_15-16	13	14	25
LV_Line_16-17	14	15	50
LV_Line_14-18	12	16	40
LV_Line_18-19	16	17	50
LV_Line_6-20	4	18	25

LV_Line_20-21	18	19	50
LV_Line_21-22	19	20	75
LV_Line_22-23	21	22	50
LV_Line_5-24	3	22	50
LV_Line_24-25	22	23	75
LV_Line_25-26	23	24	50

## 5.2 Θερμικό δίκτυο

Το θερμικό δίκτυο που χρησιμοποιήθηκε στην παρούσα διπλωματική είναι αυτό της Εικόνας 66.



Εικόνα 66: Δίκτυο τηλεθέρμανσης

Στο δίκτυο τηλεθέρμανσης είναι συνδεδεμένο σε έναν άπειρο ζυγό θερμικής ενέργειας όμως. Το νερό, ως θερμικό μέσο, ρέει από τον άπειρο ζυγό προς τα φορτία με την θερμοχωρητικότητα του να θεωρείται ίση με  $Cp = 4.180 \text{ kJ/kg}$ . Όπως φαίνεται και στην παραπάνω εικόνα στο δίκτυο τηλεθέρμανσης υπάρχει και το δίκτυο επιστροφής. Όλοι οι κόμβοι επισημαίνονται με κόκκινο, οι βαλβίδες με γκρι με σύνδεση στους αντίστοιχους κόμβους όπως φαίνεται στον Πίνακα 4 και τα θερμικά στοιχεία ανταλλαγής ενέργειας (heat exchangers) δηλαδή τους καταναλωτές με τα στοιχεία τους να δίνονται στον Πίνακα 5.

Ανάλογα με τη μόνωση του σωλήνα και τη θερμοκρασία του περιβάλλοντος, η θερμοκρασία του νερού που μεταφέρεται μέσω του σωλήνα μεταβάλλεται. Η διάμετρος ορίζεται στα  $d = 0.1 \text{ m}$  για όλους τους σωλήνες, σταθερό συντελεστή μεταφοράς θερμότητας  $alpha = 1.5 \text{ Wm}^{-2}\text{K}^{-1}$  και σταθερή τραχύτητα των σωληνών στα  $k = 0.01 \text{ mm}$ . Τα μήκη των σωληνώσεων που συνδέουν το δίκτυο παροχής δίνονται από τον Πίνακα 6 και το δίκτυο επιστροφής από τον Πίνακα 7.

Πίνακας 4: Βαλβίδες

Όνομα	Από Κόμβο	Προς Κόμβο
sub_v1	n5sv	n5s
sub_v2	n7sv	n7s



sub_v3	n9sv	n9s
sub_v4	n11sv	n11s
sub_v5	n13sv	n13s
sub_v6	n15sv	n15s
sub_v7	n17sv	n17s

Πίνακας 5: Καταναλωτές

Όνομα	Από Κόμβο	Προς Κόμβο
hex1	n5s	n5r
hex2	n7s	n7r
hex3	n9s	n9r
hex4	n11s	n11r
hex5	n13s	n13r
hex6	n15s	n15r
hex7	n17s	n17r

Πίνακας 6: Μήκος σωληνώσεων παροχής

Όνομα	Από κόμβο	Προς κόμβο	Μήκος(km)
l1s	n1s	n2s	0.5
l2s	n3s	n4s	0.5
l3s	n4s	n5sv	0.01
l4s	n4s	n6s	0.5
l5s	n6s	n7sv	0.01
l6s	n6s	n8s	0.5
l7s	n8s	n9sv	0.01
l8s	n8s	n10s	0.5

l9s	n10s	n11sv	0.01
l10s	n10s	n12s	0.5
l11s	n12s	n13sv	0.01
l12s	n12s	n14s	0.5
l13s	n14s	n15sv	0.01
l14s	n14s	n16s	0.5
l15s	n16s	n17sv	0.01
l16s	n16s	n18s	0.01

Πίνακας 7: Μήκος σωληνώσεων επιστροφής

Όνομα	Από κόμβο	Προς κόμβο	Μήκος(km)
l1r	n2r	n1r	0.5
l2r	n4r	n3r	0.5
l3r	n5r	n4r	0.01
l4r	n6r	n4r	0.5
l5r	n7r	n6r	0.01
l6r	n8r	n6r	0.5
l7r	n9r	n8r	0.01
l8r	n10r	n8r	0.5
l9r	n11r	n10r	0.01
l10r	n12r	n10r	0.5
l11r	n13r	n12r	0.01
l12r	n14r	n12r	0.5
l13r	n15r	n14r	0.01
l14r	n16r	n14r	0.5
l15r	n17r	n16r	0.01

l16r	n18r	n16r	0.01
------	------	------	------

Το εξωτερικό θερμικό δίκτυο λειτουργεί ως πηγή πίεσης με προκαθορισμένη θερμοκρασία τροφοδοσίας στους 75 °C και η πίεση ορίζεται στα 6 bar. Θεωρείται ιδανική μονάδα θέρμανσης χωρίς περιορισμούς όσον αφορά τη ροή μάζας.

Επιπλέον, θεωρήθηκε σταθερή εξωτερική θερμοκρασία αγωγών στους 8 °C.

### 5.3 Εγκατάσταση μετατροπής της ηλεκτρικής ενέργειας σε θερμική

Η εγκατάσταση που μετατρέπει την ηλεκτρική ενέργεια σε θερμική αποτελείται από την αντλία θερμότητάς και την δεξαμενή αποθήκευσης της θερμικής ενέργειας. Τα εξαρτήματά της ενεργοποιούνται από τον ελεγκτή θερμότητας. Ο οποίος έχει ελάχιστη και μέγιστη θερμοκρασία τους 65°C και 75°C. Η αντλία θερμότητας ορίζεται με μέγιστη κατανάλωση ισχύος στα 100 kW.

Η δεξαμενή λειτουργεί με δύο βρόγχους των οποίων η λειτουργία αναφέρθηκε σε παραπάνω κεφάλαιο. Τα χαρακτηριστικά της είναι:

- Χωρητικότητα  $V = 100 \text{ m}^3$
- Ύψος  $h = 9.2 \text{ m}$
- Πάχος της μόνωσης  $d = 0.1 \text{ m}$
- Ειδική θερμική αγωγιμότητα της μόνωσης  $k = 0.03 \text{ Wm}^{-1}\text{K}^{-1}$

## Κεφάλαιο 6: Αποτελέσματα

---

Η ανάλυση πραγματοποιήθηκε με γνώμονα την διασύνδεση μεταξύ των δύο δικτύων σε απομακρυσμένους ζυγούς του ηλεκτρικού δικτύου. Στους απομακρυσμένους ζυγούς της τοπολογίας του ηλεκτρικού συστήματος δύναται να είναι εντονότερα φαινόμενα όπως διακύμανση της και παραβιάσεις τάσεων. Η ενσωμάτωση μονάδων διεσπαρμένης παραγωγής, μπορούν να αλλάξουν τη δυναμική του δικτύου και να επηρεάσουν την ισορροπία ισχύος, προκαλώντας είτε πλεονάζουσα παραγωγή είτε ελλείψεις. Ειδικότερα, η ενσωμάτωση μονάδων διεσπαρμένης παραγωγής ενέργειας, όπως φωτοβολταϊκά ή αιολικά πάρκα σε απομακρυσμένους ζυγούς δύναται να εγείρει περαιτέρω ανησυχίες σχετικά με πιθανή πρόσκληση ζητημάτων υπερτάσεων. Η συνδυασμένη λειτουργία του ηλεκτρικού και θερμικού δικτύου μέσα από την εφαρμογή της προτεινόμενης μεθοδολογίας αξιολογείται από την άποψη της μείωσης των παραβάσεων της τάσης και μείωση της υπερφόρτωσης της γραμμής στις δύο προσομοιώσεις του πειράματος με ενεργοποιημένο τον ελεγκτή τάσης και απενεργοποιημένο.

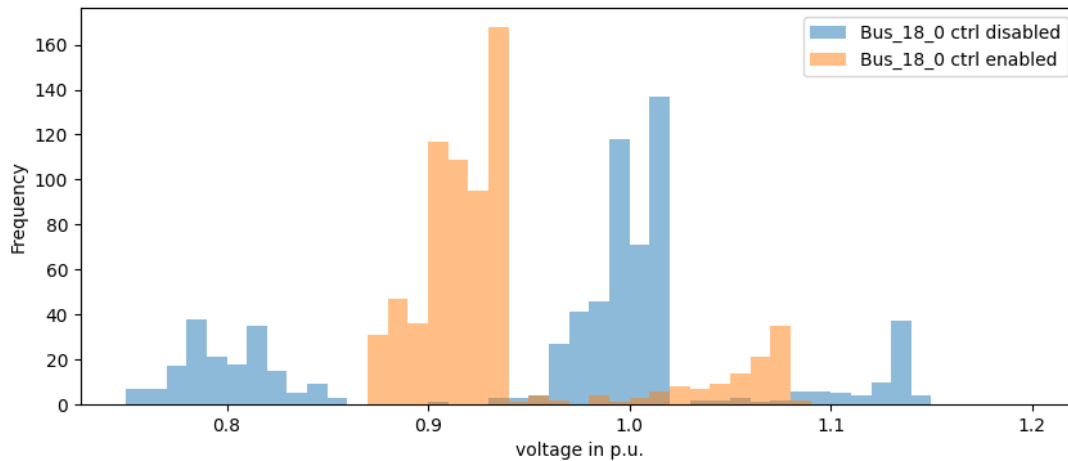
### 6.1 Αποτελέσματα μελέτης

Μελετήθηκαν 5 σενάρια: στα πρώτα τρία σενάρια η αντλία θερμότητας μαζί με τον ελεγκτή τάσης τοποθετήθηκε σε τρεις διαφορετικούς ζυγούς στον ζυγό 18 (σενάριο 1), 22 (σενάριο 2) και 10 (σενάριο 3), ενώ στα σενάρια 4 και 5 αυξήθηκε η παραγωγή των φωτοβολταϊκών κατά 50% στο σενάριο 4 και κατά 100% στο σενάριο 5 θεωρώντας την αντλία θερμότητας και τον ελεγκτή τάσης ενσωματωμένα στον ζυγό 18.

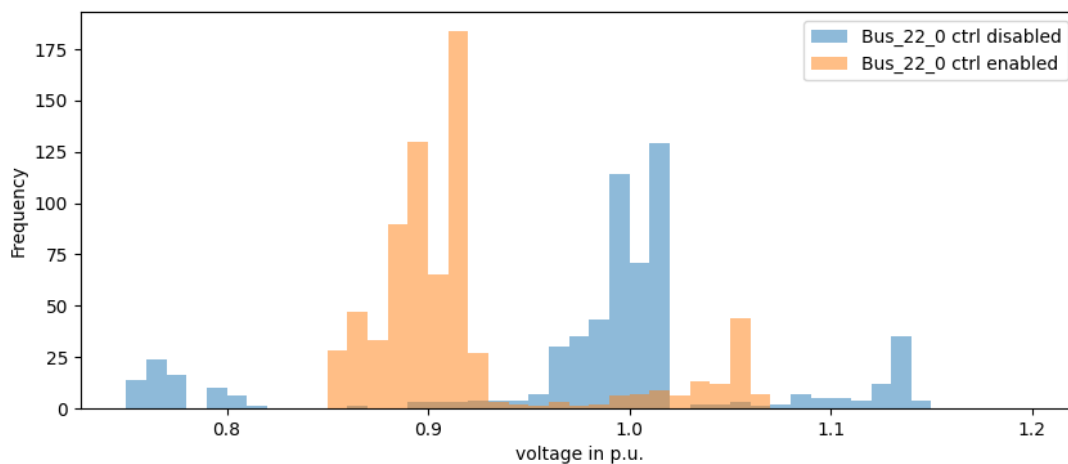
Το Γράφημα 1 απεικονίζει την τάση του ζυγού 18 κατά το σενάριο 1 με και χωρίς την εφαρμογή του ελεγκτή τάσης. Παρατηρώντας το Γράφημα 1 βλέπουμε ότι στον Ζυγό 18 που έχουμε τοποθετημένο τον ελεγκτή τάσης και την αντλία θερμότητας ότι όταν ο ελεγκτής είναι απενεργοποιημένος, η τάση φαίνεται να διαφέρει περισσότερο, με πιο συχνές τιμές να είναι είτε πολύ χαμηλές (κάτω από 0.8 p.u.) είτε υψηλότερες από 1.0 p.u. Η απουσία ελέγχου επιτρέπει μεγαλύτερη διακύμανση, ιδιαίτερα σε καταστάσεις που η κατανάλωση από την αντλία θερμότητας ή η παραγωγή από τα φωτοβολταϊκά μεταβάλλονται γρήγορα. Τα υψηλά επίπεδα τάσης κοντά στο 1.1 p.u. υποδεικνύουν τις στιγμές όπου τα φωτοβολταϊκά παράγουν περισσότερη ισχύ, ενώ τα χαμηλά επίπεδα δείχνουν ότι η κατανάλωση από την αντλία θερμότητας προκαλεί πτώση της τάσης όταν υπάρχει έλλειμμα φωτοβολταϊκής παραγωγής.

