



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

**Λήψη αποφάσεων υπό αβεβαιότητα για συστήματα διανομής
ηλεκτρικής ενέργειας με prosumers**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πέτρος, Χ. Έλληνας

Επιβλέπων : Εμμανουήλ Βαρβαρίγος

Καθηγητής

Αθήνα, Φεβρουάριος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ



ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ

**Λήψη αποφάσεων υπό αβεβαιότητα για συστήματα διανομής
ηλεκτρικής ενέργειας με prosumers**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Πέτρος, Χ. Έλληνας

Επιβλέπων : Εμμανουήλ Βαρβαρίγος

Καθηγητής

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 3^η Φεβρουαρίου 2023

.....

Εμμανουήλ Βαρβαρίγος

Καθηγητής

.....

Θεοδώρα Βαρβαρίγου

Καθηγητής

.....

Συμεών Παπαβασιλείου

Καθηγητής

Αθήνα, Φεβρουάριος 2023

Πέτρος, Χ. Έλληνας

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Πέτρος, Χριστόδουλος Έλληνας, 2023.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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 δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

ΠΕΡΙΛΗΨΗ

Στις μέρες μας, τα συστήματα διανομής ενέργειας με υψηλή εισχώρηση κατανεμημένων πόρων παραγωγής αντιμετωπίζουν μεγάλη αβεβαιότητα. Έτσι το παραδοσιακό πρόβλημα βέλτιστης ροής φορτίου μετατρέπεται σε πρόβλημα σειριακής λήψης αποφάσεων υπό αβεβαιότητα. Σε αυτό το πλαίσιο, η λύση παίρνει την μορφή πολιτικής, για παράδειγμα, μια μέθοδος για αποφάσεις αποστολής έχοντας την κατάσταση συστήματος σε πραγματικό χρόνο. Η δικαιολόγηση των αποφάσεων και της αβεβαιότητας για το μέλλον γίνεται ιδιαίτερα δύσκολη καθώς γίνονται πιο πολλοί οι πόροι και έχοντας μικρό αριθμό δεδομένων για τις τυχαίες μεταβλητές.

Έτσι παρουσιάζουμε μια κατανεμημένη πολιτική βασισμένη στα δεδομένα, για λήψη αποφάσεων σε πραγματικό χρόνο. Η πολιτική αυτή εγγυάται την ικανοποίηση των περιορισμών συστήματος και αποδεικνύεται πειραματικά ότι έχει σχεδόν βέλτιστη επίδοση. Η πολιτική αυτή δοκιμάστηκε σε δύο είδη συστημάτων ηλεκτρικής ενέργειας, ενός συστήματος κόμβο με κόμβο και ενός συστήματος διανομής ηλεκτρικής ενέργειας γεμάτο με ευέλικτους κατανεμημένους πόρους παραγωγής, αποθήκευσης και κατανάλωσης ηλεκτρικής ενέργειας.

Λέξεις Κλειδιά: επαναλαμβανόμενη λήψη αποφάσεων υπό αβεβαιότητα, νευρωνικά δίκτυα, ADMM, δίκτυο διανομής, ανταλλαγή ενέργειας κόμβο με κόμβο

ABSTRACT

Modern distribution systems with high penetration of distributed energy resources face multiple sources of uncertainty. This transforms the traditional Optimal Power Flow problem into a problem of sequential decision-making under uncertainty. In this framework, the solution concept takes the form of a policy, i.e., a method of making dispatch decisions when presented with a real-time system state. Reasoning over the future uncertainty realization and the optimal online dispatch decisions is especially challenging when the number of resources increases and only a small dataset is available for the system's random variables.

In this paper, we present a data-driven distributed policy for making dispatch decisions online and under uncertainty. The proposed policy is guaranteed to satisfy the system's constraints while experimentally shown to achieve a performance close to the optimal-in-hindsight solution. Moreover, the proposed policy was tested in a peer-to-peer energy system scenario and in an electricity distribution system with many flexible energy resources.

Λέξεις Κλειδιά: sequential decision making under uncertainty, neural networks, ADMM, distribution networks, peer to peer energy trading

ΕΥΧΑΡΙΣΤΙΕΣ

Καταρχάς θα ήθελα να ευχαριστήσω τον καθηγητή κ. Εμμανουήλ Βαρβαρίγο και τον βοηθό καθηγητή(DTU) κ. Γεώργιο Τσαούσογλου για την επίβλεψη αυτής της διπλωματικής εργασίας και για την συνεργασία μας. Θα ήθελα να ευχαριστήσω την οικογένειά μου και τους ανθρώπους που ήταν κοντά μου όλα αυτά τα χρόνια για την καθοδήγηση και την συμπαράσταση που μου προσέφεραν απλόχερα.

ΠΕΡΙΕΧΟΜΕΝΑ

Περίληψη

Abstract

Ευχαριστίες

Εισαγωγή

1.1 Αντικείμενο διπλωματικής.....	7
1.2 Οργάνωση Τόμου.....	7

I. Θεωρητικό Μέρος

Θεωρητικό υπόβαθρο

2.1 Μαθηματική Βελτιστοποίηση.....	9
2.1.1 Κυρτά προβλήματα βελτιστοποίησης.....	9
2.1.2 Κανονική μορφή κυρτού προβλήματος βελτιστοποίησης.....	10
2.1.3 Δυσκότητα- Χαλάρωση Lagrange.....	10
2.1.4 Αποσύνθεση ADMM.....	11
2.2 Νευρωνικά Δίκτυα.....	11
2.2.1 Multi-layer Perceptron.....	12
2.2.2 Graph Neural Networks.....	13
2.3 Κατηγορίες προβλημάτων εύρεσης πολιτικής λύσης(Policy).....	14
2.4 Ενεργειακά Συστήματα.....	15
2.4.1 Smart Grids.....	15
2.4.2 Αποκεντρωμένα Ενεργειακά Συστήματα (Decentralized Energy Systems).....	16

II. Πρακτικό Μέρος

Εφαρμογή αλγορίθμου σε αποκεντρωμένα δίκτυα ηλεκτρικής ενέργειας

3.1 Μοντελοποίηση Συστήματος.....	18
3.2 Διαμόρφωση προβλήματος.....	20
3.3 Προσέγγιση λύσης	21
3.3.1 Αποκεντρωμένη βελτιστοποίηση με πρόβλεψη σημείου (POINT FORECAST).....	22
3.3.2 Αποκεντρωμένο μοντέλο προβλεπόμενου ελέγχου (MODEL PREDICTIVE CONTROL) με πρόβλεψη σημείου (POINT FORECAST).....	23
3.3.3 Αποκεντρωμένο μοντέλο προβλεπόμενου ελέγχου βασισμένο στη μάθηση (LEARNING BASED).....	24
3.3.4 Νευρωνικό δίκτυο για την πρόβλεψη διυδικών μεταβλητών.....	27
3.4 Έλεγχος και αποτελέσματα αλγόριθμου.....	27
3.4.1 Αλγόριθμος Σύγκρισης.....	27
3.5 Περιβάλλον συστήματος.....	28
3.6 Αποτελέσματα προσομοίωσης	29
Εφαρμογή αλγορίθμου σε σύστημα ηλεκτρικής ενέργειας με ευέλικτα φορτία	
4.1 Μοντελοποίηση συστήματος.....	36
4.1.1 Δίκτυο διανομής.....	38
4.2 Παρουσίαση μοντελοποίησης προβλήματος	39
4.3 Αλγόριθμοι σύγκρισης.....	39
4.4 Προτεινόμενος αλγόριθμος	40
4.5 Πειραματική εκτίμηση	43
4.6 Εκτίμηση Αποτελεσμάτων.....	45
III. Επίλογος	
Επίλογος	
5.1 Συμπεράσματα.....	51
5.2 Μελλοντικές Επεκτάσεις.....	51
Βιβλιογραφία	
Απόδοση ξενόγλωσσων όρων	

1. ΕΙΣΑΓΩΓΗ

1.1 Αντικείμενο Διπλωματικής

Η επίτευξη των στόχων μετάβασης σε ένα πιο πράσινο μέλλον, που τέθηκαν απ' όλες τις κυβερνήσεις ανά τον κόσμο, για την διάσωση του πλανήτη από την καταστροφή του περιβάλλοντος, απαιτεί να γίνουν δραστικές αλλαγές στον τρόπο ζωής των ανθρώπων αλλά και σε σημαντικά συστήματα που βοηθούν την ζωή μας. Με αυτό τον τρόπο είναι αναγκαίο να γίνουν σημαντικές τροποποιήσεις στην λειτουργία και την δομή των σύγχρονων ενεργειακών συστημάτων.

Στις μέρες μας έχουν γίνει σημαντικά βήματα με σκοπό να οδηγηθούμε στην βιώσιμη παραγωγή και διανομή ενέργειας. Οι αλλαγές που γίνονται, όμως, κουβαλούν μαζί τους πολλά προβλήματα, τα οποία πρέπει να επιλυθούν. Συγκεκριμένα η εντατική ενσωμάτωση κατανεμημένων ενεργειακών πόρων, η απελευθέρωση της αγοράς ηλεκτρικής ενέργειας και η ύπαρξη ανταγωνισμού μεταξύ των παροχών ηλεκτρικής ενέργειας και η ενσωμάτωση προγραμμάτων demand response είναι μόνο λίγοι από τους παράγοντες που προκαλούν μεγάλη αβεβαιότητα στην λήψη αποφάσεων όσο αναφορά την λειτουργία και επέκταση των σύγχρονων συστημάτων ενέργειας.

Έχοντας υπόψη τα πιο πάνω είναι απαραίτητο να δημιουργηθούν, υπολογιστικά αποδοτικά, εργαλεία τα οποία θα μπορούν να βοηθούν την λήψη αποφάσεων υπό αβέβαιες συνθήκες, μειώνοντας το ρίσκο στην λήψη αποφάσεων. Το γεγονός αυτό αποτελεί ανοικτό πρόβλημα, στις μέρες μας, αφού δεν έχει προταθεί μια μέθοδος που θα είναι ισάξια καλή για όλες τις περιπτώσεις χρήσης της.

1.2 Οργάνωση Τόμου

Στο θεωρητικό υπόβαθρο θα παρουσιαστούν οι μέθοδοι και τεχνολογίες που χρησιμοποιήθηκαν για την θεωρητική εδραίωση του αλγορίθμου και την εφαρμογή του για την επίλυση των προβλημάτων που θα παρουσιαστούν στο πρακτικό κομμάτι. Στο πρακτικό μέρος θα παρουσιαστεί η Μοντελοποίηση των 2 προβλημάτων που επιλύθηκαν και ο ακριβής τρόπος επίλυσης. Τέλος θα παρουσιαστεί ο επίλογος και οι πιθανές μελλοντικές επεκτάσεις του αλγορίθμου που προτάθηκε

I. ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ

2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

2.1 Μαθηματική Βελτιστοποίηση

Μαθηματική Βελτιστοποίηση (Mathematical Optimization) ή αλλιώς μαθηματικός προγραμματισμός (Mathematical Programming) είναι η επιλογή της καλύτερης λύσης, λαμβάνοντας υπόψη κάποιους περιορισμούς (Constraints). Η μαθηματική βελτιστοποίηση [1] χρησιμοποιείται για την λύση προβλημάτων σε όλους τους τομείς, όπως την πληροφορική, την μηχανική και τα οικονομικά. Τα προβλήματα αυτά αποτελούνται από μια συνάρτηση στόχο (Objective function), η οποία απαιτείται να μεγιστοποιηθεί ή να ελαχιστοποιηθεί. Η συνάρτηση διασπάται σε κάποιες παραμέτρους (Parameters) για τις οποίες το πρόβλημα ξέρει τις τιμές α -priori και τις μεταβλητές απόφασης (Decision variables) για τις οποίες ο αλγόριθμος αναμένεται να βρει τις τιμές ώστε να επιτυγχάνεται ο στόχος της και από κάποιες συναρτήσεις οι οποίες αποτελούν τα Constraints. Τα Constraints έχουν ως στόχο τον περιορισμό των λύσεων που μπορεί να έχει το πρόβλημα.