Από την άλλη πλευρά, όταν ο ελεγκτής είναι ενεργοποιημένος, παρατηρείται πιο σταθερή συμπεριφορά της τάσης. Η μέση τάση φαίνεται να είναι πιο κοντά στο 1.0 p.u., με μικρότερη

διακύμανση τόσο προς τα κάτω όσο και προς τα πάνω. Αυτό υποδηλώνει ότι ο ελεγκτής τάσης ρυθμίζει την παραγωγή του φωτοβολταϊκού ώστε να αποφεύγονται μεγάλες πτώσεις ή αυξήσεις της τάσης. Έτσι η μειωμένη συχνότητα στις ακραίες τιμές (κάτω από 0.9 και πάνω από 1.1 p.u.) δείχνει ότι ο ελεγκτής εξομαλύνει την τάση του ζυγού. Σε μια παρόμοια προσέγγιση, η τάση στον ζυγό 22 εξομαλύνεται μέσα από την κατάλληλη διαχείριση της αντλίας θερμότητας με την εφαρμογή του ελεγκτή τάσης στο σενάριο 2 όπως περιγράφεται στο Γράφημα 2.

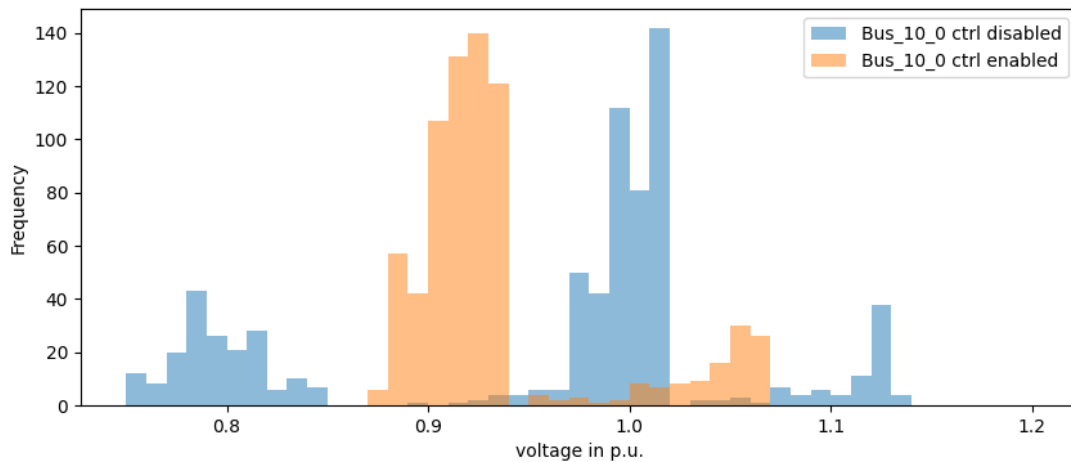


Γράφημα 1: Η τάση του ζυγού 18 σε p.u. (σενάριο 1)



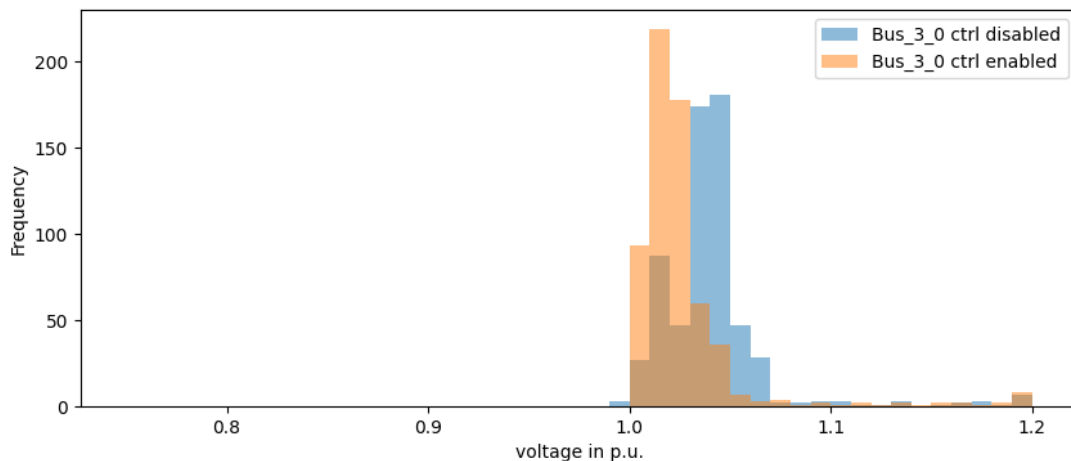
Γράφημα 2: Η τάση του ζυγού 22 σε p.u. (σενάριο 2)

Σε αντίθεση με τα σενάρια 1 και 2, στο σενάριο 3, δεν υπάρχει παραγωγή ισχύος (όπως θα έκανε ένα φωτοβολταϊκό) στον ζυγό που ενσωματώνεται η αντλία θερμότητας και ο ελεγκτής. Κατά συνέπεια ο ζυγός δύναται να αυξήσει και να μειώσει της συνολική κατανάλωση που μπορεί να παρέχει σύμφωνα με την παραγόμενη ισχύ του δικτύου. Όταν ο ελεγκτής τάσης είναι ενεργοποιημένος (enabled) στον ζυγό 10, παρατηρείται καλύτερη σταθεροποίηση της τάσης κοντά στο 1.0 p.u. με μικρές διακυμάνσεις. Αυτό δείχνει ότι ο ελεγκτής δύναται να διαχειριστεί την λειτουργία της αντλίας θερμότητας βελτιώνοντας την τάση, αντισταθμίζοντας τις διακυμάνσεις που προκαλούνται από το φορτίο. Αντίθετα, όταν ο ελεγκτής τάσης είναι απενεργοποιημένος (disabled), η τάση στον ζυγό 10 διατηρείται σχετικά κοντά στο 1.0 p.u. με μεγάλη συχνότητα, δημιουργώντας ωστόσο μεγαλύτερες διακυμάνσεις στις ακραίες τιμές της προκαλούμενης τάσης. Συγκεκριμένα, η απουσία του ελεγκτή προκαλεί τάσεις χαμηλότερες του 0.8 p.u. καθώς και τάσεις μεγαλύτερες του 1.1 p.u.

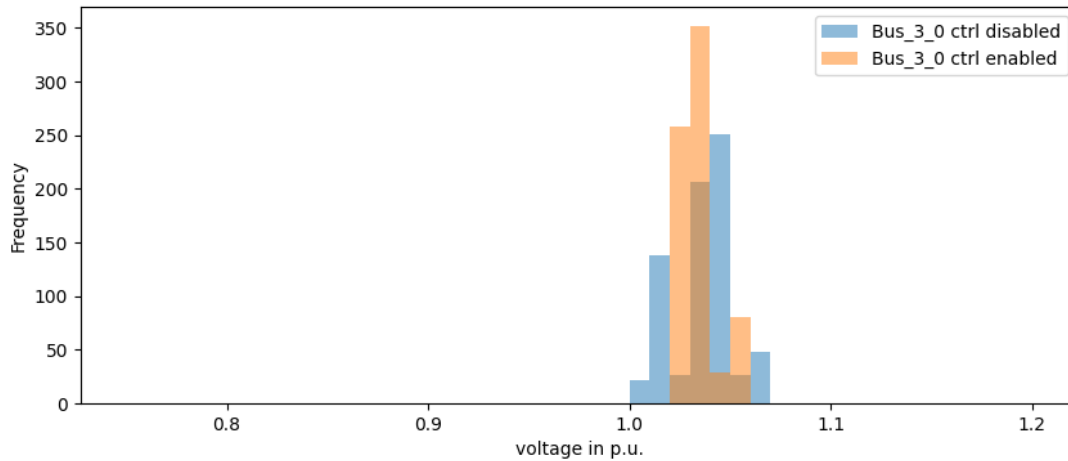


Γράφημα 3: Η τάση του ζυγού 10 σε p.u. (σενάριο 3)

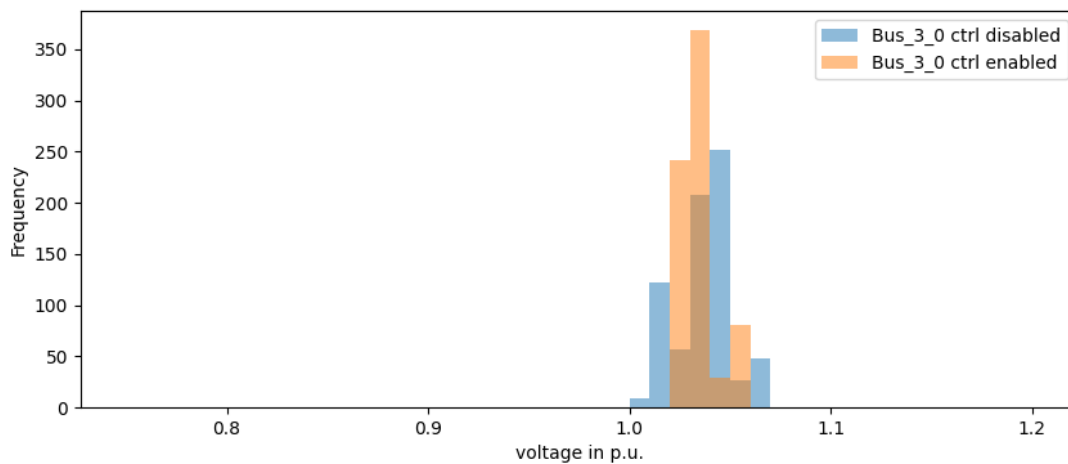
Σε όλα τα σενάρια, η τάση στον ζυγό 3 παραμένει πολύ σταθερή και κοντά στο 1.0 p.u., ανεξάρτητα από την τοποθέτηση της αντλίας θερμότητας στους ζυγούς 18 (Γράφημα 4), 22 (Γράφημα 5) και 10 (Γράφημα 6). Το γεγονός αυτό δείχνει ότι η σύνδεση με τον άπειρο ζυγό επιφέρει μεγάλη σταθερότητα στην τάση. Σε κάθε περίπτωση, η διαφορά μεταξύ του ελεγκτή τάσης ενεργοποιημένου και απενεργοποιημένου είναι ελάχιστη στον ζυγό 3 γεγονός που υποδηλώνει ότι η ρύθμιση της τάσης μέσω του ελεγκτή δεν έχει μεγάλη επίδραση εδώ, λόγω της ισχυρής επίδρασης του άπειρου ζυγού στη σταθερότητα της τάσης.



Γράφημα 4: Η τάση στον ζυγό 3 σε p.u. (σενάριο 1)



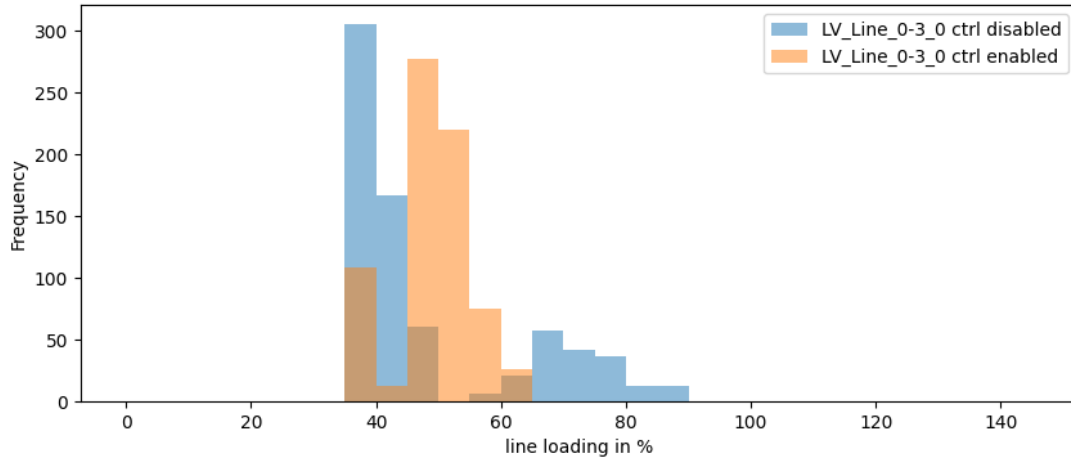
Γράφημα 5: Η τάση στον ζυγό 3 σε p.u. (σενάριο 2)



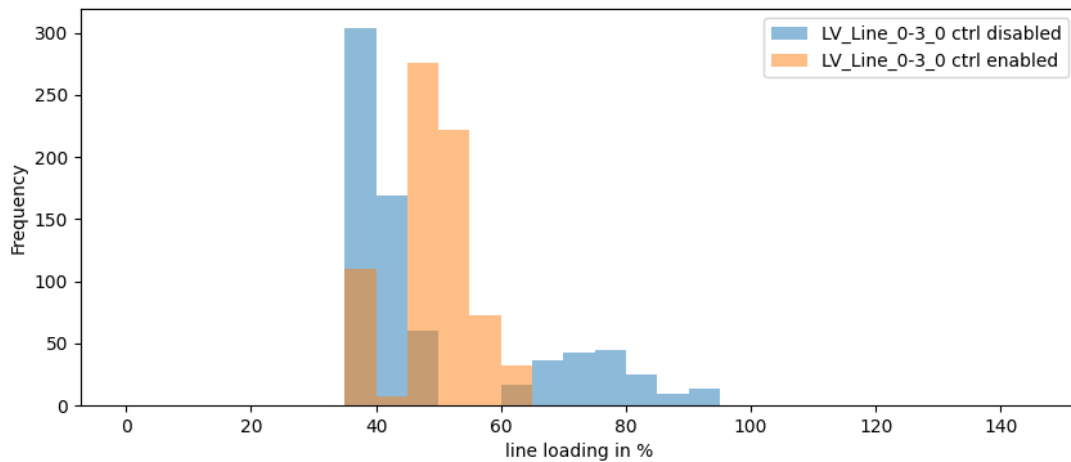
Γράφημα 6: Η τάση στον ζυγό 3 σε p.u. (σενάριο 3)

Οι ζυγοί που βρίσκονται κοντά στον μετασχηματιστή που συνδέει το δίκτυο χαμηλής με την μέση τάση δύναται να παρουσιάζουν χαμηλότερες διακυμάνσεις στις τιμές τάσεις που προκαλούνται. Η απόσταση των ζυγών από τον μετασχηματιστή επηρεάζει την διακύμανση στην προκαλούμενη τάση. Όσο πιο κοντά βρίσκεται ο ζυγός στον μετασχηματιστή, τόσο λιγότερες είναι οι διακυμάνσεις που προκαλούνται στην τάση αυτών των ζυγών.

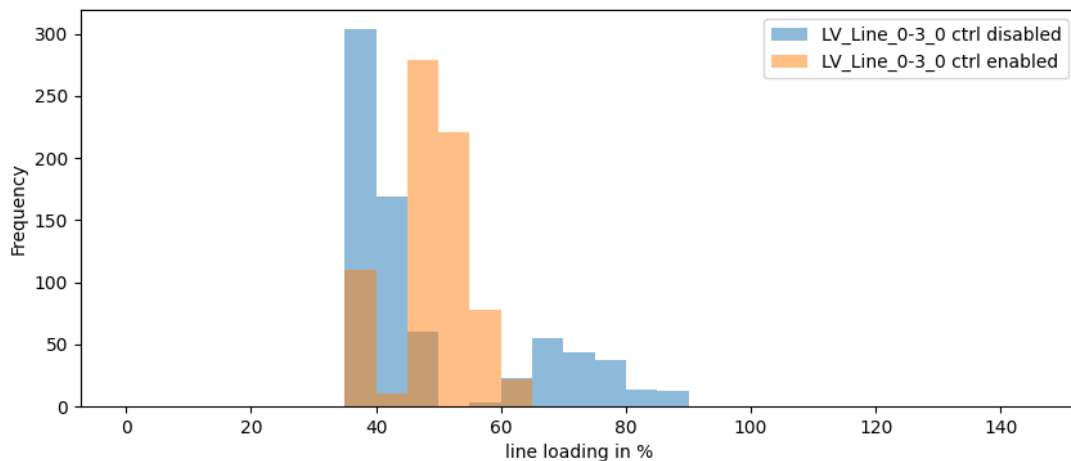
Στο Γράφημα 7 η φόρτωση της γραμμής του σεναρίου 1 είναι πιο συγκεντρωμένη γύρω από τις τιμές 40% - 60% με τη λειτουργία του ελεγκτή τάσης να αυξάνει την φόρτωση στη γραμμή για να καλύψει την απόσταση και τις ανάγκες του ζυγού 18. Υπάρχει μια πιο ομοιόμορφη κατανομή σε αυτό το εύρος, κάτι που υποδηλώνει ότι ο ελεγκτής τάσης συμβάλλει στη διανομή του φορτίου με τρόπο που αποφεύγει την υπερβολική φόρτωση. Στο Γράφημα 8 φαίνεται πως η φόρτωση της γραμμής ήταν ελαφρώς υψηλότερη από το σενάριο 1, καθώς ο ζυγός 22 είναι πιο απομακρυσμένος από τον εξωτερικό ζυγό, γεγονός που απαιτεί περισσότερη ισχύ μέσω της γραμμής. Τέλος στο Γράφημα 9 η φόρτωση της γραμμής στο σενάριο 3 είναι παρόμοια με τα προηγούμενα σενάρια, αλλά ο ελεγκτής τάσης φαίνεται να αυξάνει τη φόρτωση πιο ομοιόμορφα, συγκεντρώνοντας την κατανομή κοντά στο 50%, αποφεύγοντας τις υψηλές συμφορήσεις που προκάλεσε η μη εφαρμογή του.



Γράφημα 7: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 1)



Γράφημα 8: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 2)

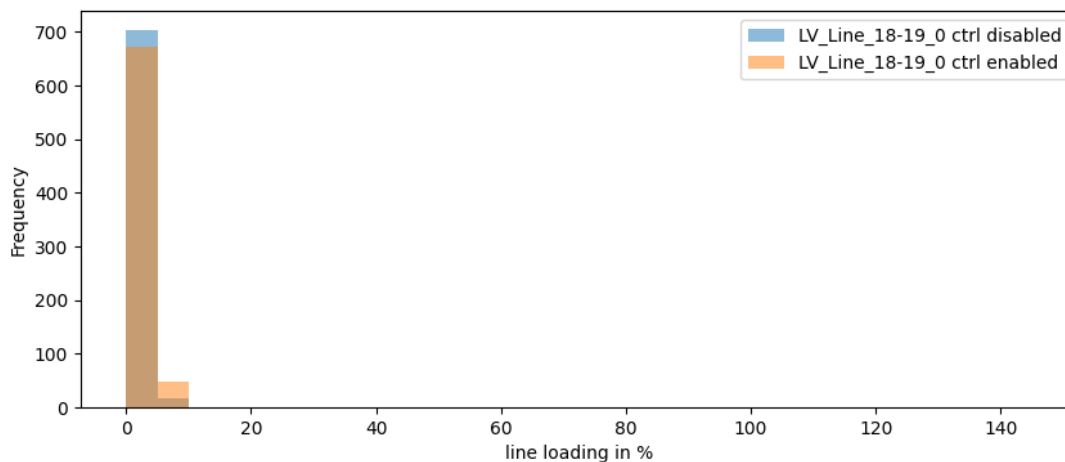


Γράφημα 9: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 3)

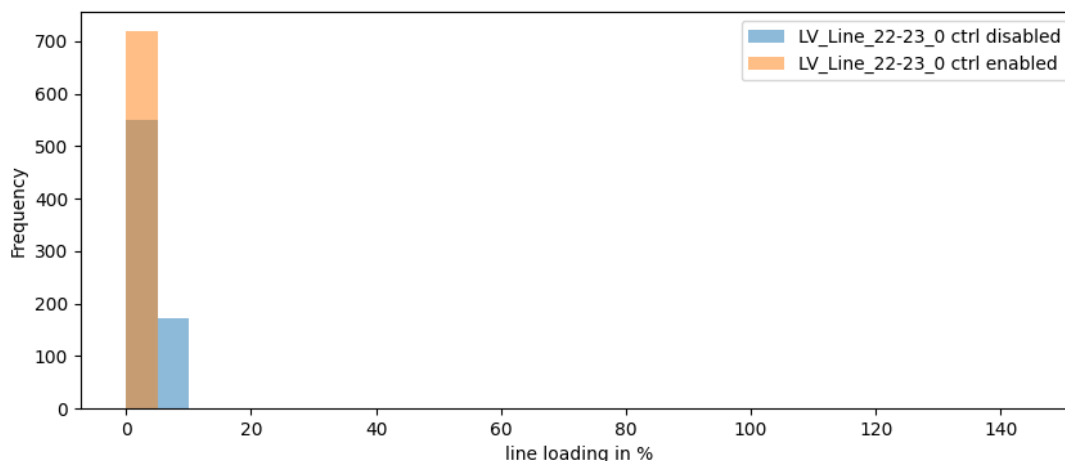
Στο Γράφημα 10 και Γράφημα 11 παρατηρούμε ότι η φόρτωση την γραμμής που συνδέει το ζυγό 18 με τον τελείως ακριανό ζυγό 19 και τον ζυγό 22 με τον 23 και με τις δύο λειτουργίες του ελεγκτή τάσης είναι αρκετά χαμηλή προσεγγίζοντας την μηδενική. Παρόλο που στα δύο σενάρια (1 και 2) στον ζυγό 19 και 22 στις άκρες υπάρχουν φορτία, κάτι που είναι αντιφατικό με την ύπαρξη φορτίου



στους ζυγούς. Αυτό συμβαίνει καθώς τα φορτία υπό εξέταση έχουν πολύ χαμηλή κατανάλωση ισχύος έτσι η γραμμή που συνδέει τον ζυγό 18-19 και τον ζυγό 22-23 μπορεί να μην χρειάζεται να μεταφέρει σημαντική ποσότητα ισχύος. Έτσι, ακόμα και με ένα φορτίο, η φόρτωση της γραμμής παραμένει εξαιρετικά χαμηλή..

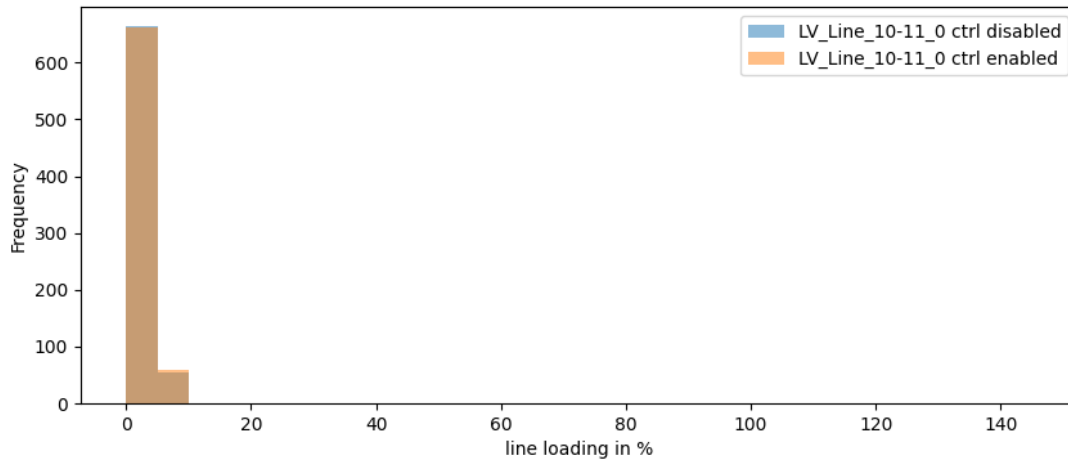


Γράφημα 10: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 1)



Γράφημα 11: Η φόρτωση την γραμμής των ζυγών 22-23 (σενάριο 2)

Το ίδιο συμβαίνει και στο σενάριο 3 στο Γράφημα 12 με την αντλία θερμότητας στον ζυγό 10 με την φόρτωση της γραμμής όμως να έχει παίρνει σχεδόν μηδενική φόρτωση σε σχέση με τον ελεγκτή τάσης απενεργοποιημένο.