2.1.1 Κυρτά προβλήματα βελτιστοποίησης

Ένα κυρτό πρόβλημα βελτιστοποίησης (convex optimization problem)[2] είναι ένα πρόβλημα το οποίο έχει κυρτή Objective function και το σύνολο(feasible set) των εφικτών λύσεων του είναι κυρτό. Επεξηγηματικά, μια συνάρτηση f που χαρτογραφεί κάποιο υποσύνολο του R^n μέσα στο $R \cup \{\pm\infty\}$ είναι κυρτή αν ο τομέας της είναι κυρτός και για όλα τα $\theta \in [0,1]$ και για όλα τα x, y ισχύει $f(\theta x + (1 - \theta)y) \leq \theta f(x) + (1 - \theta)f(y)$. Συμπληρωματικά, ένα σύνολο S είναι κυρτό αν για όλα τα στοιχεία του συνόλου, $x, y \in S$ και για όλα τα $\theta \in [0,1]$ ισχύει $\theta x + (1 - \theta)y \in S$. Άρα ένα κυρτό πρόβλημα βελτιστοποίησης μπορεί να εκφραστεί σαν ένα πρόβλημα εύρεσης κάποιου $x^* \in S$ επιτυγχάνοντας

$$\inf\{f(x): x \in S\}$$

όπου η $f: S \subseteq R^n \rightarrow R$ είναι η κυρτή objective function, και το S το κυρτό σύνολο εφικτών λύσεων. Αν το x^* υπάρχει τότε αυτό ονομάζεται βέλτιστη λύση του προβλήματος, ή αν υπάρχουν πολλά, τότε ονομάζεται το βέλτιστο σύνολο.

2.1.2 Κανονική μορφή κυρτού προβλήματος βελτιστοποίησης

Ένα κυρτό πρόβλημα βελτιστοποίησης βρίσκεται στην κανονική μορφή του (standard form), όταν γράφεται σαν:

$$\begin{aligned} \min_x F(x) \\ \text{s.t. } g_i(x) \leq 0, i = 1 \dots m \quad (2) \\ h_j(x) = 0, j = 1 \dots p \quad (3) \end{aligned} \quad (1)$$

όπου το $x \in R^n$ είναι το decision variable του προβλήματος, η $f: S \subseteq R^n \rightarrow R$ είναι η κυρτή objective function και οι περιορισμοί ανισότητας $g_i(x) \subseteq R^n \rightarrow R, i = 1 \dots m$ είναι κυρτές συναρτήσεις. Επίσης οι περιορισμοί ισότητας $h_j(x) \subseteq R^n \rightarrow R, j = 1 \dots p$ πρέπει να είναι συγγενείς μετασχηματισμοί της μορφής $h_j = a_j * x - b_j$, όπου a_j είναι διάνυσμα και b_j είναι βαθμωτό μέγεθος.

2.1.3 Δυσκότητα- Χαλάρωση Lagrange

Η θεωρία της δυαδικότητας (Duality) βασίζεται στην θεωρία του να επιτρέπονται κάποιοι περιορισμοί ενός προβλήματος βελτιστοποίησης να παραβιάζονται τιμωρώντας όμως το objective function προσθέτοντας κάποιο συντελεστή τιμωρίας. Αυτό οδηγεί σε ένα καινούργιο πρόβλημα βελτιστοποίησης το οποίο ονομάζεται δυκό πρόβλημα (Dual problem). [3]

Έχοντας το πρόβλημα βελτιστοποίησης στην κανονική του μορφή (1) όπως περιεγράφηκε πιο πάνω μπορούμε να χαλαρώσουμε τους περιορισμούς του. Αυτό σημαίνει ότι επιτρέπουμε στην λύση να παραβεί τους περιορισμούς, με συγκεκριμένο κόστος στην objective function. Έτσι η συνάρτηση Lagrange $L: C \subseteq R^n \times R^m \times R^p$ ως:

$$L(x, \lambda, \mu) = F(x) + \sum_{i=1}^m \lambda_i g_i(x) + \sum_{i=1}^p \lambda_i h_i(x) \quad (2)$$

Διπλωματική Εργασία

Στο δυικό πρόβλημα η L γίνεται η objective function, η οποία είναι το άθροισμα της objective function της (1) με το σταθμισμένο άθροισμα των περιορισμών που χαλαρώθηκαν. Τα βάρη λ_i για τους περιορισμούς ανισότητας (2) και τα μ_i για τους περιορισμούς ισότητας (3) του προβλήματος βελτιστοποίησης (1) ονομάζονται πολλαπλασιαστές Lagrange και αποτελούν τους συντελεστές τιμωρίας, οι οποίοι αυξάνουν το καινούργιο objective function, με την παραβίαση τους.

2.1.4 Αποσύνθεση ADMM

Ο αλγόριθμος ADMM (*alternating direction method of multipliers*)[4] είναι ένας αλγόριθμος, οποίος λύνει κυρτά προβλήματα βελτιστοποίησης, σπάζοντας τα σε μικρότερα κομμάτια, τα οποία έχουν μικρότερη υπολογιστική πολυπλοκότητα. Έχοντας το πρόβλημα

$$\begin{aligned} \min_{x,z} f(x) + g(z) \\ \text{s. t. } Ax + Bz = c \end{aligned}$$

Όπου f και z είναι κυρτές και ορίζοντας κάποιο $\rho > 0$ έχουμε

$$L_\rho = f(x) + g(z) + y^T (Ax + Bz - c) + \left(\frac{\rho}{2}\right) \|Ax + Bz - c\|_2^2$$

με

$$x^{k+1} := \operatorname{argmin}_x L_\rho(x, z^k, y^k) \quad x - \text{minimization}$$

$$z^{k+1} := \operatorname{argmin}_z L_\rho(x^{k+1}, z, y^k) \quad z - \text{minimization}$$

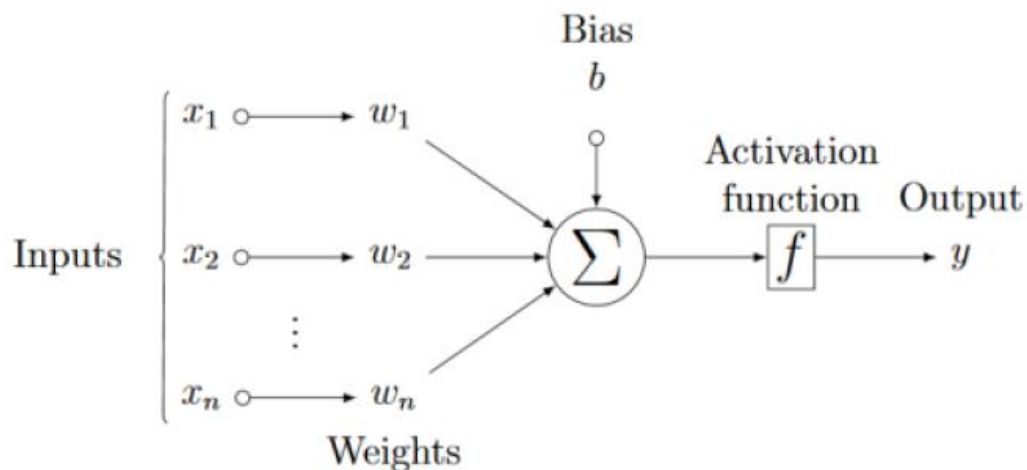
$$y^{k+1} := y^k + \rho(Ax^{k+1} + Bz^{k+1} - c) \quad \text{dual - update}$$

2.2 Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks - ANN)[5] ή σε συντομία Νευρωνικά Δίκτυα είναι από τα κύρια εργαλεία στη μηχανική μάθηση. Χρησιμοποιούν μόνο την κεντρική ιδέα της λογικής λειτουργίας του, ξεκινώντας από μία συλλογή μονάδων, που είναι αντίστοιχες προς τους νευρώνες, και προσπαθούν να επιτελέσουν τις ανάλογες διεργασίες. Τελικά όπως τα βιολογικά και τα τεχνητά δίκτυα διαφέρουν πάρα πολύ ως προς

την αρχιτεκτονική και τις ιδιότητες τους. Αναπτύχθηκαν τα τελευταία χρόνια και έχουν πετύχει αρκετά εντυπωσιακά αποτελέσματα, αλλά έχει φανεί επίσης ότι έχουν και αρκετά μειονεκτήματα. Οι περιορισμοί δημιουργούνται συχνά όταν το μέγεθος και η πολυπλοκότητα του συστήματος αυξάνουν. Είναι ιδιαίτερα ικανά σε συνδυαστικά προβλήματα και σε ταξινομήσεις. Αντίθετα, δεν είναι ικανά σε προβλήματα λογικής και σε ακριβείς υπολογισμούς. Υπάρχουν πάρα πολλά μοντέλα δικτύων με διαφορετική φιλοσοφία και τρόπο λειτουργίας, και πολλές και διάφορες εφαρμογές

2.2.1 Multi-layer Perceptron



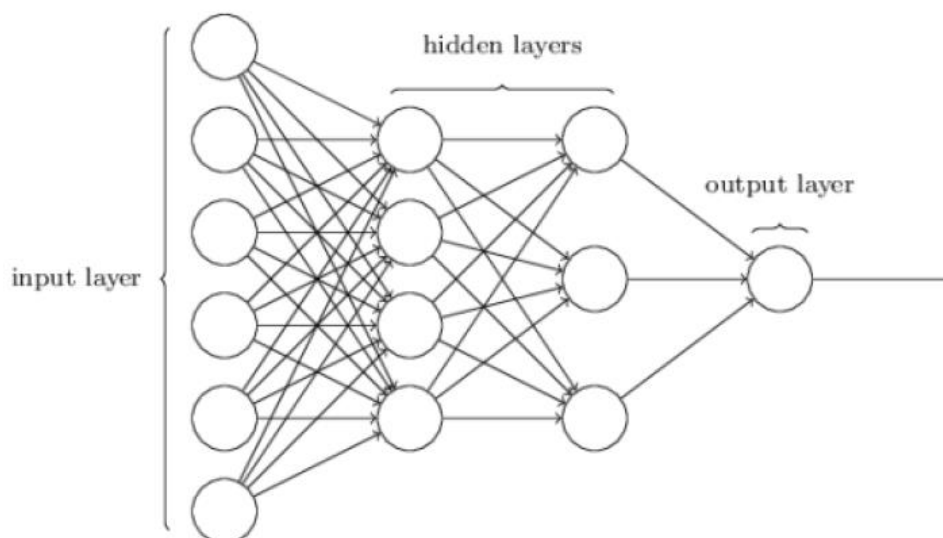
Η βασική μονάδα υπολογισμού σε ένα νευρωνικό δίκτυο είναι ο νευρώνας (Neuron), συχνά ονομαζόμενος και ως κόμβος.[6] Λαμβάνει εισόδους από άλλους κόμβους ή από μια εξωτερική πηγή και υπολογίζει μια έξοδο. Κάθε είσοδος πολλαπλασιάζεται με το αντίστοιχο άρος (Weight) και υπολογίζεται το ολικό άθροισμα των γινομένων. Ο κόμβος εφαρμόζει μια συνάρτηση ενεργοποίησης σε αυτό το άθροισμα και υπολογίζεται η έξοδος του νευρώνα.