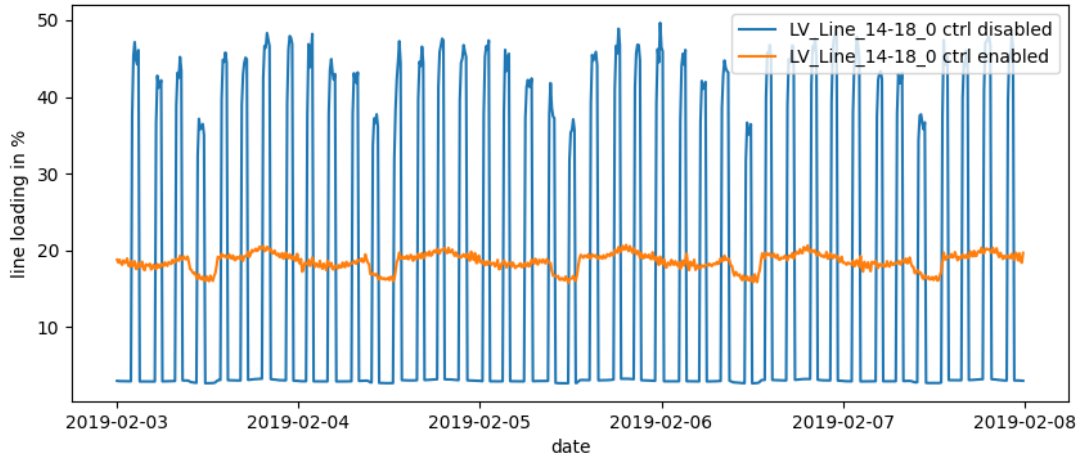
Γράφημα 12: Η φόρτωση της γραμμής των ζυγών 10-11 (σενάριο 3)

Στο Γράφημα 13 για το σενάριο 1 όταν ο ελεγκτής τάσης είναι ενεργοποιημένος, η φόρτωση της γραμμής των ζυγών 14-18 παραμένει σταθερή, γύρω στο 18-20%. Υπάρχουν μικρές διακυμάνσεις, αλλά συνολικά η γραμμή διατηρεί μια σταθερή φόρτωση, υποδεικνύοντας ότι ο ελεγκτής λειτουργεί αποτελεσματικά στη ρύθμιση της τάσης και στη διατήρηση μιας ομοιόμορφης ροής ισχύος.

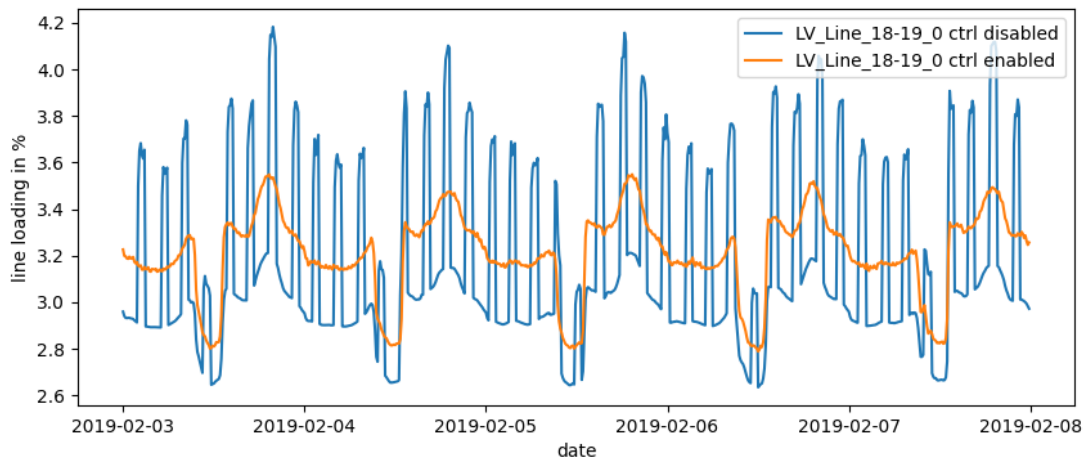
Όταν ο ελεγκτής είναι απενεργοποιημένος, παρατηρούμε μεγάλες διακυμάνσεις στη φόρτωση της γραμμής, με αιχμές που φτάνουν μέχρι και 50% φόρτωση. Αυτό υποδηλώνει ότι, χωρίς τον έλεγχο τάσης, η γραμμή είναι πιο επιρρεπής σε έντονες αυξήσεις και μειώσεις της ροής ισχύος, πιθανώς λόγω των απότομων αλλαγών στη ζήτηση από την αντλία θερμότητας.

Στο Γράφημα 14 η γραμμή 18-19 εμφανίζει παρόμοια συμπεριφορά, με μεγαλύτερη σταθερότητα όταν ο ελεγκτής τάσης είναι ενεργοποιημένος και μεγαλύτερες διακυμάνσεις όταν ο ελεγκτής είναι απενεργοποιημένος. Ωστόσο, οι διακυμάνσεις εδώ δεν είναι τόσο έντονες όσο στη γραμμή 14-18, πιθανώς επειδή ο ζυγός 19 είναι λιγότερο επιβαρυσμένος και έχει μικρότερη κατανάλωση σε σχέση με τον ζυγό 18.

Η σταθεροποίηση της τάσης και της φόρτωσης μέσω του ελεγκτή είναι σημαντική για την αποφυγή μεγάλων διακυμάνσεων και πιθανών προβλημάτων υπερφόρτωσης στις γραμμές του δικτύου.

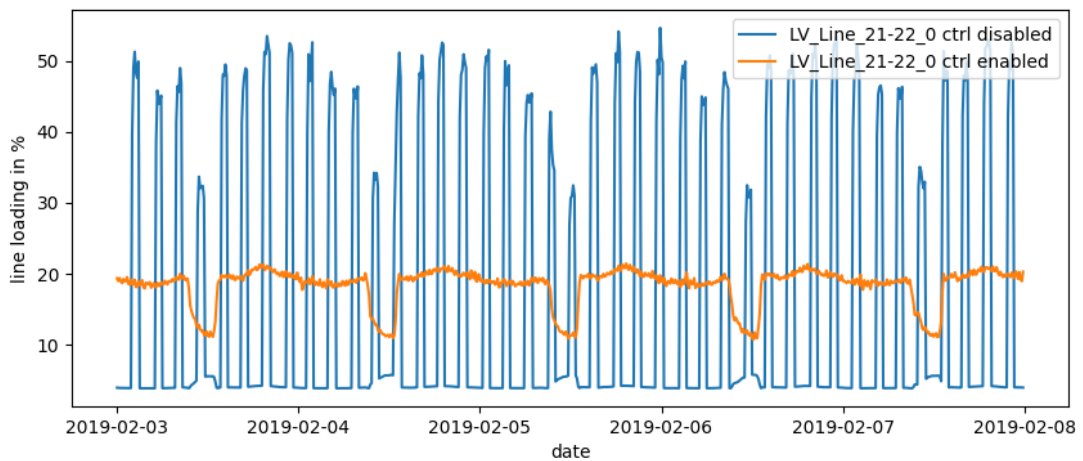


Γράφημα 13: Η φόρτωση της γραμμής των ζυγών 14-18 (σενάριο 1)

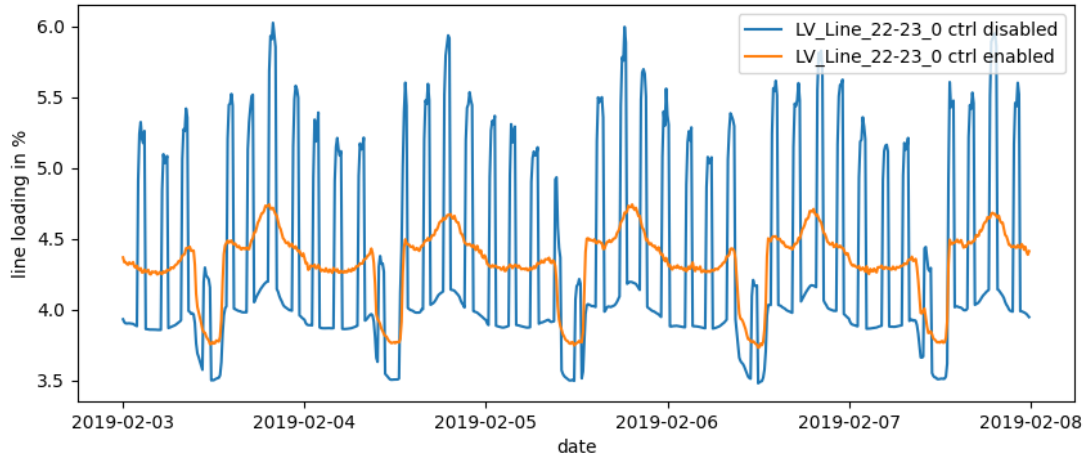


Γράφημα 14: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 1)

Παρομοίως συμβαίνει στο σενάριο 2 στο Γράφημα 15 και Γράφημα 16 όπου παρουσιάζεται η φόρτωση των διπλανών γραμμών του ζυγού 22.

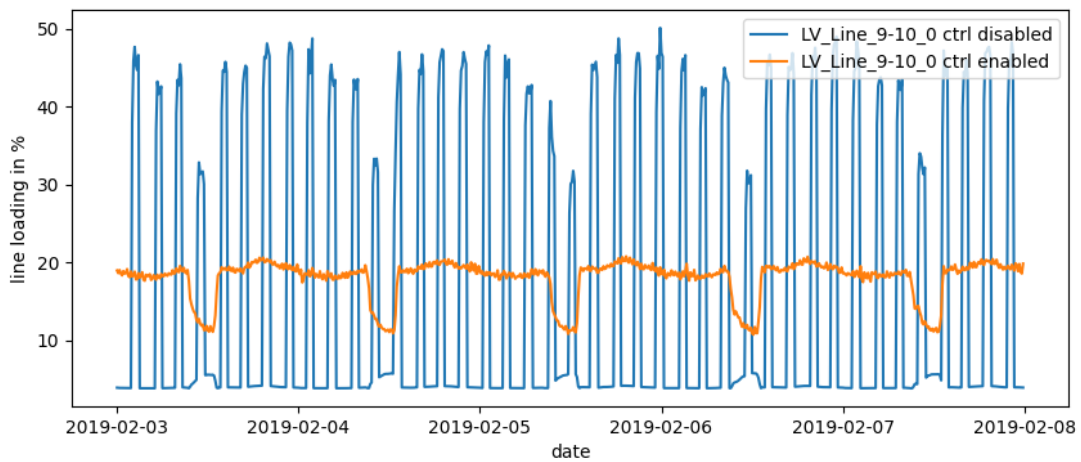


Γράφημα 15: Η φόρτωση της γραμμής των ζυγών 21-22 (σενάριο 2)

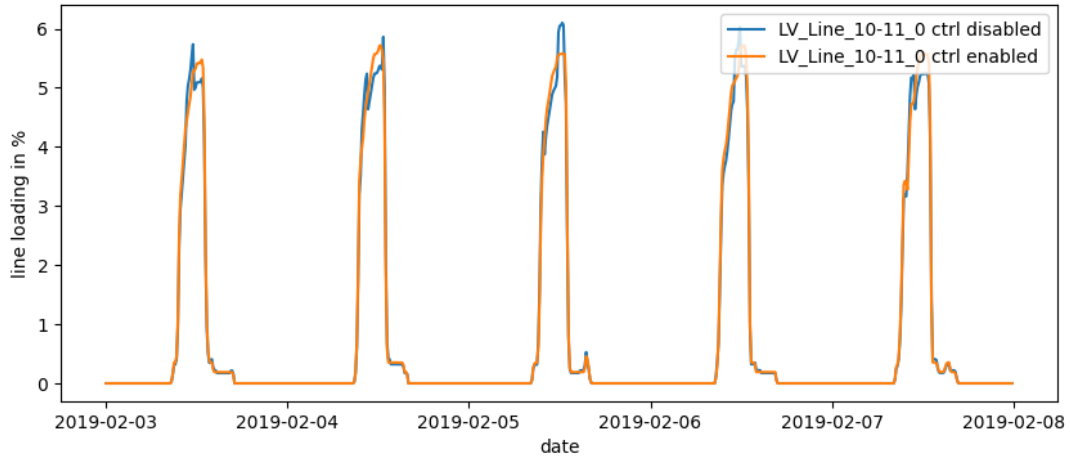


Γράφημα 16: Η φόρτωση της γραμμής των ζυγών 22-23 (σενάριο 2)

Παρόλο που ότι συμβαίνει στα παραπάνω γραφήματα συμβαίνει και στο Γράφημα 17 του σεναρίου 3 κατά το οποίο εξετάζεται η φόρτωση της γραμμής 9-10, ωστόσο στο σενάριο 3 στο Γράφημα 18 όπου εξετάζεται η φόρτωση της γραμμής 10-11, δεν παρατηρούμε ότι με τον ελεγκτή ενεργοποιημένο και απενεργοποιημένο είναι σχεδόν στα ίδια επίπεδα η φόρτιση της γραμμής 10-11. Κατά συνέπεια, η διαχείριση της αντλίας θερμότητας του ζυγού 10 δεν δύναται να επηρεάσει σημαντικά την ενέργεια που καλείται να αποσταλεί ή να ληφθεί από και προς τον ζυγό 11.