Παρακάτω φαίνεται η αναπαράσταση ενός νευρώνα με τρεις εισόδους καθώς και η εξίσωση της εξόδου. Το παραπάνω δίκτυο παίρνει ως εισόδους τα X_1, X_2 έως X_n που έχουν για βάρη τα W_1, W_2 έως W_n αντίστοιχα. Επιπλέον υπάρχει ακόμα μια είσοδος 1 με βάρος b η οποία ονομάζεται πόλωση (bias). Η συνάρτηση f είναι μη γραμμική και ονομάζεται συνάρτηση ενεργοποίησης (activation function). Ο σκοπός της συνάρτησης ενεργοποίησης είναι να εισάγει μη γραμμικότητα στην έξοδο ενός νευρώνα. Αυτό είναι σημαντικό καθώς σχεδόν όλα τα πραγματικά δεδομένα είναι μη γραμμικά. Παραδείγματα τέτοιων

συναρτήσεων είναι η σιγμοειδής συνάρτηση, η υπερβολική εφαπτομένη και η ReLU (Rectified Linear Unit) την οποία και θα χρησιμοποιήσουμε στις εφαρμογές αυτής τη εργασίας. Η ReLU αποδίδει συχνά καλύτερα από άλλες συναρτήσεις ενεργοποίησης για κρυφά επίπεδα. Ο βασικός λόγος της αυξημένης απόδοσης οφείλεται στο γεγονός ότι η ReLU είναι μια γραμμική συνάρτηση μη κορεσμού. Ο κορεσμός είναι το μεγαλύτερο πρόβλημα των δυο προηγούμενων σιγμοειδών συναρτήσεων.

Σε αντίθεση λοιπόν με την logistic ή tanh, η ReLU δεν έχει κορεσμό στο -1, 0 ή 1. Οι πιο πρόσφατες έρευνες αναφέρουν ότι τα κρυμμένα επίπεδα των Νευρωνικών Δικτύων πρέπει να χρησιμοποιούν την ενεργοποίηση του ReLU.

Η βασικότερη μορφή ενός νευρωνικού δικτύου είναι τα Πολυεπίπεδα Perceptrons (Multilayer Perceptrons - MLP). [6] Οι νευρώνες στα MLP είναι οργανωμένοι σε επίπεδα (layers) και δεν υπάρχουν συνδέσεις μεταξύ νευρώνων του ίδιου επιπέδου. Το πρώτο από αυτά τα επίπεδα ονομάζεται επίπεδο εισόδου (layer) και χρησιμοποιείται για την εισαγωγή των δεδομένων. Τα στοιχεία αυτού του επιπέδου δεν αποτελούν νευρώνες καθώς δεν εκτελούν κάποιον υπολογισμό. Ακολουθούν ένα ή περισσότερα κρυφά επίπεδα (hidden layers) και τέλος υπάρχει το επίπεδο εξόδου (output layer). Στην παρακάτω εικόνα φαίνεται ένα παράδειγμα ενός MLP με 6 εισόδους, 2 κρυφά επίπεδα με 4 και 3 νευρώνες αντίστοιχα για το κάθε επίπεδο και ένα επίπεδο εξόδου.



2.2.2 Graph Neural Networks

Το νευρωνικό δίκτυο γράφων [7] είναι μια κλάση τη βαθιάς μάθησης σχεδιασμένα για να βρίσκουν αποτελέσματα σε δεδομένα που περιγράφονται από γράφους. Μπορούν να εφαρμοστούν απευθείας σε γράφους και δίνουν ένα εύκολο τρόπο για να κάνουν προβλέψεις σε επίπεδο κόμβων και επίπεδο γράφου. Το πιο βασικό στοιχείο των GNNs είναι ο γράφος. Στην επιστήμη υπολογιστών ένας γράφος είναι μια δομή δεδομένων που αποτελείται από κόμβους και ακμές.

Το προτεινόμενο μοντέλο αποτελείται από δύο μέρη:

- i. Ένα μέρος που χρησιμοποιεί ένα μηχανισμό διάχυσης της πληροφορίας που ορίζεται από μία συνάρτηση μετάβασης (transition function)
- ii. Ένα μέρος που χρησιμοποιεί τη συσσωρευμένη πληροφορία για να διενεργεί πράξεις συμπερασμού βάση του συγκεκριμένου προβλήματος που καλείται το μοντέλο να λυθεί και που ορίζεται από μια συνάρτηση εξόδου (output function)

Και τα δύο μέρη συντίθενται σε ένα δίκτυο κωδικοποίησης (encoding network) που είναι όμοιο με ένα δίκτυο MLP το οποίο όμως βασίζεται στην υποκείμενη τοπολογία του γράφου. Και η συνάρτηση μετάβασης και η συνάρτηση εξόδου αναπαρίστανται από επιμέρους MLPs.

2.3 Κατηγορίες προβλημάτων εύρεσης πολιτικής λύσης(Policy)

- Προσέγγιση συνάρτησης πολιτικής: Αυτή η τεχνική συμπεριλαμβάνει μια παραμετρική συνάρτηση πολιτικής, για παράδειγμα μια συνάρτηση κλειστής μορφής, που αντιστοιχεί την κατάσταση συστήματος στις εντολές ελέγχου, ή νευρωσικού που δίνει ένα σύνολο από εντολές ελέγχου όταν ξέρει την κατάσταση συστήματος.
- Προσέγγιση συνάρτησης κόστους: Αυτή η τεχνική υπολογίζει την αβεβαιότητα αλλάζοντας την συνάρτηση κόστους του συστήματος, και στην συνέχεια λύνεται ένα ντετερμινιστικό σύστημα που υπολογίζει τις εντολές ελέγχου
- Προσέγγιση συνάρτησης αξίας: Μια από τις πιο χρησιμοποιημένες τεχνικές στον τομέα της τεχνητής νοημοσύνης. Προσομοιώνει την ιδέα της συνάρτησης Bellman, υπολογίζοντας το κόστος που έχει να πάρουμε μια συγκεκριμένη πράξη δεδομένου ότι είμαστε σε ένα συγκεκριμένο state. Αν η συνάρτηση αξίας υπολογιστή καλά, θα

μπορούμε να πάρουμε αποφάσεις που οδηγούν σε λύσεις κοντά στο βέλτιστο, λύνοντας ένα ντετερμινιστικό πρόβλημα βελτιστοποίησης

- Πολιτικές άμεσης πρόβλεψης: Η τελευταία οικογένειες μεθόδων είναι τεχνικές που προσεγγίζουν τις μελλοντικές τροχιές του συστήματος χρησιμοποιώντας μοντέλο με τωρινές πληροφορίες. Έτσι οι εντολές ελέγχου αποφασίζονται, λύνοντας ένα ντετερμινιστικό πρόβλημα βελτιστοποίησης. Ένα παράδειγμα είναι το Model Predictive Control. [8]–[10]

2.4 Ενεργειακά Συστήματα

2.4.1 Smart Grids

Ο όρος έξυπνο δίκτυο ενέργειας [11] δεν έχει έναν ακριβή και κοινώς αποδεκτό ορισμό, αλλά είναι μια ευρεία έννοια που εμπεριέχει μια πληθώρα αλλαγών στο σύστημα ηλεκτρικής ενέργειας [12]. Η έννοια αυτή επιδέχεται πολλές ερμηνείες, οι οποίες εξαρτώνται από πολλούς παράγοντες όπως για παράδειγμα η χώρα, η περιοχή και τα κίνητρα και επιθυμητά οφέλη των διάφορων φορέων που εμπλέκονται. Η Ευρωπαϊκή Πλατφόρμα Τεχνολογίας (European Technology Platform - ETP) ορίζει το έξυπνο δίκτυο ως εξής:

“Ένα ηλεκτρικό δίκτυο που μπορεί έξυπνα να ενοποιήσει τις δράσεις όλων των χρηστών που συνδέονται σε αυτό - παραγωγών, καταναλωτών, και αυτών που είναι και τα δύο - με σκοπό να παρέχει αποτελεσματικά βιώσιμη, οικονομική και ασφαλή παροχή ηλεκτρισμού.[13]

Σύμφωνα με το Υπουργείο Ενέργειας των ΗΠΑ :

Ο όρος “έξυπνο δίκτυο” αναφέρεται σε μια κλάση τεχνολογιών η οποία χρησιμοποιείται για να φέρει τα συστήματα παροχής ηλεκτρισμού στον 21ο αιώνα, χρησιμοποιώντας απομακρυσμένο έλεγχο και αυτοματισμούς βασισμένους σε ηλεκτρονικούς υπολογιστές. Αυτά τα συστήματα είναι δυνατό να λειτουργήσουν χρησιμοποιώντας τεχνολογίες αμφίδρομης επικοινωνίας και υπολογιστικής επεξεργασίας οι οποίες εδώ και δεκαετίες χρησιμοποιούνται σε άλλους τομείς. Έχουν ξεκινήσει να χρησιμοποιούνται και στο δίκτυο ηλεκτρισμού, από τα εργοστάσια παραγωγής και τα αιολικά πάρκα μέχρι τους οικιακούς και εμπορικούς καταναλωτές. Προσφέρουν πολλά οφέλη στις επιχειρήσεις και στους καταναλωτές, τα οποία κυρίως γίνονται εμφανή στις μεγάλες βελτιώσεις της ενεργειακής απόδοσης στο δίκτυο ηλεκτρισμού και στα σπίτια και τα γραφεία των καταναλωτών.[14]

2.4.2 Αποκεντρωμένα Ενεργειακά Συστήματα (Decentralized Energy Systems)

Ο στόχος για μετάβαση στην παραγωγή καθαρής ενέργειας, οδήγησε στην εύρεση πολλών καινοτόμων λύσεων στην δομή των Ενεργειακών συστημάτων. Μια τέτοια καινοτομία, είναι τα αποκεντρωμένα συστήματα ηλεκτρικής ενέργειας, στα οποία οι καταναλωτές μπορούν να έχουν και την ταυτότητα του παραγωγού. Με αυτό τον τρόπο μπορούν να ανταλλάζουν ενέργεια μεταξύ τους, μέσω την υποδομή του συστήματος ηλεκτρικής ενέργειας. Με αυτό τον τρόπο οι συμμετέχοντες, σε αυτό το μοντέλο, έχουν ενεργό ρόλο, δίνοντας τους επιπλέον κίνητρο για επενδύσεις σε περισσότερες «πράσινες» πηγές ενέργειας.[15]

Οι αποφάσεις για τις ενέργειες που γίνονται σε αυτά τα μοντέλα, παίρνονται, ως επί το πλείστον, με την βοήθεια αποκεντρωμένων αλγορίθμων. Συγκεκριμένα αυτά μπορούν να αναπαρασταθούν σαν ένα γράφο, ο οποίος έχει για κόμβους τους συμμετέχοντες και για ακμές τις αμφίδρομες συνδέσεις μεταξύ τους. Ο κάθε συμμετέχοντας τιμολογεί την ενέργεια που δίνει ή παίρνει από το δίκτυο με την δικής του συνάρτηση τιμολόγησης. Τέλος το πρόβλημα εκκαθάρισης αγοράς στο συγκεκριμένο μοντέλο αναφέρεται στην απόφαση του ποσού της ενέργειας που θα καταναλωθεί ή θα πωληθεί από κάθε συμμετέχοντα και του ποσού ανά KWh που θα χρεωθεί η ενέργεια για μεταφορά της στην κάθε ακμή του γράφου. [16]

II. ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ

3 ΕΦΑΡΜΟΓΗ ΑΛΓΟΡΙΘΜΟΥ ΣΕ ΑΠΟΚΕΝΤΡΩΜΕΝΑ ΔΙΚΤΥΑ ΗΛΕΚΤΡΙΚΗΣ ΕΝΕΡΓΕΙΑΣ

3.1 Μοντελοποίηση Συστήματος

Υποθέτουμε ότι έχουμε ένα σύνολο από N συμμετέχοντες ενός δικτύου με δυνατότητα παραγωγής και κατανάλωσης (prosumers) , οι οποίοι είναι ενωμένοι με ένα Γράφο επικοινωνίας. Ο κάθε συμμετέχοντας $n \in N$ αλληλεπιδρά με τους γείτονικούς του συμμετέχοντες, οι οποίοι συμβολίζονται $C_n \subset N$. Θα εφαρμόσουμε τον αλγόριθμο με σκοπό την λήψη αποφάσεων για την διαχείριση και την ανταλλαγή ενέργειας μεταξύ των N σε ένα χρονικό ορίζοντα T χωρισμένο σε διακριτά κομμάτια (timeslots)

Στο timeslot $t \in T$, ένας prosumer $n \in N$ παράγει μια ποσότητα ενέργειας $p_{n,t}^g$, έχοντας κόστος παραγωγής $c_{n,t}^g$, το οποίο ορίζεται ως

$$c_{n,t}^g = a_n (p_{n,t}^g)^2 + b_n p_{n,t}^g + d_n \quad \forall n \in N, t \in T \quad (1)$$

όπου το a_n, b_n, d_n είναι θετικές σταθερές. Θεωρούμε επίσης ένα σύνολο D_n το οποίο συμβολίζει την ευέλικτη ζήτηση όλων των συμμετέχοντων. Η ζήτηση ενός συμμετέχοντα $d_n \in D_n$ συμβολίζεται σαν μια πλειάδα $\Omega_d = (t_d^a, \widetilde{t}_d^a, t_d^d, E_d, \overline{p}_d, w_d)$. Το t_d^a συμβολίζει το διακριτό κομμάτι χρόνου που εκδηλώθηκε η ζήτηση, το \widetilde{t}_d^a τον επιθυμητό χρόνο ολοκλήρωσης της ζήτησης, το $t_d^d > \widetilde{t}_d^a$ το διακριτό κομμάτι χρόνου που επιβάλεται να τελειώσει η μεταφορά ενέργειας για την κάλυψη της ζήτησης. Ακόμη το E_d εκφράζει την συνολική ενέργεια για την κάλυψη της ζήτησης, το \overline{p}_d εκφράζει την μέγιστη ενέργεια κατανάλωσης σε ένα κομμάτι χρόνου, και τέλος το w_d εκφράζει την ελαστικότητα της ζήτησης, δηλαδή πόσο σημαντική είναι η γρήγορη ολοκλήρωσή της. Η ενέργεια που κατανέμεται στην ζήτηση $d \in D_n$ στο χρονικό διάστημα t συμβολίζεται με $p_{d,t}^f$, και η συνολική ενέργεια που ζητείται πρέπει να δοθεί στο χρονικό διάστημα $[t_d^a, t_d^d]$ ώστε

$$\sum_{[t_d^a, t_d^d]} p_{d,t}^f = E_d \quad \forall d \in D_n, n \in N \quad (2)$$

Ένας συμμετέχοντας που εξέφρασε μια ζήτηση μπορεί να καταναλώσει ένα ποσό ενέργειας μεταξύ του μηδέν και της μέγιστης κατανάλωσης ενέργειας σε ένα χρονικό διάστημα. Έτσι

$$0 \leq p_{d,t}^f \leq \bar{p}_d \quad (3)$$

Μπορούμε να παρατηρήσουμε ότι η ανελαστική ζήτηση ενός συμμετέχοντα μπορεί να μοντελοποιηθεί σαν ελαστική ως $E_d = \bar{p}_d(t_d^d - t_d^a)$. Επιπρόσθετα, όταν ένα κομμάτι της ζήτησης ικανοποιηθεί μετά το επιθυμητό χρονικό σημείο \tilde{t}_d^d , υπάρχει ένα επιπρόσθετο κόστος $c_{d,t}^d$ το οποίο ορίζεται ως

$$c_{d,t}^d = \begin{cases} \frac{p_{d,t}^f w_d^{t-\tilde{t}_d^d}}{E_d} & \forall d \in D_n, n \in N, t > \tilde{t}_d^d \\ 0 & \forall d \in D_n, n \in N, t \leq \tilde{t}_d^d \end{cases} \quad (4)$$

Ένας συμμετέχοντας μπορεί να ικανοποιήσει την ζήτηση του με την δική του παραγωγή και/ή να ανταλλάξει ενέργεια με τους γείτονές του $m \in C_n$. Συμβολίζουμε την ενέργεια που στέλνει ο συμμετέχοντας n στον γείτονα του $m \in C_n$ στο χρονικό σημείο $t \in T$ σαν $p_{nm,t}$. Η ενέργεια αυτή είναι περιορισμένη ανάλογα με τα χαρακτηριστικά των δύο συμμετεχόντων στην ανταλλαγή, ως εξής:

$$\underline{p}_{nm} \leq |p_{nm,t}| \leq \overline{p}_{nm} \quad \forall n \in N, m \in C_n, t \in T \quad (5)$$

Η εσωτερική ενεργειακή ισορροπία από τον κάθε συμμετέχοντα εκφράζεται ως

$$p_{n,t}^g = \sum_{d \in D_n} p_{d,t}^f + \sum_{m \in C_n} p_{nm,t} \quad \forall n \in N, t \in T \quad (6)$$

Τέλος για κάθε μεταφορά ενέργειας πρέπει να ισχύει

$$p_{nm,t} = -p_{mn,t} \quad \forall n, m \in N, t \in T \quad (7)$$

Οι συμμετέχοντες επιλέγουν με ποιους γείτονες θα ανταλλάξουν ενέργεια, αφού ο κάθε συμμετέχοντας έχει κόστος q_{mn} για την αγορά ενέργειας από τον $m \in C_n$. Έτσι το κόστος ανταλλαγής ενέργειας

$$c_{n,t}^{ex} = - \sum_{m \in C_n} q_{mn} p_{mn,t} \quad \forall n \in N, t \in T \quad (8)$$

3.2 Διαμόρφωση προβλήματος

Σε κάθε timeslot t , για τις αποφάσεις για την παραγωγή, την κατανάλωση και τις Peer to Peer ανταλλαγές ενέργειας, για όλους τους prosumers, πρέπει να λαμβάνεται υπόψη το μέλλον, αφού



Fig. 1: Graphical illustration of the system's architecture.

Ο στόχος είναι να βρεθεί ένας κανόνας ελέγχου(control policy) $\pi = (\pi)_{\{n \in N\}}$ για τις μεταβλητές $u_t \triangleq (p_{n,t}^g, (p_{d,t}^f)_{d \in D_n}, (p_{nm,t})_{m \in C_n}) \forall n \in N$. Η κατάσταση(state) του συστήματος περιλαμβάνει όλες τις σχετικές πληροφορίες, για να αποφασιστεί η ενέργεια χειρισμού(control action), και ορίζεται ως

$$x_t \triangleq \left(t, ((\Omega_d)_{d \in D_n})_{n \in N}, \left(\sum_{\tau \in [t_d^q, t-1]} p_{d,\tau}^f \right)_{d \in D_n, n \in N} \right) \quad (9)$$

όπου η πρώτη μεταβλητή του state, δηλώνει το τωρινό timeslot, το δεύτερο σύνολο μεταβλητών δηλώνει όλα τα demands όλων των prosumers, και η τρίτη μεταβλητή δηλώνει το πόση ενέργεια δόθηκε μέχρι το t για να ικανοποιηθεί το demand κάθε prosumer.

Για τα demands τα οποία είναι ανενεργά στο t (τα οποία τελείωσαν ή δεν δηλώθηκαν ακόμα), βάζουμε τις τιμές τους στο Ω_d σαν άκυρες (null). Αφίξη ενός καινούργιου demand, στο t σηματοδοτεί την διατάραξη (disturbance) w του συστήματος. Τελικά το ολικό κόστος συστήματος, καθώς φεύγουμε από το timeslot t , προκύπτει από την εξίσωση:

$$c_t = \sum_{n \in N} c_{n,t}^g + \sum_{d \in D_n} c_{d,t}^g + c_{n,t}^{ex} \quad (10)$$

Η αντικειμενική συνάρτηση μοντελοποιείται σαν ένα πρόβλημα βέλτιστου ελέγχου [17]

$$\min_{\pi} \left\{ E \left[\sum_{t \in T} c_t \right] \right\} \quad (11)$$

$$s. t. \quad x_{t+1} = f(x_t, u_t, w_t)$$

$$(1) - (10)$$

3.3 Προσέγγιση λύσης

Εκμεταλλευόμαστε μια ειδική ιδιότητα του προβλήματος: «καθώς στα γενικά προβλήματα βέλτιστου ελέγχου, οι διαταράξεις w_t είναι (πιθανοτηκές) συναρτήσεις της κατάστασης και του ελέγχου, $w_t \in P(\cdot | x_t, u_t)$, το σύστημα έχει την ειδική ιδιότητα των εξωγενών ή των ανεξάρτητων από τον έλεγχο διαταραχών w_t , αφού τα καινούργια demand που έρχονται (ευέλικτα ή μη ευέλικτα) δεν ακολουθούν τις αποφάσεις διαχείρισης ενέργειας αλλά ακολουθούν μια ανεξάρτητη στοχαστική διαδικασία». Αυτό μας επιτρέπει να παράγουμε δείγματα του βέλτιστου ελέγχου για συγκεκριμένες διαταραχές. Πρώτα παράγουμε ένα σύνολο S από σενάρια από διαταραχές, όπου κάθε σενάριο $s \in S$ αντιπροσωπεύει, μια τυχαία πορεία διαταραχής (disturbance trajectories) στο σύνολο όλων των διαταραχών.

$$s \triangleq [w_t]_{t \in T} \quad (12)$$

Μετά βρίσκουμε της βέλτιστες ενέργειες ελέγχου για κάθε σενάριο, λύνοντας το ντετερμινιστικό πρόβλημα βελτιστοποίησης. Αυτά τα δείγματα αποφάσεων, θα χρησιμοποιηθούν για το κτίσιμο της

Στα επόμενα υποκεφάλαια θα παρουσιάσουμε διαφορετικές πολιτικές(policies) οι οποίες κτίζονται βάση της θεωρίας του παρόντος υποκεφαλαίου.