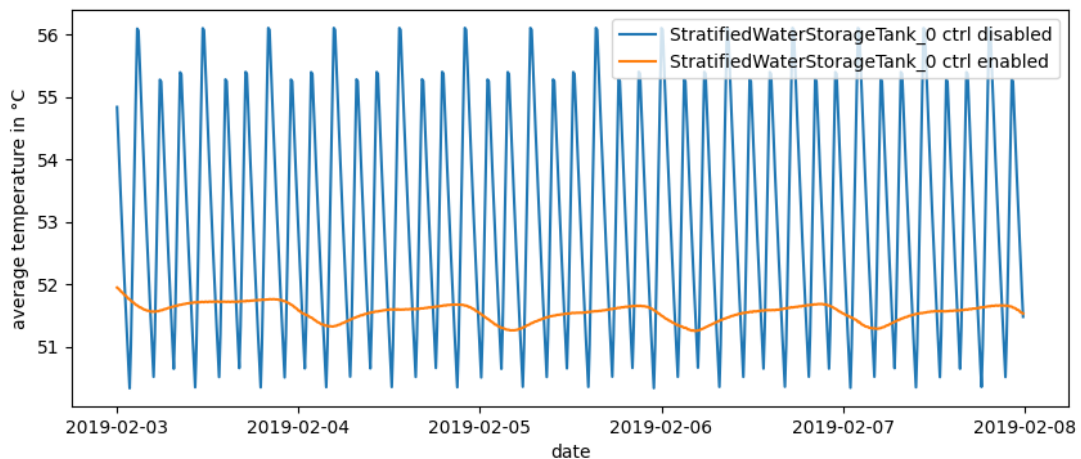


Γράφημα 17: Η φόρτωση της γραμμής των ζυγών 9-10 (σενάριο 3)



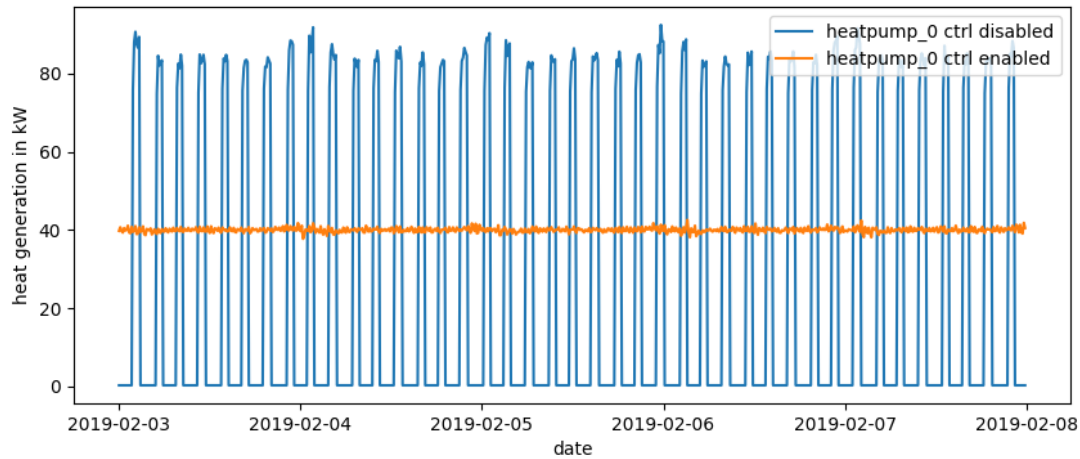
Γράφημα 18: Η φόρτωση της γραμμής των ζυγών 10-11 (σενάριο 3)

Στα τρία σενάρια η επίδραση στο θερμικό δίκτυο είναι ίδια και παρουσιάζεται στο Γράφημα 19, όπου απεικονίζεται η εξέλιξη της μέσης θερμοκρασίας της δεξαμενής με και χωρίς ενεργοποιημένο τον ελεγκτή τάσης. Με τον ελεγκτή τάσης απενεργοποιημένο κάθε φορά που η θερμοκρασία της δεξαμενής φτάνει το κατώτερο όριο, ο ελεγκτής θερμότητας σταματά την εκφόρτιση της δεξαμενής προς τη γραμμή παροχής και ενεργοποιεί την αντλία θερμότητας (λειτουργία 2 στην Εικόνα 6). Όταν η θερμοκρασία φτάσει στο ανώτερο όριο, ο ελεγκτής θερμότητας συνεχίζει την εκφόρτιση της δεξαμενής και απενεργοποιεί την αντλία θερμότητας (λειτουργία 5 στην Εικόνα 6). Ωστόσο με τον ελεγκτή τάσης ενεργοποιημένο η λειτουργία της δεξαμενής και της αντλίας γίνεται πιο σύνθετη (λειτουργία 6 στην Εικόνα 6), με αποτέλεσμα η δεξαμενή να εκφορτίζεται και να βοηθάει στην παροχή και να θερμαίνεται ταυτόχρονα από την αντλία θερμότητας, η οποία λειτουργεί με μερικό φορτίο.



Γράφημα 19: Μέση τιμή θερμοκρασίας της δεξαμενής (σενάρια 1,2,3)

Στο Γράφημα 20 και των τριών σεναρίων στην περίπτωση χωρίς έλεγχο τάσης, η αντλία θερμότητας λειτουργεί με τη μέγιστη κατανάλωση ισχύος (100 kWel) όταν ενεργοποιείται από τον ελεγκτή θερμότητας. Ενώ στην περίπτωση με τον έλεγχο τάσης η κατανάλωση έχει μειωθεί αισθητά.

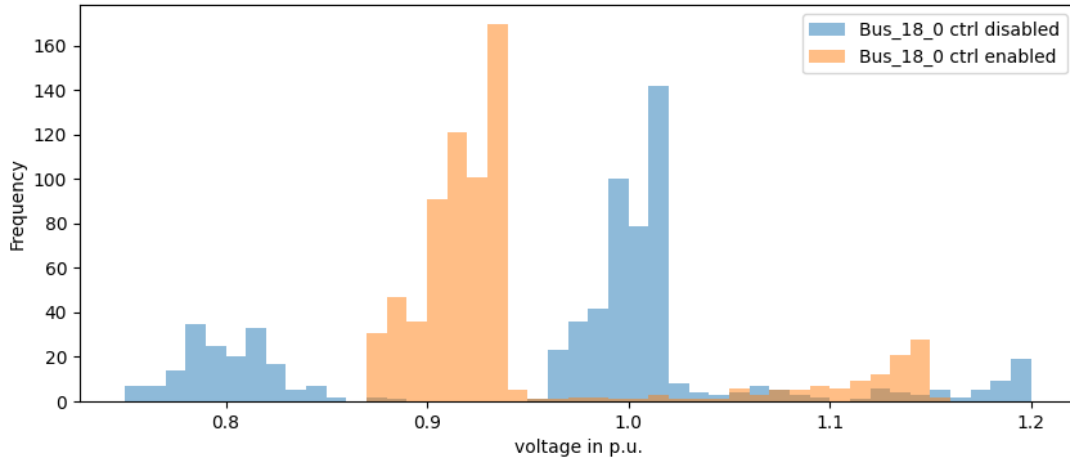


Γράφημα 20: Κατανάλωση ισχύος της αντλίας θερμότητας (σενάρια 1,2,3)

Στη συνέχεια της μελέτης, εξετάστηκε η επέκταση του σεναρίου 1, με την αντλία θερμότητας και τον ελεγκτή τάσης τοποθετημένα στον ζυγό 18, και διερευνήθηκε στα σενάρια 4 και 5 πώς η αύξηση της παραγωγής από τα φωτοβολταϊκά επηρεάζει το συνολικό σύστημα. Στο σενάριο 4, η παραγωγή των φωτοβολταϊκών αυξήθηκε κατά 50% σε σύγκριση με το σενάριο 1, ενώ στο σενάριο 5 η παραγωγή διπλασιάστηκε. Αυτή η ανάλυση είχε ως στόχο να εκτιμηθεί η επίδραση της αυξημένης παραγωγής ανανεώσιμης ενέργειας στην απόδοση του πολυενεργειακού συστήματος, ειδικά όσον αφορά τη σταθερότητα της τάσης και τη λειτουργία της αντλίας θερμότητας.

Στο Γράφημα 21 του σεναρίου 4 με την αυξημένη παραγωγή από το φωτοβολταϊκά, παρατηρούμε ότι η τάση αυξάνεται σημαντικά σε σχέση με το Γράφημα 1. Όταν ο ελεγκτής τάσης είναι απενεργοποιημένος, υπάρχουν πλέον περισσότερες τιμές τάσης πάνω από το 1.0 p.u., ενώ φτάνουν μέχρι και το 1.2 p.u.. Όταν ο ελεγκτής τάσης είναι ενεργοποιημένος, η τάση παραμένει πιο σταθερή, οι οποίες κυμαίνονται κυρίως γύρω από το 0.9-1.0 p.u., αλλά παρατηρούμε και κάποιες τιμές κοντά στο 1.1 p.u., υποδεικνύοντας ότι η αυξημένη παραγωγή από τα φωτοβολταϊκά επηρεάζει τον έλεγχο της τάσης.

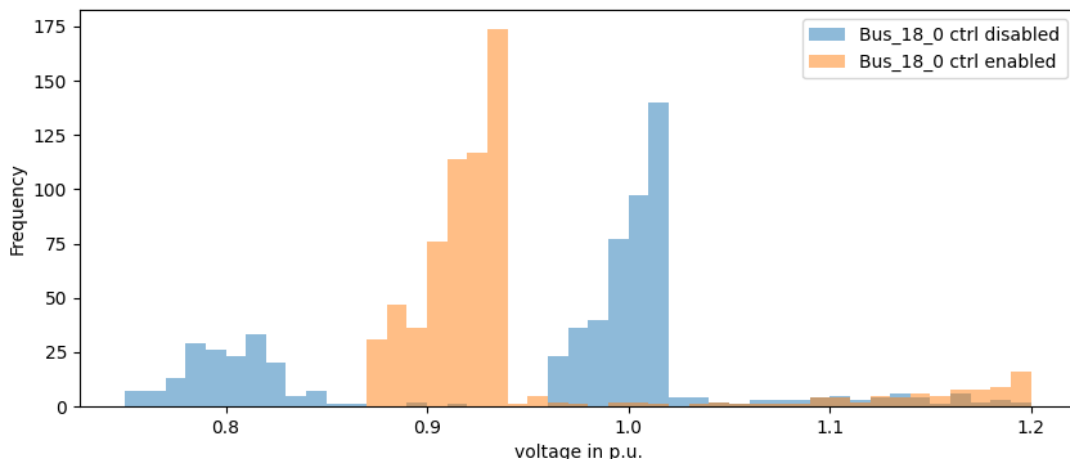
Ο ελεγκτής τάσης προσπαθεί να διατηρήσει την τάση σταθερή, ειδικά όταν η παραγωγή από τα PV αυξάνεται. Επιπλέον καταφέρνει να περιορίσει την τάση κοντά στο 0.9-1.0 p.u., αποτρέποντας τις υπερβολικές αυξήσεις. Παρόλα αυτά, ακόμη και με τον ελεγκτή ενεργοποιημένο, υπάρχει μια μικρή αύξηση της τάσης όταν η παραγωγή των φωτοβολταϊκών είναι υψηλή, κάτι που δείχνει ότι η αντλία θερμότητας δεν μπορεί να απορροφήσει όλη την ισχύ που παράγεται.