3.3.1 Αποκεντρωμένη βελτιστοποίηση με πρόβλεψη σημείου (POINT FORECAST)

Χρησιμοποιώντας ένα σενάριο $s \in S$ σαν πρόβλεψη σημείου, μπορούμε να μοντελοποιήσουμε ένα ντετερμινιστικό πρόβλημα πολυμερούς οικονομικής κατανομής(multilateral economic dispatch problem). Αυτό ορίζεται ως την ελαχιστοποίηση όλων των κοστών των prosumers, υπό τους περιορισμούς ανταλλαγής ενέργειας και ικανοποίησης των demand.

$$\min_{p_{n,t}^g, p_{n,t}^f, p_{nm,t}} \left\{ \sum_{t \in T} \sum_{n \in N} (c_{n,t}^g + \sum_{d \in D_n} c_{d,t}^g + c_{n,t}^{ex}) \right\} \quad (13)$$

$$s. t. (1) - (8)$$

Το (13) είναι κυρτό πρόβλημα βελτιστοποίησης οπότε μπορεί να λυθεί εύκολα. Επίσης η βέλτιστη λύση μπορεί να βρεθεί με ένα καταναμημένο αλγόριθμο που ονομάζεται ADMM. Έτσι θα δείξουμε πως το ντετερμινιστικό πρόβλημα (13) θα μετασχηματιστεί σε ADMM αλγόριθμο. Πρώτα χαλαρώνουμε τους περιορισμούς ισορροπίας (7) και έτσι το augmented Lagrangian του (13) γράφεται ως

$$L = \sum_{t \in T} \sum_{n \in N} (c_{n,t}^g + \sum_{d \in D_n} c_{d,t}^g + c_{n,t}^{ex}) - \sum_{m \in C_n} (\lambda_{nm,t} (p_{nm,t} + p_{mn,t}) - \frac{\rho}{2} (p_{nm,t} + p_{mn,t})^2) \quad (14)$$

Όπου το $\lambda_{nm,t}$ είναι ο πολλαπλασιαστής Langrange των περιορισμών (7). Στο ADMM, κάθε prosumer, λύνει επαναληπτικά ένα τοπικό πρόβλημα βελτιστοποίησης, δηλαδή λύνει ένα πρόβλημα για τις δικές του μεταβλητές $p_{n,t}^g, p_{n,t}^f, p_{nm,t}$. Οι συμβολισμοί $p_{n,t}^g[k], p_{n,t}^f[k], p_{nm,t}[k]$ συμβολίζουν τις αποφάσεις του n στην επανάληψη k και οι μεταβλητές $(\lambda_{nm,t}[k])_{n,m \in N, t \in T}$ συμβολίζουν την πλειάδα όλων των πολλαπλασιαστών Langrange στην επανάληψη k . Οπότε το τοπικό πρόβλημα βελτιστοποίησης για τον prosumer n στην επανάληψη k γράφεται ως

$$p_{n,t}^g[k], p_{n,t}^f[k], p_{nm,t}[k] = \operatorname{argmin}\{L\} \quad (15)$$

$$s. t. (1) - (6), (8),$$

Διπλωματική Εργασία

$$p_{i,t}^g, p_{i,t}^f, p_{ij,t} = p_{i,t}^g[k-1], p_{i,t}^f[k-1], p_{ij,t}[k-1] \forall i \neq n$$

Ο τελευταίος περιορισμός δηλώνει ότι για το τοπικό πρόβλημα του n prosumer, όλες οι μεταβλητές απόφασης των άλλων prosumers που αποφασίστηκαν, είναι σταθερές, στις τιμές που αποφασίστηκαν στην προηγούμενη επανάληψη. Κάθε prosumer λύνει το δικό του τοπικό πρόβλημα (15) παράλληλα με τους άλλους και έπειτα ανανεώνονται οι πολλαπλασιαστές:

$$\lambda_{nm,t}[k+1] = \lambda_{nm,t}[k] + \rho(p_{nm,t}[k] + p_{mn,t}[k]) \quad (16)$$

$$\lambda_{mn,t}[k+1] = \lambda_{mn,t}[k] - \rho(p_{nm,t}[k] + p_{mn,t}[k])$$

$$\forall n, m \in N, t \in T$$

Η διαδικασία συνεχίζεται μέχρι

$$\max_{n,m \in N, t \in T} \{p_{nm,t} + p_{mn,t}\} \leq \epsilon \quad (17)$$

Δηλαδή η μεγαλύτερη παραβίαση του περιορισμού (7) πρέπει να είναι κάτω από κάποιο όριο.

3.3.2 Αποκεντρωμένο μοντέλο προβλεπόμενου ελέγχου (MODEL PREDICTIVE CONTROL) με πρόβλεψη σημείου (POINT FORECAST)

Ο προσδιορισμός των τιμών των μεταβλητών ελέγχου για όλο τον χρονικό ορίζοντα, πριν το πρώτο timeslot με την χρήση του point forecast, ονομάζεται πολιτική ανοικτού βρόγχου. Στο καταναμημένο Model Predictive Control, ο ADMM αλγόριθμος, εκτελείτε όπως δείξαμε στο προηγούμενο υποκεφάλαιο, με την διαφορά ότι μετά την μετάβαση του συστήματος στο επόμενο timeslot και την παρατήρηση των νέων διαταραχών, ανανεώνονται οι προβλέψεις σημείου και η διαδικασία του ADMM εκτελείτε ξανά για να βρεθούν οι νέες τιμές των μεταβλητών ελέγχου.

Ορίζουμε το τ ως την παράμετρο που μας δείχνει το τωρινό timeslot στην εκτέλεση του αλγόριθμου σε πραγματικό χρόνο, και τις μεταβλητές $\widetilde{p}_{n,\tau}^g, \widetilde{p}_{n,\tau}^f, \widetilde{p}_{nm,\tau}$ δηλώνουν τις τελικές αποφάσεις για το τ . Τότε το τοπικό πρόβλημα για τον prosumer n σε κα'θε timeslot τ , με χρήση της εξίσωσης (15) και προσθέτοντας τον περιορισμό ότι οι παλιές αποφάσεις δεν μπορούν να αλλάξουν γράφεται ως

$$p_{n,t}^g[k], p_{n,t}^f[k], p_{nm,t}[k] = \operatorname{argmin}\{L\} \quad (18)$$

s. t. (1) – (6), (8),

$$p_{i,t}^g, p_{i,t}^f, p_{ij,t} = p_{i,t}^g[k-1], p_{i,t}^f[k-1], p_{ij,t}[k-1] \forall i \neq n$$

$$p_{n,t}^g, p_{n,t}^f, p_{nm,t} = \widetilde{p}_{n,\tau}^g, \widetilde{p}_{n,\tau}^f, \widetilde{p}_{nm,\tau} \quad \forall t < \tau, n, m \in \mathcal{N}$$

Έτσι ο αλγόριθμος μπορεί να γραφεί ως

Algorithm 1 Distributed point-forecast model predictive control

- 1: Initialize $\tau = 0$
 - 2: **While** $\tau \in \mathcal{T}$
 - 3: Observe system state \mathbf{x}_τ
 - 4: Generate a point-forecast for future disturbances
 - 5: Initialize $k = 1, (\lambda_{nm,\tau}[0])_{n,m \in \mathcal{N}} = 0$
 - 6: Initialize $(p_{n,t}^s[0], p_{d,t}^f[0], p_{nm,t}[0])_{t \in \mathcal{T}}$ at random
 - 7: **while** $\max_{n \in \mathcal{N}, m \in \mathcal{C}_n, t \in \mathcal{T}} \{p_{nm,t}[k] + p_{mn,t}[k]\} > \varepsilon$:
 - 8: Each prosumer solves problem (18) to decide
 $p_{n,t}^s[k], p_{d,t}^f[k], p_{nm,t}[k]$
 - 9: Set $\lambda_{nm,t}[k+1]$ using (16)
 - 10: Set $k = k + 1$
 - 11: Set $(\widetilde{p}_{n,\tau}^s, \widetilde{p}_{n,\tau}^f, \widetilde{p}_{nm,\tau})_{n,m \in \mathcal{N}}$ to the converged values
 $(p_{n,\tau}^s[k-1], p_{d,\tau}^f[k-1], p_{nm,\tau}[k-1])_{n,m \in \mathcal{N}}$
 - 12: Set $\tau = \tau + 1$
 - 13: Realize disturbances
 - 14: System transitions to the new state
-

Η αδυναμία όμως αυτού του αλγορίθμου είναι ότι χρησιμοποιεί ένα μόνο point forecast(σενάριο) για μελλοντικές διαταραχές, και άρα ο αλγόριθμος αγνοεί την πλήρη στατιστική πληροφορία που υπάρχει. Συμπληρωματικά, δεν μπορούμε να μετατρέψουμε το πρόβλημα μας σε πρόβλημα στοχαστικού προγραμματισμού με πολλά σενάρια, λόγω του πλήθους των μεταβλητών με αβεβαιότητα, που υπάρχουν στο πρόβλημα.

Έτσι στο επόμενο υποκεφάλαιο θα παρουσιάσουμε την προτεινόμενη λύση.

3.3.3 Αποκεντρωμένο μοντέλο προβλεπόμενου ελέγχου βασισμένο στη μάθηση (LEARNING BASED)

Για την λύση αυτή, λύνουμε πολλά στιγμιότυπα του ντετερμινιστικού προβλήματός μας (13), σε μη πραγματικό χρόνο(offline), αποθηκεύοντας τα states $[x_t^s]_{t \in \mathcal{T}}$ και τις αντίστοιχες

βέλτιστες αποφάσεις ελέγχου $[u_t^s]_{t \in T}$. Έτσι επιτυγχάνεται η παραγωγή του συνόλου Y που περιλαμβάνει πολλά ζευγάρια $y \in Y$ της μορφής

$$y \triangleq (x_t^s, u_t^s) \quad (19)$$

Θεωρητικά μπορούμε να εκπαιδεύσουμε ένα μοντέλο μηχανικής μάθησης το οποίο θα μας εξάγει τις βέλτιστες ενέργειες χειρισμού u_t παίρνοντας σαν είσοδο το τωρινό state του συστήματος, x_t σε πραγματικό χρόνο. Έτσι η υπολογιστική δυσκολία του προβλήματος μεταφέρεται στην εκπαίδευση του δικτύου offline, και άρα σε πραγματικό χρόνο θα μπορούμε να έχουμε την λύση σε δευτερόλεπτα. Αυτό ονομάζεται στην βιβλιογραφία σαν “learn to optimize”.

Αυτή η μέθοδος όμως έχει δύο προβλήματα. Πρώτα απ’ όλα δεν μπορούμε να αποδείξουμε ότι τα παραγόμενα αποτελέσματα θα τηρούν τους περιορισμούς (6) και (7). Επίσης η εκπαίδευση για μεγάλες διαστάσεις του state, το καθιστά αδύνατο.