Γράφημα 21: Η τάση του ζυγού 18 σε p.u. (σενάριο 4)

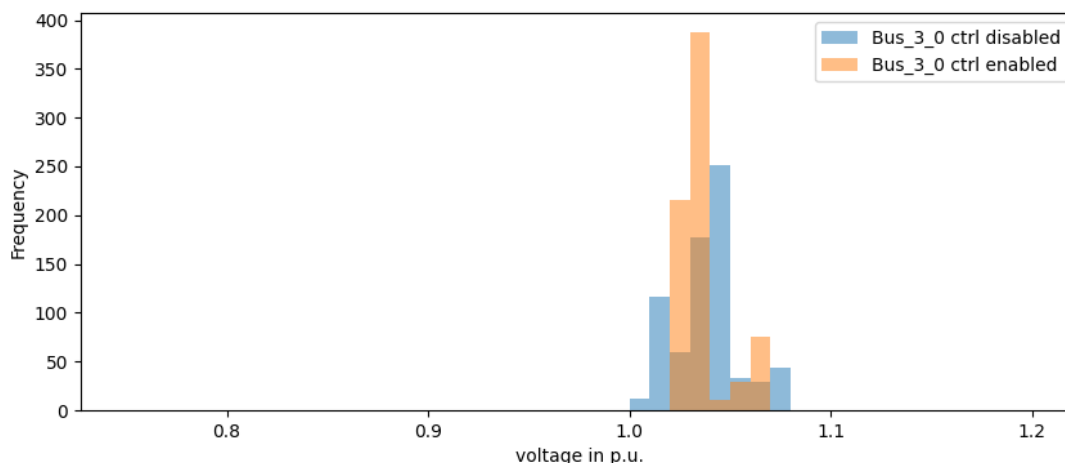
Παρομοίως στο Γράφημα 22 παρατηρείται η επίδραση του διπλασιασμού της παραγωγής από τα φωτοβολταϊκά στον ζυγό 18 σε σχέση με το σενάριο 1. Όταν ο ελεγκτής τάσης είναι απενεργοποιημένος παρατηρούμε ότι η τάση συγκεντρώνεται κυρίως γύρω από το 1.0 p.u.. Υπάρχουν λίγες περιπτώσεις υψηλότερων τάσεων, με μερικές τιμές να αγγίζουν ή να ξεπερνούν ελαφρώς το 1.1 p.u. και με μερικές να παίρνουν κάτω από 0.8 p.u.. Η τάση παραμένει πιο σταθερή γύρω από το 1.0 p.u., με τη μεγάλη πλειοψηφία των τιμών να βρίσκεται κοντά σε αυτό το σημείο. Όταν ο ελεγκτής τάσης είναι ενεργοποιημένος, παρατηρούμε μεγαλύτερη διακύμανση στις τάσεις, με τιμές να κατανέμονται από 0.9 p.u. έως και πάνω από 1.2 p.u. Οι χαμηλότερες τιμές τάσης γύρω από το 0.9 p.u. είναι πολύ συχνές, γεγονός που δείχνει ότι ο ελεγκτής προσπαθεί να διατηρήσει την τάση σταθερή σε χαμηλότερα επίπεδα.

Παρά την προσπάθεια του ελεγκτή, υπάρχουν αρκετές περιπτώσεις όπου η τάση ανεβαίνει πάνω από το 1.1 p.u., κάτι που υποδηλώνει ότι η παραγωγή από τα φωτοβολταϊκά έχει αυξηθεί σημαντικά και είναι δύσκολο να ελεγχθεί πλήρως. Ο διπλασιασμός της παραγωγής από τα φωτοβολταϊκά στον ζυγό 18 προκαλεί σημαντική αύξηση της τάσης, ειδικά όταν ο ελεγκτής τάσης είναι απενεργοποιημένος. Ο ελεγκτής τάσης βοηθά στη σταθεροποίηση της τάσης, αλλά η παραγωγή των φωτοβολταϊκών φαίνεται να είναι τόσο υψηλή που οι παραβάσεις της τάσης δεν μπορούν να αποφευχθούν εντελώς.

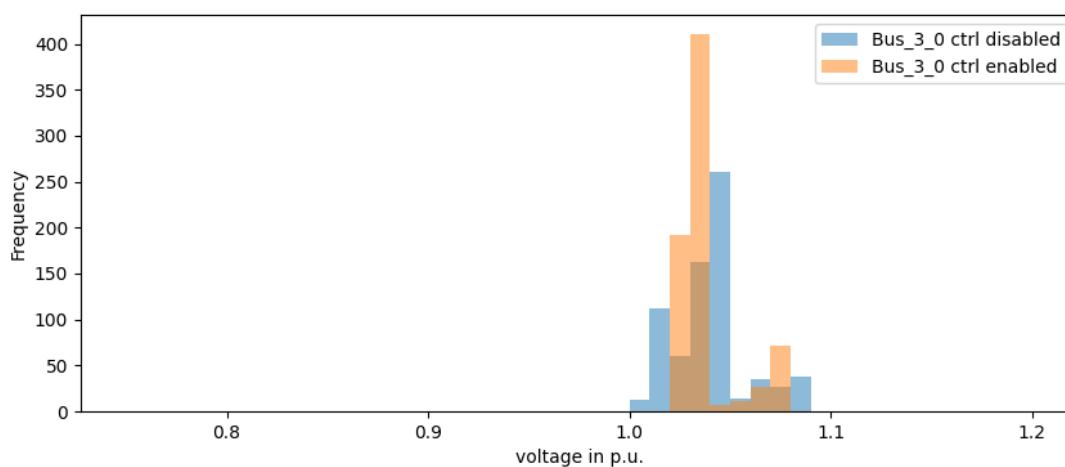


Γράφημα 22: Η τάση του ζυγού 18 σε p.u. (σενάριο 5)

Στο Γράφημα 23 η τάση στον ζυγό 3 είναι παρόμοια με τα γραφήματα των τριών πρώτων σεναρίων αλλά στο Γράφημα 24 παρατηρούμε ότι η τάση ανέβηκε λίγο παραπάνω πιο κοντά στο 1.1 p.u. Η άνοδος της τάσης στον ζυγό που βρίσκεται πιο κοντά στον εξωτερικό ζυγό οφείλεται κυρίως στην αύξηση της τοπικής παραγωγής από τα φωτοβολταϊκά. Παρόλο που κάποια από αυτήν την αύξηση έχει απορροφήσει η αντλία θερμότητας αλλά είναι τόσο μεγάλη η παραγωγή που προκαλεί ροή ισχύος προς το εξωτερικό δίκτυο, κάτι που μπορεί να προκαλέσει τοπική αύξηση της τάσης στους ζυγούς, ακόμα και σε εκείνους που βρίσκονται κοντά στον εξωτερικό ζυγό.



Γράφημα 23: Η τάση του ζυγού 3 σε p.u. (σενάριο 4)

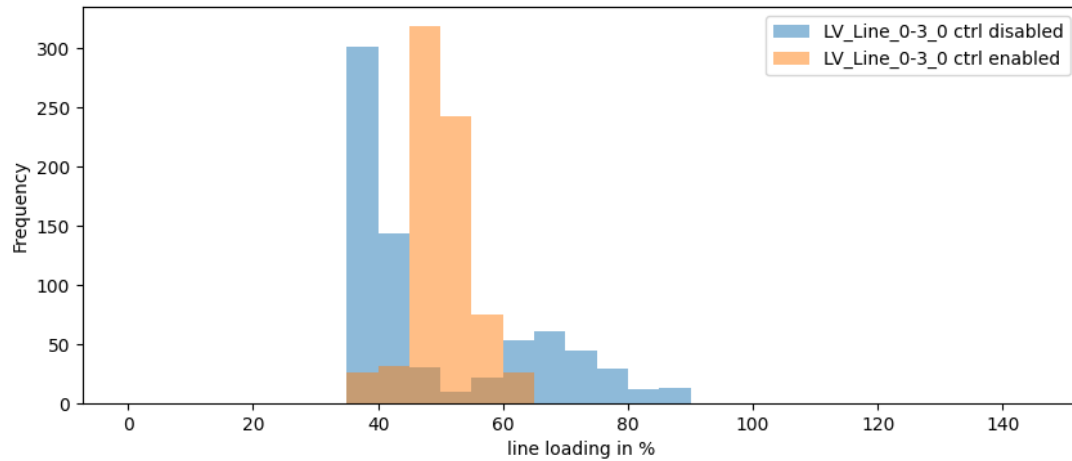


Γράφημα 24: Η τάση του ζυγού 3 σε p.u. (σενάριο 5)

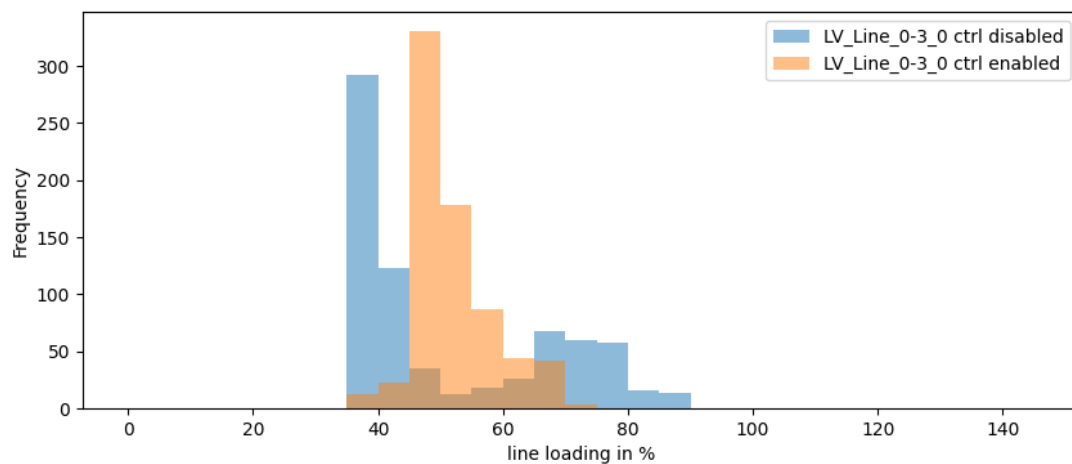
Στο Γράφημα 25 του σεναρίου 4 παρατηρείται ότι η φόρτωση της γραμμής βρίσκεται κυρίως μεταξύ 50% και 60% όταν ο ελεγκτής τάσης είναι ενεργοποιημένος. Όταν ο ελεγκτής τάσης είναι απενεργοποιημένος, η φόρτωση συγκεντρώνεται κυρίως κοντά στο 40-50%, ωστόσο υπάρχουν κάποιες περιπτώσεις υψηλότερης φόρτωσης, που ξεπερνούν το 80%..

Στο Γράφημα 26 του σεναρίου 5 όπου η παραγωγή από τα PV έχει διπλασιαστεί, η φόρτωση της γραμμής αυξάνεται ελαφρώς. Η κατανομή της φόρτωσης όταν ο ελεγκτής είναι ενεργοποιημένος παραμένει παρόμοια, συγκεντρωμένη κυρίως μεταξύ 40% και 60%, αλλά υπάρχουν ελαφρώς περισσότερες περιπτώσεις όπου η φόρτωση φτάνει ή ξεπερνά το 60-70%. Όταν ο ελεγκτής τάσης είναι απενεργοποιημένος, η φόρτωση παραμένει πάλι κοντά στο 40%, αλλά η συχνότητα εμφάνισης μεγαλύτερων φορτίων αυξάνονται με την φόρτωση της γραμμής να ξεπερνά το 80%.



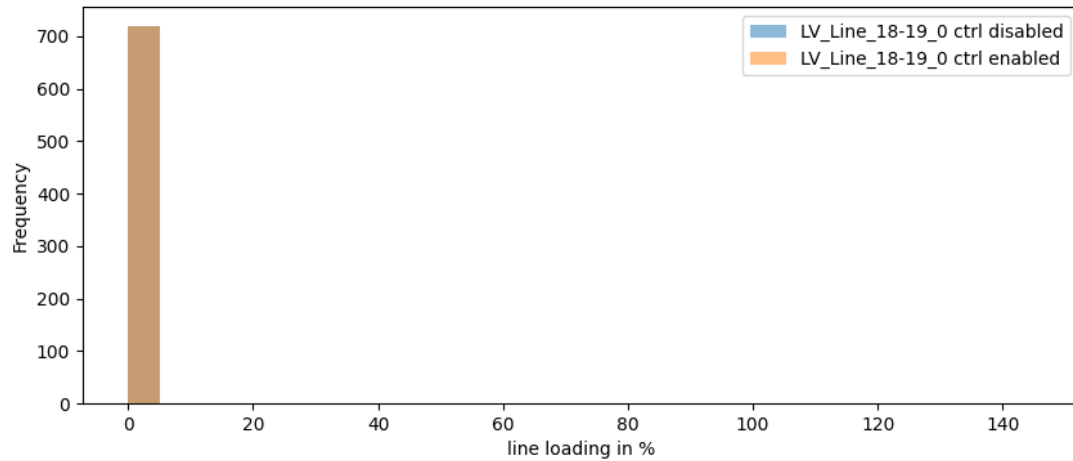


Γράφημα 25: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 4)



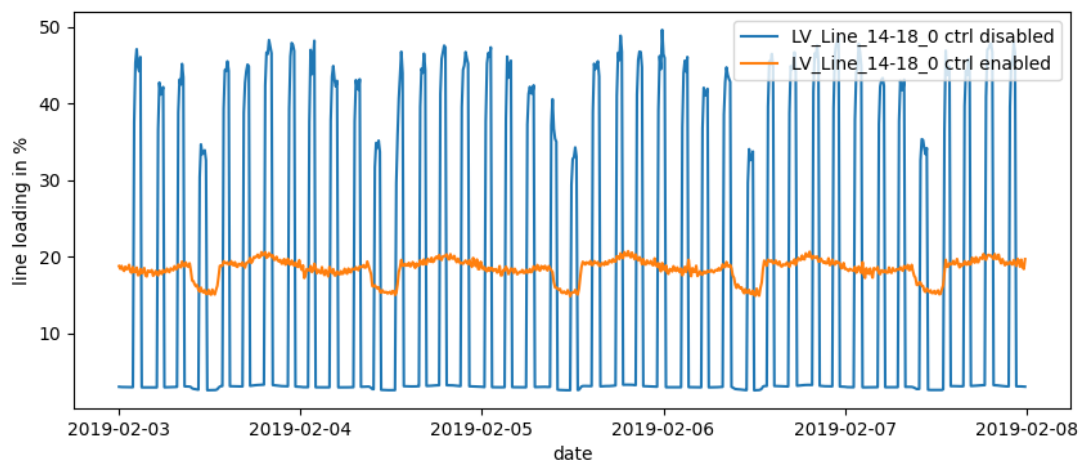
Γράφημα 26: Η φόρτωση της γραμμής των ζυγών 0-3 (σενάριο 5)

Στο Γράφημα 27 παρατηρούμε ότι η φόρτωση της γραμμής που συνδέει το ζυγό 18 με τον τελείως ακριανό ζυγό 19 και στα δύο τα σενάρια είναι κοντά στο μηδέν. Σύμφωνα και με το προηγούμενα γραφήματα κάτι τέτοιο συμβαίνει καθώς η γραμμή που συνδέει τον ζυγό 18 με τον ζυγό 19 δεν φαίνεται να μεταφέρει σημαντική ισχύ, όπως υποδεικνύεται από τη μηδενική φόρτωση, τόσο στο σενάριο 4 όσο και στο σενάριο 5. Παρά την αύξηση της παραγωγής των φωτοβολταϊκών, δεν υπάρχει καμία ένδειξη αύξησης της φόρτωσης, γεγονός που υποδηλώνει είτε μηδενική κατανάλωση στον ζυγό 19 είτε μηδενική ροή ισχύος μέσω της συγκεκριμένης γραμμής.

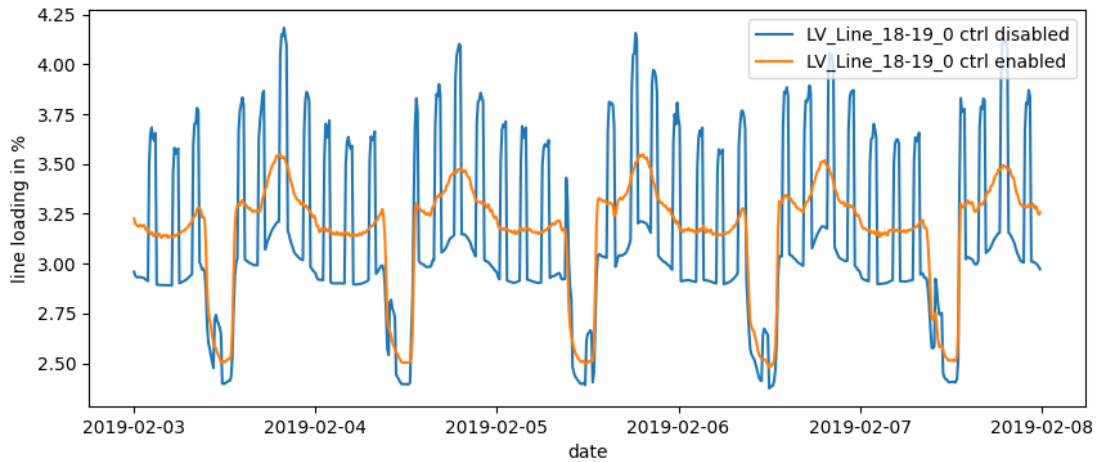


Γράφημα 27: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 4,5)

Η φόρτωση της γραμμής γύρω από τον ζυγό 18 στο σενάριο 4 και 5, όπως φαίνεται στο Γράφημα 28 και Γράφημα 29, δεν παρουσιάζει σημαντική αλλαγή παρά την αύξηση της παραγωγής από τα φωτοβολταϊκά. Ο ελεγκτής τάσης που είναι ενεργοποιημένος φαίνεται να διατηρεί τη φόρτωση της γραμμής σε σταθερά επίπεδα, παρόλο που η παραγωγή από τα PV αυξάνεται.



Γράφημα 28: Η φόρτωση της γραμμής των ζυγών 14-18 (σενάριο 4,5)



Γράφημα 29: Η φόρτωση της γραμμής των ζυγών 18-19 (σενάριο 4,5)

Επίσης και στα σενάρια 4,5 όπως και στα τρία πρώτα η μέση θερμοκρασία της δεξαμενής είναι ακριβώς η ίδια για όλα τα σενάρια σύμφωνα με το Γράφημα 19. Σε όλα τα πειράματα η λειτουργία με ενεργοποιημένο τον ελεγκτή τάσης είναι κατά κύριο λόγο στην λειτουργία 6 που σημαίνει ότι η δεξαμενή να εκφορτίζεται και να βοηθάει στην παροχή και να θερμαίνεται ταυτόχρονα από την αντλία θερμότητας. Έτσι η τοπική κατανάλωση ισχύος αυξάνεται ταυτόχρονα με την παραγωγή των φωτοβολταϊκών, τότε η παραγόμενη ενέργεια καταναλώνεται άμεσα και δεν αποθηκεύεται, με αποτέλεσμα να μην παρατηρείται αλλαγή στη μέση τιμή της δεξαμενής.

Τα αποτελέσματα από τα σενάρια προσομοίωσης ανέδειξαν τη σημασία της συνεργασίας μεταξύ του ηλεκτρικού και θερμικού δικτύου για την αποτελεσματική διαχείριση της ενέργειας και τη σταθεροποίηση της τάσης. Στο σενάριο 1, παρατηρήθηκε ότι χωρίς τον ελεγκτή τάσης, η τάση στον ζυγό 18 έπεφτε κάτω από τα αποδεκτά όρια, ενώ με την ενεργοποίηση του ελεγκτή σταθεροποιήθηκε στο 0,95 p.u. Αντίστοιχα, στα σενάρια 2 και 3, η χρήση της αντλίας θερμότητας συνέβαλε στη μείωση της υπερφόρτωσης στις γραμμές, καθώς η επιπλέον ενέργεια απορροφήθηκε από το θερμικό δίκτυο. Στα σενάρια 4 και 5, η αποδοτική συνεργασία μεταξύ της αντλίας θερμότητας και της δεξαμενής αποθήκευσης θερμότητας επέτρεψε τη μείωση της φόρτωσης σε κρίσιμες γραμμές και τη σταθεροποίηση της τάσης, ενισχύοντας τη συνολική αποδοτικότητα του συστήματος και προσφέροντας ευστάθεια, ειδικά σε περιόδους αυξημένης παραγωγής από ανανεώσιμες πηγές.

## **Κεφάλαιο 7: Συμπεράσματα**

---

Η παρούσα διπλωματική εργασία διερεύνησε τη δυνατότητα σύνδεσης των ηλεκτρικών δικτύων με δίκτυα τηλεθέρμανσης μέσω της χρήσης αντλιών θερμότητας και της εφαρμογής σύγχρονων μεθόδων ελέγχου τάσης και θερμικού ελέγχου. Σκοπός ήταν να μελετηθεί πώς αυτή η διασύνδεση μπορεί να προσφέρει επικουρικές υπηρεσίες στα ηλεκτρικά δίκτυα και να ενισχύσει τη σταθερότητα και την αποδοτικότητα των σύγχρονων ενεργειακών συστημάτων, ειδικά σε περιβάλλοντα με αυξημένη διείσδυση ανανεώσιμων πηγών ενέργειας. Τέλος, προτείνονται πιθανές βελτιώσεις και μελλοντικές κατευθύνσεις.

### **7.1 Συμπεράσματα της Μελέτης**

Η παρούσα διπλωματική εργασία επικεντρώνεται στη μοντελοποίηση ενός πολυενεργειακού συστήματος, το οποίο συνδυάζει το ηλεκτρικό και το θερμικό δίκτυο, διερευνώντας πώς το δίκτυο τηλεθέρμανσης μπορεί να συνεισφέρει στην ενίσχυση του ηλεκτρικού δικτύου, ενώ ταυτόχρονα μεγιστοποιείται η απορρόφηση ενέργειας από ανανεώσιμες πηγές ενέργειας. Η μοντελοποίηση υλοποιείται μέσω της γλώσσας προγραμματισμού Python, με χρήση των βιβλιοθηκών pandarpower και pandaripes, για το ηλεκτρικό και θερμικό δίκτυο αντίστοιχα. Η διασύνδεση των δύο δικτύων πραγματοποιείται μέσω αντλίας θερμότητας, η οποία μετατρέπει την ηλεκτρική ενέργεια σε θερμική και αντιμετωπίζεται στο ηλεκτρικό δίκτυο ως ωμικό φορτίο, πάνω στο οποίο ενσωματώνονται δύο ελεγκτές. Ο βασικός στόχος της διασύνδεσης είναι η διατήρηση της τάσης εντός του εύρους 0.9 – 1.1 p.u.. Τα κύρια συμπεράσματα της μελέτης συνοψίζονται ως εξής:

#### **7.1.1 Αποτελεσματικότητα της Αντλίας Θερμότητας**

Η χρήση της αντλίας θερμότητας για τη μετατροπή της πλεονάζουσας ηλεκτρικής ενέργειας σε θερμική ενέργεια αποδείχθηκε εξαιρετικά αποδοτική. Τα πλεονεκτήματα αυτής της λύσης έγκεινται στην ικανότητα αποθήκευσης της θερμικής ενέργειας στη δεξαμενή, η οποία στη συνέχεια μπορεί να χρησιμοποιηθεί για την κάλυψη θερμικών αναγκών σε μελλοντικές χρονικές στιγμές, βελτιώνοντας τη συνολική απόδοση του συστήματος. Σύμφωνα με τα αποτελέσματα η ενσωμάτωση της αντλίας θερμότητας και του ελέγχου τάσης μπορεί να βελτιώσει σημαντικά τη σταθερότητα του δικτύου, ειδικά στις περιπτώσεις όπου η ζήτηση ηλεκτρικής ενέργειας είναι μεταβλητή λόγω της παρουσίας ΑΠΕ. Η δυνατότητα αποθήκευσης ενέργειας σε θερμικές δεξαμενές και η σταδιακή απελευθέρωσή της για τη θέρμανση επέτρεψε τη μείωση των διακυμάνσεων της τάσης, καθώς το δίκτυο ήταν σε θέση να διαχειριστεί καλύτερα τις

αυξομειώσεις στην παραγωγή. Αυτός ο μηχανισμός αποθήκευσης μειώνει σημαντικά την ανάγκη για περικοπή της παραγωγής ηλεκτρικής ενέργειας σε περιόδους χαμηλής ζήτησης και συμβάλλει στην εξομάλυνση των διακυμάνσεων που προκαλούνται από τις ΑΠΕ.

### **7.1.2 Έλεγχος Τάσης και Θερμικός Έλεγχος**

Στα σενάρια που εξετάστηκαν, ο ελεγκτής τάσης έδειξε θετική επίδραση στη σταθερότητα του συστήματος, διατηρώντας την τάση του δικτύου εντός επιτρεπτών ορίων. Ο συνδυασμός του ελεγκτή τάσης και θερμότητας απέδειξε τη δυνατότητα αποτελεσματικής διαχείρισης της αλληλεξάρτησης των δύο δικτύων. Οι προσομοιώσεις έδειξαν ότι ο ελεγκτής τάσης συνέβαλε στη διατήρηση της τάσης σε επιτρεπτά όρια, ακόμη και σε περιόδους αιχμής, όπου η ζήτηση ενέργειας υπερέβαινε την παραγωγή και στο αντίθετο όταν η παραγωγή υπερέβαινε της ζήτησης. Η αλληλεπίδραση μεταξύ των δύο ελεγκτών οδήγησε σε σταθερή λειτουργία του δικτύου, με μειωμένα προβλήματα υπέρτασης, υπότασης και υπερφόρτωσης γραμμών.