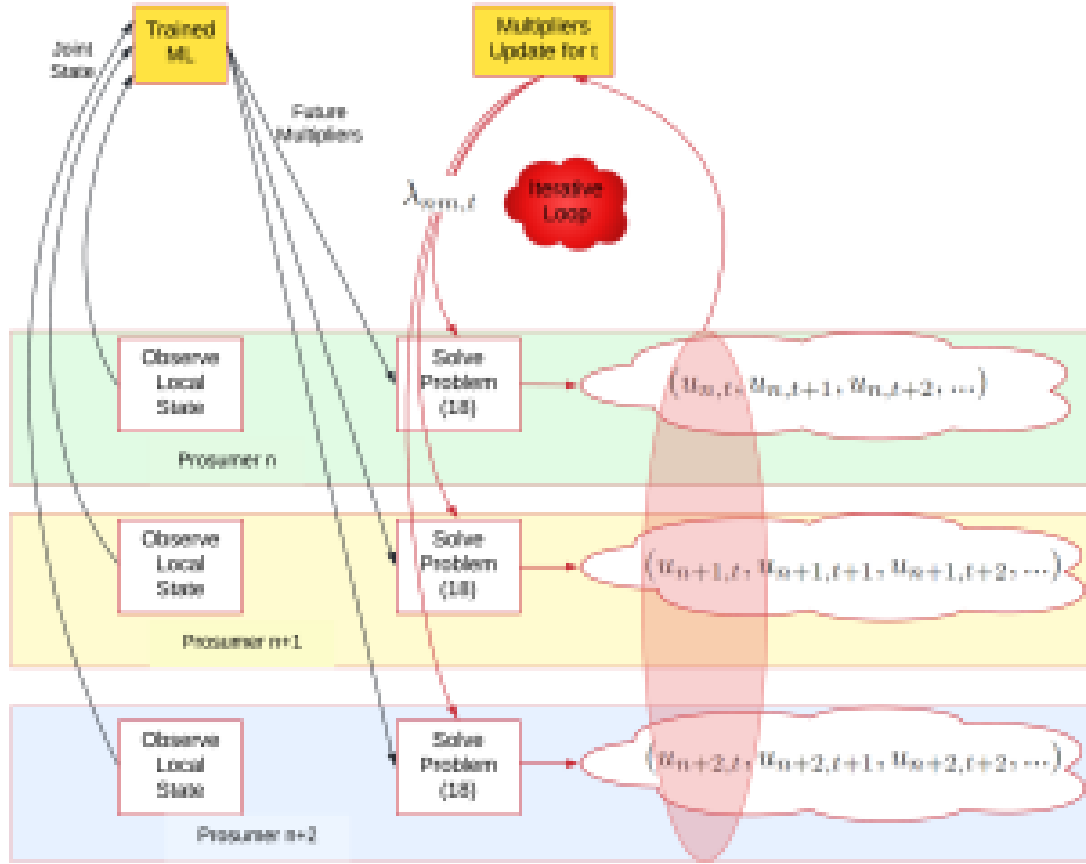


Fig. 2: Online procedure at a timeslot t .

Παρατηρούμε ότι το πρόβλημα βελτιστοποίησης (13) είναι κυρτό και άρα με την λύση ενός σεναρίου s μπορούμε να ξέρουμε τους πολλαπλασιαστές Lagrange $\lambda_{n,m,t}$ που αντιστοιχούν στον περιορισμό (7). Αρά προτείνουμε να χρησιμοποιήσουμε τα σεσάρια και την βέλτιστη λύση τους ώστε να παράγουμε το εξής

$$y^* \triangleq (x_t^s, (\lambda_{n,m,\tilde{t}})_{n \in N, m \in C_n, \tilde{t} \in [t+1, |T|]})$$

Ενώνοντας το κάθε state με τους αντίστοιχους πολλαπλασιαστές Lagrange, αντιμετωπίζοντας έτσι και τα προβλήματα που αναφέρθηκαν πιο πάνω. Με αυτό τον τρόπο μειώθηκαν οι διαστάσεις των μεταβλητών απόφασης σε $\frac{1}{2} |N| |C_n| |T|$, και άρα το νευρωνικό εκπαιδεύεται για να αποφασίζει λιγότερες μεταβλητές.

Αυτό μπορεί να χρησιμοποιηθεί σε εφαρμογές πραγματικού χρόνου, ως μέρος του direct look ahead policy, προβλέποντας τα $\lambda_{n,m,t}$ και εισάγοντας τα στο πρόβλημα . Στην προτεινόμενη μέθοδο όμως σε κάθε επανάληψη k στο τωρινό timeslot τ , κάθε prosumer $n \in \mathcal{N}$ λύνει το δικό του τοπικό πρόβλημα (18), ικανοποιώντας τους δικούς του περιορισμούς. Στην συνέχεια με την εκτέλεση του ADMM, οι υπολογισμένοι πολλαπλασιαστές Lagrange για το τωρινό timeslot $\lambda_{n,m,t}$ ενημερώνονται, ενώ οι $(\lambda_{n,m,\tau})_{\tau > t}$ είναι σταθερά αυτοί που υπολογίστηκαν από το machine learning μοντέλο.

Όταν ο αλγόριθμος ADMM συγκλίνει, εφαρμόζονται οι ενέργειες ελέγχου και το σύστημα προχωρά στο επόμενο timeslot $\tau + 1$, στο οποίο προβλέπονται καινούργιες τιμές από το

Algorithm 2 Model predictive control with direct lookahead policy

- 1: Initialize $\tau = 0$
 - 2: **While** $\tau \in \mathcal{T}$
 - 3: Observe system state \mathbf{x}_τ
 - 4: Feed state \mathbf{x}_τ to the ML
 - 5: Set the future multipliers $(\lambda_{nm,t})_{t > \tau, n \in \mathcal{N}, m \in \mathcal{C}_n}$ as prescribed by the ML
 - 6: Initialize $k = 1, (\lambda_{nm,\tau}[0])_{n,m \in \mathcal{N}} = 0$
 - 7: Initialize $(p_{n,t}^g[0], p_{d,t}^f[0], p_{nm,t}[0])_{t \in \mathcal{T}}$ at random
 - 8: **while** $\max_{n \in \mathcal{N}, m \in \mathcal{C}_n} \{p_{nm,\tau}[k] + p_{mn,\tau}[k]\} > \varepsilon$:
 - 9: Each prosumer solves problem (18) to decide $p_{n,t}^g[k], p_{d,t}^f[k], p_{nm,t}[k]$
 - 10: Set $\lambda_{nm,\tau}[k+1]$ using (16)
 - 11: Set $k = k + 1$
 - 12: Set $(\tilde{p}_{n,\tau}^g, \tilde{p}_{n,\tau}^f, \tilde{p}_{nm,\tau})_{n,m \in \mathcal{N}}$ to the converged values $(p_{n,\tau}^g[k-1], p_{d,\tau}^f[k-1], p_{nm,\tau}[k-1])_{n,m \in \mathcal{N}}$
 - 13: Set $\tau = \tau + 1$
 - 14: Realize disturbances
 - 15: System transitions to the new state
-

νευρωνικό δίκτυο σύμφωνα με το τωρινό state και επαναλαμβάνεται η διαδικασία του αλγόριθμου ADMM.

3.3.4 Νευρωνικό δίκτυο για την πρόβλεψη διυδικών μεταβλητών

Για την πρόβλεψη των δυικών μεταβλητών $(\lambda_{n,m,t})_{t \in \mathcal{T}; t > \tau}$ το νευρωνικό παίρνει σαν είσοδο το state του συστήματός μας \mathbf{x}_τ την χρονική στιγμή τ . Το νευρωνικό δίκτυο που χρησιμοποιήθηκε είναι ένα απλό Multilayer Perceptron με 5 ενδιάμεσα πυκνά στρώματα. Σε κάθε layer εφαρμόζαμε batch normalization για να φέρνουμε τα δεδομένα στο διάστημα

[0,1]. Επίσης χρησιμοποιήθηκαν τα dropout layers για αποφυγή του overfitting, μετά από κάθε επίπεδο batch normalization. Το πρώτο επίπεδο νευρώνων έχει μέγεθος, όσο οι διαστάσεις του state και το τελευταίο επίπεδο νευρώνων έχει μέγεθος όσο οι πολλαπλασιαστές language του συστήματος σε όλο τον χρονικό ορίζοντα. Κάθε νευρώνας του δικτύου ενεργοποιείται με την Relu συνάρτηση.

3.4 Έλεγχος και αποτελέσματα αλγόριθμου

3.4.1 Αλγόριθμος Σύγκρισης

Για να παρατηρήσουμε την αποτελεσματικότητα του αλγορίθμου μας, το συγκρίνουμε με 2 άλλους αλγόριθμους στα ίδια σενάρια. Ο πρώτος αλγόριθμος είναι ο βέλτιστος, θεωρώντας δηλαδή ότι ξέρει όλες τις πληροφορίες (πχ τη ζήτηση από κάθε prosumer) από πριν. Αυτό ο αλγόριθμος μπορεί να εφαρμοστεί μόνο στην θεωρία και οι αποφάσεις παίρνονται από το ντετερμινιστικό πρόβλημα βελτιστοποίησης (13).

Ο δεύτερος αλγόριθμος με τον οποίο συγκρίνουμε τον προτεινόμενο, χρησιμοποιεί μια πιο συντηρητική πολιτική, προσπαθώντας να ικανοποιήσει την ζήτηση κάθε prosumer μόλις αυτή εμφανιστεί, όσο πιο σύντομα γίνεται. Αυτό είναι και η πολιτική που χρησιμοποιείται στα παρόντα συστήματα ηλεκτρικής ενέργειας, στα οποία η ζήτηση θεωρείται ανελαστική και εξυπηρετείται με την άφιξή της, προσπαθώντας να ελαχιστοποιήσει το κόστος παραγωγής. Το μοντέλο μας τροποποιείται στο συντηρητικό αλγόριθμο, σταθεροποιώντας την κατανάλωση .

$$p_{d,t}^f = \min \left\{ \bar{p}_d, \left[E_d - \sum_{\tau \in [t_d^a, t-1]} p_{d,\tau}^f \right]^+ \right\}$$

όπου το $[x]^+$ συμβολίζει την πράξη $\max\{0, x\}$. Μετά λύνουμε το πρόβλημα μόνο για να αποφασίσουμε για την βέλτιστη παραγωγή και τις ανταλλαγές ενέργειας,

$$\begin{aligned} \min_{p_{n,t}^g, p_{n,t}^f, p_{nm,t}} \left\{ \sum_{n \in N} (c_{n,t}^g + \sum_{d \in D_n} c_{d,t}^g + c_{n,t}^{ex}) \right\} \\ s. t. (1) - (8), \quad (21) \end{aligned}$$

3.5 Περιβάλλον συστήματος

TABLE I: Parameter values

Prosumer	a_n	b_n	$q_{n,m}$	$p_{n,m}$	$\bar{p}_{n,m}$
Agent 0	0	0	0	-34	0
Agent 1	0	0	0	-54	0
Agent 2	0	0	0	-47	0
Agent 3	0	0	0	-58	0
Agent 4	0	0	0	-50	0
Agent 5	0	0	0	-49	0
Agent 6	0.3351	0.3297	0.9326	0	52
Agent 7	0.0498	0.4701	0.6676	0	36
Agent 8	0.8167	0.4464	0.4040	0	56
Agent 9	0.2919	0.6128	0.6654	0	51
Agent 10	0.4254	0.2279	0.4208	0	50
Agent 11	0.5444	0.4998	0.9972	0	57

Οι αλγόριθμοι αυτοί εφαρμόζονται σε ένα σύστημα με 12 prosumers, από τους οποίους οι 6 είναι συνεχώς παραγωγικοί και οι 6 συνεχώς καταναλωτές, το οποίο έχει χρονικό ορίζοντα 24 timeslots. Σε κάθε σενάριο οι συνδέσεις μεταξύ των prosumers είναι σταθερές και φαίνονται στο fig(3) ενώ οι υπόλοιπες τιμές του συστήματος φαίνονται στο table 1.

Θεωρούμε ότι κάθε καταναλωτής μπορεί να δηλώσει μια ζήτηση κατά την διάρκεια των 24 timeslots. Η διάρκεια και η αναχώρηση κάθε ζήτησης εξάχθηκαν από μια κατανομή Poisson με παράγοντα 5 και 15 αντίστοιχα.