### **7.1.3 Συμβολή στη Σταθερότητα του Δικτύου**

Το σύστημα που αναπτύχθηκε με την ενσωμάτωση αντλίας θερμότητας και την αποθήκευση θερμικής ενέργειας αποδείχθηκε αποτελεσματικό στη βελτίωση της σταθερότητας του ηλεκτρικού δικτύου. Τα αποτελέσματα των προσομοιώσεων κατέδειξαν ότι το δίκτυο τηλεθέρμανσης μπορεί να λειτουργήσει ως μηχανισμός υποστήριξης για το ηλεκτρικό δίκτυο, απορροφώντας την πλεονάζουσα ηλεκτρική ενέργεια και αποτρέποντας διακοπές στην παραγωγή των ΑΠΕ. Συγκεκριμένα η φόρτωση των γραμμών που βρίσκονται κοντά στους ζυγούς όπου έγινε η προσομοίωση επίσης βελτιώθηκε με τη χρήση του ελεγκτή τάσης. Στα σενάρια αυτά, η χρήση της αντλίας θερμότητας χωρίς έλεγχο προκαλούσε υπερφόρτωση των γραμμών, κυρίως λόγω της μεταφοράς ενεργού ισχύος. Η ενεργοποίηση του ελεγκτή βοήθησε στη μείωση της φόρτισης των γραμμών, αποτρέποντας τις υπερτάσεις. Αυτό το ενσωματωμένο σύστημα σταθεροποίησε το σύστημα και μείωσε τις αστάθειες.

### **7.1.4 Βελτιστοποίηση Χρήσης Ανανεώσιμων Πηγών Ενέργειας**

Ένα από τα βασικά πλεονεκτήματα του προτεινόμενου συστήματος είναι η βελτίωση της απορρόφησης ενέργειας από ανανεώσιμες πηγές. Χάρη στην ευελιξία που προσέφερε το σύστημα αποθήκευσης θερμότητας, η ενέργεια που παράγεται από ΑΠΕ μπορεί να χρησιμοποιηθεί σε περιόδους χαμηλής ζήτησης, αντί να χάνεται ή να περιορίζεται. Αυτή η απορρόφηση των ανανεώσιμων πηγών συνεισφέρει στη συνολική μείωση της εξάρτησης από συμβατικές πηγές ενέργειας και προωθεί τη βιωσιμότητα του δικτύου. Ωστόσο θα πρέπει με ένα όριο η απορρόφηση ενέργειας από ΑΠΕ καθώς σύμφωνα με τα αποτελέσματα η αρκετά μεγάλη παραγωγή οδηγεί σε αστάθειες σύστημα.

## **7.2 Μελλοντική Εργασία**

Ο στόχος αυτής της διπλωματικής εργασίας ήταν η ρύθμιση της τάσης σε ένα ηλεκτρικό δίκτυο μέσω της διασύνδεσής του με ένα δίκτυο τηλεθέρμανσης. Η ανάλυση έδειξε πώς η μεταφορά ισχύος μεταξύ των δύο δικτύων είναι επωφελής και για τα δύο. Το δίκτυο τηλεθέρμανσης, λόγω της φύσης του, είναι δύσκολο να μελετηθεί μεμονωμένα, και ακόμη πιο απαιτητικό όταν εξετάζεται σε συνδυασμό με άλλα δίκτυα. Παρά τα θετικά αποτελέσματα της μελέτης, υπάρχουν ορισμένοι περιορισμοί και τομείς που χρήζουν περαιτέρω διερεύνησης.

Ένας από τους κύριους περιορισμούς είναι η δυσκολία στη μελέτη του δικτύου τηλεθέρμανσης όταν εξετάζεται σε συνδυασμό με άλλα δίκτυα. Αν και το σύστημα πέτυχε στη διαχείριση της ροής θερμότητας, υπήρξαν δυσκολίες στον υπολογισμό της κατανομής πίεσης όταν περισσότερες από μία πηγές, όπως το εξωτερικό θερμικό δίκτυο και η δεξαμενή αποθήκευσης, τροφοδοτούν τη γραμμή του δικτύου. Η βελτίωση του υπολογισμού της πίεσης αποτελεί ένα σημαντικό βήμα για την αποδοτική λειτουργία του δικτύου και προτείνεται ως κρίσιμος τομέας για μελλοντική έρευνα.

Επιπλέον, μια πιθανή βελτίωση θα μπορούσε να αφορά τη μεταφορά της αντλίας θερμότητας σε άλλο σημείο του δικτύου τηλεθέρμανσης. Η τοποθέτησή της σε κρίσιμους κόμβους, καθώς και η προσθήκη δεύτερης αντλίας θερμότητας, θα ενίσχυαν τη σταθερότητα και τη δυνατότητα υποστήριξης απομακρυσμένων τμημάτων του δικτύου. Αυτή η προσθήκη απαιτεί τροποποιήσεις στον κώδικα, ώστε να επιτραπεί η σύνδεση και λειτουργία πολλαπλών αντλιών στο δίκτυο, κάτι που θα ενίσχυε περαιτέρω την ευελιξία και την αποδοτικότητα του συστήματος.

Η μελλοντική έρευνα θα μπορούσε επίσης να επικεντρωθεί σε πιο προχωρημένα μοντέλα ελέγχου, που θα ενσωματώνουν τη βελτίωση της υδραυλικής συμπεριφοράς και θα βελτιώνουν τη συνολική απόδοση τόσο του ηλεκτρικού όσο και του θερμικού δικτύου. Επίσης, η ενσωμάτωση πιο ακριβών αλγορίθμων για την πρόβλεψη της ζήτησης ενέργειας και της πίεσης στο δίκτυο θα μπορούσε να συμβάλει στη βελτιστοποίηση της λειτουργίας των συστημάτων.

Τέλος, η προσομοίωση αυτών των προσεγγίσεων σε δίκτυα μεγαλύτερης κλίμακας, καθώς και η ανάλυση δυναμικών συνθηκών, θα επιτρέψει την καλύτερη κατανόηση της συμπεριφοράς των συστημάτων σε πραγματικές συνθήκες και θα αποκαλύψει επιπλέον προκλήσεις και λύσεις.

## ***Παράρτημα – Κώδικας Εφαρμογής***

---

Ο κώδικας που χρησιμοποιήθηκε πριν την παραμετροποίηση του βρίσκεται στο [17].

## Βιβλιογραφία

---

- [1] 'IEA – International Energy Agency', IEA. Ημερομηνία πρόσβασης: 14 Σεπτέμβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://www.iea.org>
- [2] *COMMUNICATION FROM THE COMMISSION TO THE EUROPEAN PARLIAMENT, THE EUROPEAN COUNCIL, THE COUNCIL, THE EUROPEAN ECONOMIC AND SOCIAL COMMITTEE AND THE COMMITTEE OF THE REGIONS The European Green Deal*. 2019. Ημερομηνία πρόσβασης: 14 Σεπτέμβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52019DC0640&qid=1726319107234>
- [3] *ΑΝΑΚΟΙΝΩΣΗ ΤΗΣ ΕΠΙΤΡΟΠΗΣ ΠΡΟΣ ΤΟ ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΟΒΟΥΛΙΟ, ΤΟ ΣΥΜΒΟΥΛΙΟ, ΤΗΝ ΕΥΡΩΠΑΪΚΗ ΟΙΚΟΝΟΜΙΚΗ ΚΑΙ ΚΟΙΝΩΝΙΚΗ ΕΠΙΤΡΟΠΗ ΚΑΙ ΤΗΝ ΕΠΙΤΡΟΠΗ ΤΩΝ ΠΕΡΙΦΕΡΕΙΩΝ «Προσαρμογή στον στόχο του 55 %»: υλοποίηση του στόχου της ΕΕ για το κλίμα με ορίζοντα το 2030 στην πορεία προς την κλιματική ουδετερότητα*. 2021. Ημερομηνία πρόσβασης: 14 Σεπτέμβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://eur-lex.europa.eu/legal-content/EL/TXT/?uri=CELEX%3A52021DC0550>
- [4] 'Renewables 2021 - Analysis and forecast to 2026', 2026.
- [5] G. Notton κ.ά., 'Intermittent and stochastic character of renewable energy sources: Consequences, cost of intermittence and benefit of forecasting', *Renewable and Sustainable Energy Reviews*, τ. 87, σελ. 96–105, Μαΐου 2018, doi: 10.1016/j.rser.2018.02.007.
- [6] X. Li, D. Hui, και X. Lai, 'Battery Energy Storage Station (BESS)-Based Smoothing Control of Photovoltaic (PV) and Wind Power Generation Fluctuations', *IEEE Transactions on Sustainable Energy*, τ. 4, τχ. 2, σελ. 464–473, Απριλίου 2013, doi: 10.1109/TSTE.2013.2247428.
- [7] *Voltage Characteristics of Electricity Supplied by Public Distribution Networks, BS EN Standard 50160*, 2010.
- [8] G. C. Kryonidis, E. O. Kontis, A. I. Chrysochos, C. S. Demoulias, και G. K. Papagiannis, 'A Coordinated Droop Control Strategy for Overvoltage Mitigation in Active Distribution Networks', *IEEE Transactions on Smart Grid*, τ. 9, τχ. 5, σελ. 5260–5270, Σεπτεμβρίου 2018, doi: 10.1109/TSG.2017.2685686.
- [9] P. Mancarella, 'MES (multi-energy systems): An overview of concepts and evaluation models', *Energy*, τ. 65, σελ. 1–17, Φεβρουαρίου 2014, doi: 10.1016/j.energy.2013.10.041.
- [10] S. Xie, Z. Hu, J. Wang, και Y. Chen, 'The optimal planning of smart multi-energy systems incorporating transportation, natural gas and active distribution networks', *Applied Energy*, τ. 269, σελ. 115006, Ιουλίου 2020, doi: 10.1016/j.apenergy.2020.115006.
- [11] S. Klyarovskiy, S. You, H. Cai, και H. W. Bindner, 'Integrated Planning of a Large-Scale Heat Pump in View of Heat and Power Networks', *IEEE Transactions on Industry Applications*, τ. 55, τχ. 1, σελ. 5–15, Ιανουαρίου 2019, doi: 10.1109/TIA.2018.2864114.



- [12] M. Geidl και G. Andersson, 'Optimal Power Flow of Multiple Energy Carriers', *IEEE Transactions on Power Systems*, τ. 22, τχ. 1, σελ. 145–155, Οκτωβρίου 2007, doi: 10.1109/TPWRS.2006.888988.
- [13] H. Lund, B. Möller, B. V. Mathiesen, και A. Dyrelund, 'The role of district heating in future renewable energy systems', *Energy*, τ. 35, τχ. 3, σελ. 1381–1390, Μαρτίου 2010, doi: 10.1016/j.energy.2009.11.023.
- [14] X. Liu, J. Wu, N. Jenkins, και A. Bagdanavicius, 'Combined analysis of electricity and heat networks', *Applied Energy*, τ. 162, σελ. 1238–1250, Ιανουαρίου 2016, doi: 10.1016/j.apenergy.2015.01.102.
- [15] F. García-Muñoz, A. F. C. Burray, και A. González-Garrido, 'Modeling Local Energy Markets Transactions for Multi-Energy Systems in Low-Voltage Unbalanced Distribution Networks', στο *2023 IEEE International Smart Cities Conference (ISC2)*, Σεπτεμβρίου 2023, σελ. 1–7. doi: 10.1109/ISC257844.2023.10293331.
- [16] J. E. Feghali κ.ά., 'Electrical Grid Flexibility via Heat Pump and Thermal Storage Control', *IFAC-PapersOnLine*, τ. 55, τχ. 11, σελ. 84–89, Ιανουαρίου 2022, doi: 10.1016/j.ifacol.2022.08.053.
- [17] *ERIGrid2/benchmark-model-multi-energy-networks*. (17 Σεπτέμβριος 2024). Python. ERIGrid 2.0. Ημερομηνία πρόσβασης: 3 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://github.com/ERIGrid2/benchmark-model-multi-energy-networks>
- [18] S. Bennett, 'Development of the PID controller', *IEEE Control Systems Magazine*, τ. 13, τχ. 6, σελ. 58–62, Σεπτεμβρίου 1993, doi: 10.1109/37.248006.
- [19] 'Getting started with Navigator — Anaconda documentation'. Ημερομηνία πρόσβασης: 17 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://docs.anaconda.com/navigator/getting-started/>
- [20] 'Welcome to Spyder's Documentation — Spyder 5 documentation'. Ημερομηνία πρόσβασης: 17 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://docs.spyder-ide.org/current/index.html>
- [21] 'pandarpower — pandarpower 3.0.0 documentation'. Ημερομηνία πρόσβασης: 17 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://pandarpower.readthedocs.io/en/latest/>
- [22] 'pandarpipes — pandarpipes 0.10.0 documentation'. Ημερομηνία πρόσβασης: 17 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://pandarpipes.readthedocs.io/en/latest/>
- [23] 'Download Anaconda Distribution', Anaconda. Ημερομηνία πρόσβασης: 3 Οκτώβριος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://www.anaconda.com/download>

# **Συνομογραφίες - Αρκτικόλεξα - Ακρωνύμια**

---

MES

ΑΠΕ

ΕΕ

EriGrid

PV

Multi Energy System

Ανανεώσιμες Πηγές Ενέργειας

Ευρωπαϊκή Ένωση

European Research Infrastructure supporting

Smart Grid System

Photovoltaic