Fig. 3: Graph of prosumers' bilateral connections

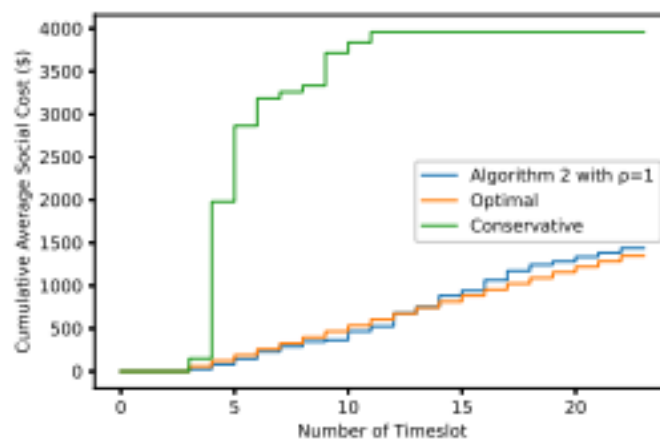


Fig. 4: Average system cost accumulated by each scheme over the horizon.

3.6 Αποτελέσματα προσομοίωσης

Το νευρωνικό δίκτυο εκπαιδεύτηκε για 2400 states και ελέγχθηκε σε 240 states. Το απόλυτο μέσο σφάλμα (MAE) των τιμών ελέγχου ήταν 0.45. Τα αποτελέσματα που θα παρουσιαστούν είναι ο μέσος όρος κάποιων σεναρίων. Στο fig.4 φαίνεται η σύγκριση των εξαγόμενων κοστών c_t συστήματος για τους 3 αλγορίθμους, όπως ορίζονται από την εξίσωση (10). Όπως παρατηρούμε ο συντηρητικός αλγόριθμος αυξάνει το κόστος του συστήματος

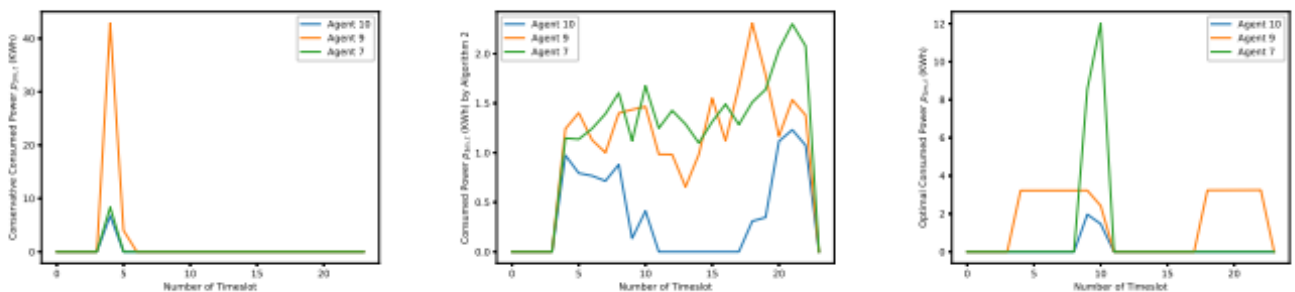
Διπλωματική Εργασία

πολύ σύντομα, αφού εξυπηρετεί τις ζητήσεις μόλις δηλωθούν, και άρα δεν αυξάνει το κόστος του μετά από κάποιο σημείο. Σε αντίθεση, ο προτεινόμενος αλγόριθμος αποφεύγει να ικανοποιήσει την ζήτηση στα αρχικά timeslots. Όμως η καθυστέρηση στην ικανοποίηση μιας ζήτησης επιφέρει επιπλέον κόστος στο σύστημα σύμφωνα με την εξίσωση (4). Συνολικά μπορούμε να πούμε ότι το τελικό κόστος του συστήματός μας είναι πολύ κοντά στο θεωρητικό βέλτιστο, παρουσιάζοντας παράλληλα το πλεονέκτημά του έναντι στον συντηρητικό αλγόριθμο. Στην πραγματικότητα η διαφορά του κόστους μεταξύ του βέλτιστου και του προτεινόμενου αλγόριθμου είναι 8.5%.

Παρακάτω παρουσιάζονται οι ανταλλαγές ενέργειας μεταξύ συγκεκριμένων prosumers (peer 3) σε κάθε ένα από τα 3 γραφήματα του fig.5 και η συνολική ενέργεια που καταναλώθηκε από τον peer 3 σε όλο τον χρονικό ορίζοντα στο fig.6

Για να μελετήσουμε την επεκτασιμότητα και την ευαισθησία του αλγορίθμου μας σε δεδομένα που παράγονται με άγνωστες κατανομές, παράξαμε δεδομένα από κατανομές poisson με διαφορετικές παραμέτρους, τα βάλαμε σαν είσοδο στο νευρωνικό μας δίκτυο και μετρήσαμε πως άλλαξε το ποσοστό λάθους με την αλλαγή της παραμέτρου Poisson. Συγκεκριμένα εξάγαγέ το γράφημα fig.7, όπου το «error ratio» είναι το ποσοστό που διαφέρει το τελικό κόστος συστήματος από το βέλτιστο.

Τέλος μελετήσαμε τον αλγόριθμό μας στην αλλαγή της ρ παραμέτρου που εισάγεται κατά την υλοποίηση του ADMM. Έτσι μπορέσαμε να εξάγουμε αποτελέσματα, για να βρούμε το βέλτιστο ρ το οποίο εξασφαλίζει επεκτασιμότητα, ταχύτητα εύρεσης λύσης και βελτιστοποίηση κόστους. Τα αποτελέσματα αυτά μπορούν να φανούν στο γράφημα 8β



(a) Power exchanges of peer 3, under the conservative policy (b) Power exchanges of peer 3, under the proposed policy (c) Optimal-in-hindsight power exchanges of peer 3.

Fig. 5: Power exchanges of peer 3 with its neighbors, under the three schemes.

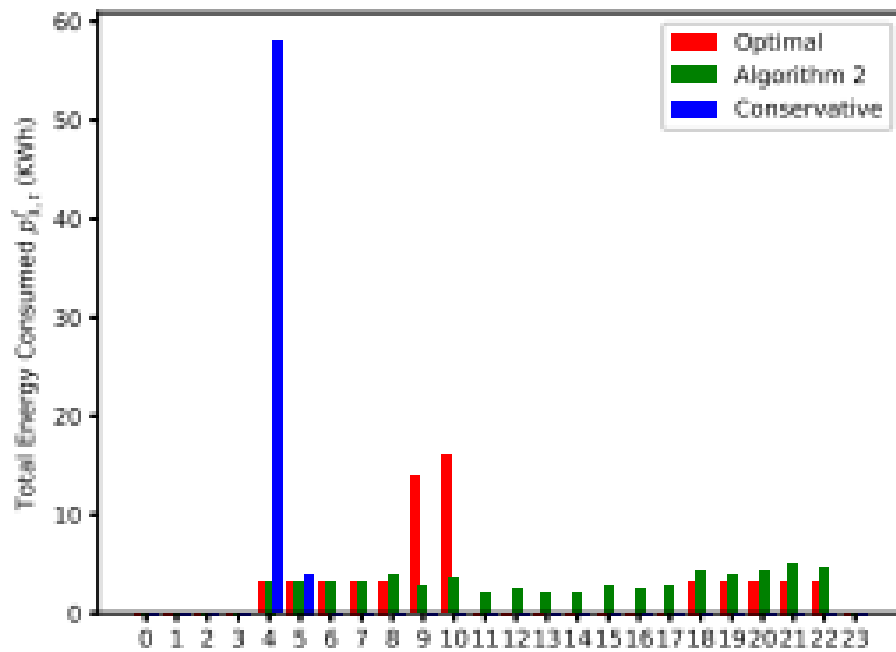


Fig. 6: Total Energy Consumed by Prosumer 3 in each timeslot

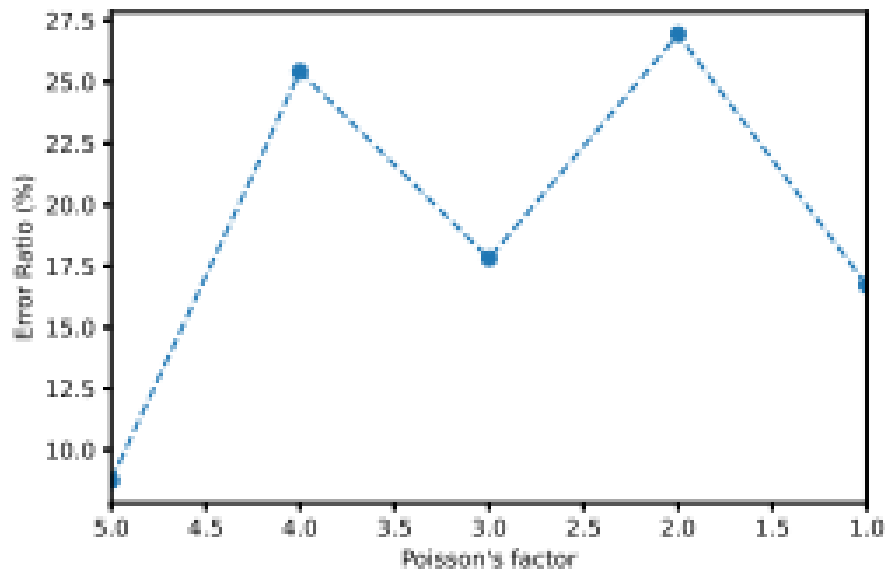
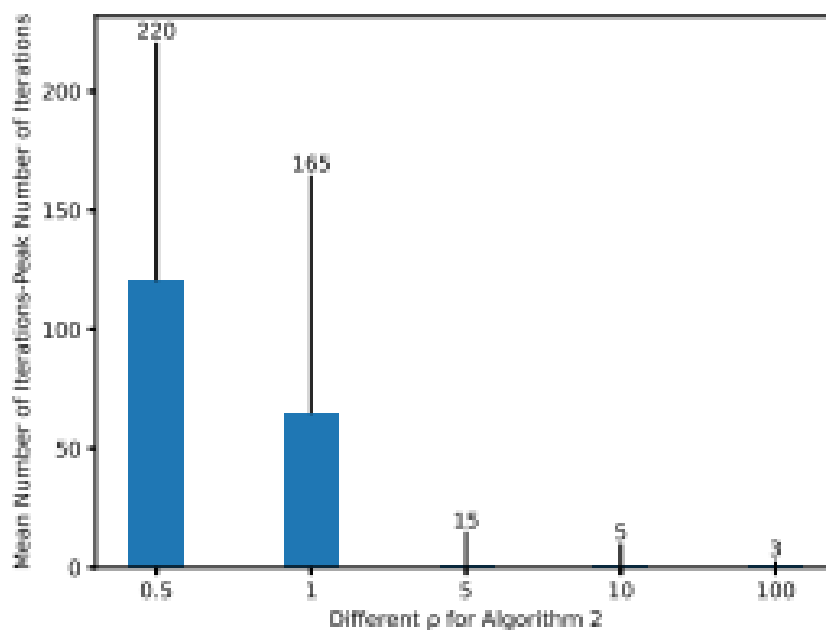
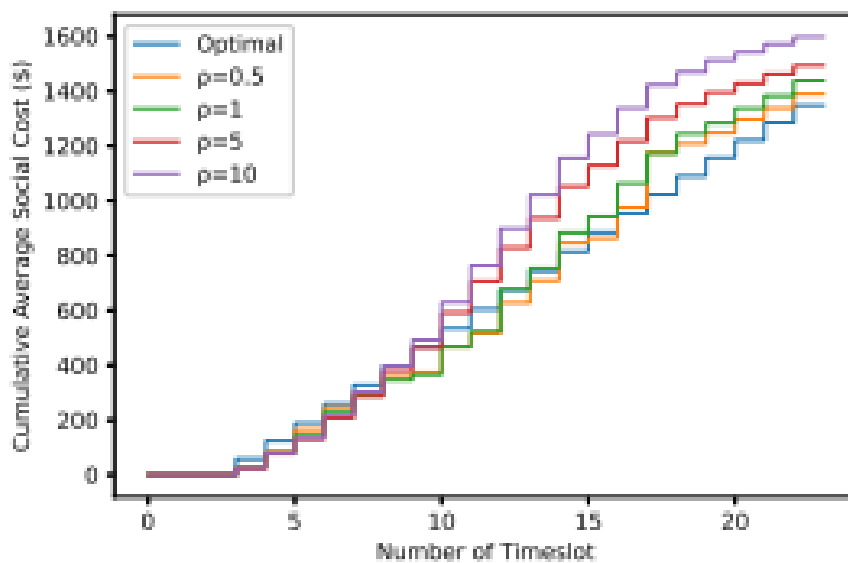


Fig. 7: Error ratio as a function of the error in the Poisson factor.



(a) Mean and peak number of iterations for different ρ



(b) Accumulated system cost for different values of ρ

Fig. 8: Effects of the choice of the ρ parameter of ADMM

4 ΕΦΑΡΜΟΓΗ ΑΛΓΟΡΙΘΜΟΥ ΣΕ ΣΥΣΤΗΜΑ ΗΛΕΚΤΡΙΚΗΣ ΕΝΕΡΓΕΙΑΣ

Ονοματολογία μεταβλητών παραμέτρων και συνόλων συστήματος

N	Σύνολο ενεργειακών πόρων
T	Σύνολο timeslots
B	Σύνολο κόμβων ενεργειακού δικτύου διανομής
C_b	Σύνολο γειτονικών κόμβων, κόμβου b
E_n	Σύνολο χαρακτηριστικών ηλεκτρικών αυτοκινήτων
U	Σύνολο μεταβλητών ελέγχου
W_t	Σύνολο τυχαίων μεταβλητών στο timeslot t
S_t	Σύνολο μεταβλητών κατάστασης(state) στο timeslot t
F_t	Σύνολο παραμέτρων άποψης(belief state) στο timeslot t
K	Σύνολο σεναρίων στον στοχαστικό προγραμματισμό
I	Σύνολο εισόδων στο Νευρωνικό δίκτυο
O	Σύνολο εξόδων στο νευρωνικό δίκτυο
$\overline{P}_n, \underline{P}_n$	Άνω ή κατω όριο στην παραγωγή ή κατανάλωση από κάποιο ενεργειακό πόρο
RU_n, RD_n	Ράμπα πάνω/κάτω γεννήτριας
g_n	Κόστος καυσίμου γεννήτριας ανα μονάδα
$\overline{E}_n, \underline{E}_n$	Άνω και κάτω όρια χωρητικότητας μπαταρίας
$\overline{V}, \underline{V}$	Άνω και κάτω όρια τάσης
$I_{z_{bb}}$	Πάνω όριο ρεύματος σε γραμμή

$\lambda_{b,t}$	Πολλαπλασιαστής lagrange για τον περιορισμό ισορροπίας ενεργού ισχύος
$\mu_{b,t}$	Πολλαπλασιαστής lagrange για τον περιορισμό ισορροπίας άεργου ισχύος
ρ	Παράμετρος του αλγόριθμου ADMM
$x_{n,t}$	Μεταβλητή ελέγχου αποστολής πόρων.
$Q_{n,t}$	Άεργος ισχύς που διοχετεύεται μέσω του πόρου n
y_n	Ενέργεια που δεν δίνεται στο EV n
$P_{ij,t}$	Ροή ενεργούς ισχύος από κόμβο i στον j στο timeslot t
$Q_{ij,t}$	Ροή άεργου ισχύος από κόμβο i στον j στο timeslot t
$I_{bc,t}^{sqr}$	Τετράγωνο του μέτρου του ρεύματος που διοχετεύεται από τον κόμβο b στον c στο timeslot t
$V_{bc,t}^{sqr}$	Τετράγωνο του μέτρου της τάσης του κόμβου b στον c στο timeslot t
$\widetilde{G}_{n,t}$	Παραγωγή διεσπαρμένων ενεργειακών μονάδων το timeslot t
\widetilde{t}_n^{arr}	Χρόνος άφιξης ηλεκτρικού αυτοκινήτου
\widetilde{t}_n^{dep}	Χρόνος αποχώρησης ηλεκτρικού αυτοκινήτου
\widetilde{E}_n^{arr}	Αρχικό επίπεδο μπαταρίας ηλεκτρικού αυτοκινήτου
\widetilde{C}_n	Χωρητικότητα μπαταρίας ηλεκτρικού αυτοκινήτου
\widetilde{E}_n^{des}	Τελική επιθυμητή κατάσταση μπαταρίας ηλεκτρικού αυτοκινήτου
\widetilde{p}_t	Τιμή ηλεκτρισμού το timeslot t
$\widetilde{D}_{n,t}$	Ζήτηση καταναλωτή n στο timeslot t

4.1 Μοντελοποίηση συστήματος

Θεωρούμε ένα σύνολο N από ευέλικτους ενεργειακούς πόρους. Κάθε πόρος ανήκει σε ένα συγκεκριμένο σύνολο N_θ τύπου $\theta \in \Theta$, όπου

$\Theta = \{Generatos, RES, Consumers, Storage, EVs\}$. Άρα $N = \cup_{\theta \in \Theta} N_\theta$. Ο συνεχόμενος χρόνος χωρίζεται σε διακριτά κομμάτια ίσου χρόνου για ένα χρονικό ορίζοντα T . Κάθε πόρος $n \in N$ χαρακτηρίζεται από άνω και κάτω όρια $\overline{P}_n, \underline{P}_n$ τα οποία είναι θετικά για πόρους που παράγουν ενέργεια ενώ αρνητικά για πόρους που καταναλώνουν ενέργεια. Έτσι ένας πόρος μπορεί να παράξει ενέργεια $x_{n,t}P_n$ έτσι ώστε

$$\overline{P}_n \leq x_{n,t}P_n \leq \underline{P}_n, \forall n \in N, t \in T \quad (1)$$

όπου το $x_{n,t} \in [0,1]$ είναι μεταβλητή απόφασης. Για ένα RES η μέγιστη παραγωγή του σε ένα timeslot είναι μια τυχαία μεταβλητή, περιορίζοντας την παραγωγή του

$$x_{n,t}\overline{P}_n \leq \widetilde{G}_{n,t} \forall n \in N_{RES}, t \in T \quad (2)$$

Οι περιορισμοί γεννητριών ramp-up και rump down είναι

$$RD_n \leq x_{n,t}\overline{P}_n - x_{n,t-1}\overline{P}_{n,t-1} \leq RU_n \quad \forall n \in N_{Gen}, t \in T \quad (3)$$

Η διάχυση άεργους ισχύος από καταναλωτές και RES, Θεωρούμε ότι έχουν σταθερό συντελεστή ισχύος

$$Q_{n,t} = x_{n,t}\overline{P}_n \tan(\cos^{-1}(pf_n)) \quad \forall n \in N_{Cons} \cup N_{RES} \quad (4)$$

Από την άλλη για τα EVs, τις μπαταρίες και τις γεννήτριες, θεωρούμε ότι η άεργος ισχύς διαφέρει ανάλογα με ένα μέγιστο συντελεστή ισχύος

$$Q_{n,t} \leq x_{n,t}\overline{P}_n \tan(\cos^{-1}(pf_n)) \quad \forall n \in N_{EVs} \cup N_{Sto} \cup N_{Gen} \quad (5)$$

Οι αποθηκευτήκες μονάδες και τα ηλεκτρικά αυτοκίνητα μπορούν να φορτίζονται και να αποφορτίζονται και άρα $\overline{P}_n \leq 0$ και $\underline{P}_n \geq 0$. Μια αποθηκευτική μονάδα(μπαταρία) έχει κάτω και πάνω όρια $\overline{E}_n, \underline{E}_n$ στην περιεκτικότητα ενέργειας

$$\overline{E}_n \leq E_{n,0} + \sum_{\tau \in [1,t]} x_{n,\tau}\overline{P}_n \leq \underline{E}_n \quad \forall t \in T, n \in N_{Sto} \quad (6)$$

όπου το $E_{n,0}$ είναι το αρχικό επίπεδο ενέργειας στην μπαταρία. Θεωρούμε επίσης ότι το τελικό επίπεδο ενέργειας στο τέλος του χρονικού ορίζοντα είναι ίσο με το επίπεδο ενέργειας στο αρχικό

$$\sum_{t \in T} x_{n,t} \bar{P}_n = 0 \quad \forall n \in N_{Sto} \quad (7)$$

Ένα EV, όμως χαρακτηρίζεται από τους χρόνους άφιξης και αναχώρησης τους $t_n^{\widetilde{arr}}, t_n^{\widetilde{dep}}$, το αρχικό επίπεδο φόρτισης $\widetilde{E}_n^{\widetilde{arr}}$, την χωρητικότητα μπαταρίας \widetilde{C}_n και την επιθυμητή τελική στάθμη τη μπαταρίας $\widetilde{E}_n^{\widetilde{des}}$. Οι αναφερόμενες παράμετροι είναι αβέβαιες μεταβλητές και αποκαλύπτονται στο σύστημα μόλις σταθμεύσει το ηλεκτρικό αυτοκίνητο.

$$0 \leq \widetilde{E}_n^{\widetilde{arr}} + \sum_{\tau \in [1,t]} x_{n,\tau} \bar{P}_n \leq \widetilde{C}_n \quad \forall t \in T, n \in N_{EVs} \quad (8)$$

Και η διαφορά μεταξύ της κατάστασης φόρτισης με την επιθυμητή κατάσταση φόρτισης του EV είναι

$$y_n = \widetilde{E}_n^{\widetilde{dep}} - \widetilde{E}_n^{\widetilde{arr}} + \sum_{\tau \in [t_n^{\widetilde{arr}}, t_n^{\widetilde{dep}}]} x_{n,\tau} \bar{P}_n \leq \widetilde{C}_n \quad \forall n \in N_{EVs} \quad (9)$$

Επίσης για λόγους ακρίβειας σημειώνουμε:

$$x_{n,t} = 0 \quad \forall n \in N_{EVs}, t \notin [t_n^{\widetilde{arr}}, t_n^{\widetilde{dep}}] \quad (10)$$

Κάθε ενεργειακός πόρος μπορεί να ελέγξει το δικό της ενεργειακό profile $x_n \triangleq (x_{n,t})_{t \in T}$ και το κόστος κάθε πόρου δίνεται από την συνάρτηση κόστους $c_n(x_n)$. Όλες οι συναρτήσεις κόστους θεωρούνται κυρτές. Συγκεκριμένα για τις γεννήτριες είναι

$$c_{n:n \in N_{Gen}}(x_n) = \sum_{t \in T} g_n(x_{n,t} \bar{P}_n)^2 - \tilde{p}_t(x_{n,t} \bar{P}_n) \quad (11)$$

Όπου g_n συσχετίζεται με την τιμή του καυσίμου και την λιανική τιμή ηλεκτρικής ενέργειας (τυχαία μεταβλητή).

Ένα RES έχει μηδενικά λειτουργικά κόστη και άρα έχει κέρδος ίσο με την λιανική τιμή ηλεκτρικής ενέργειας

$$c_{n:n \in N_{RES}}(x_n) = \sum_{t \in T} -\tilde{p}_t(x_{n,t} \bar{P}_n) \quad (12)$$

Οι αποθηκευτικές μονάδες έχουν κέρδος που εξαρτάται από την λιανική τιμή ηλεκτρικής ενέργειας, το κόστος από την εκφόρτιση της αποθηκευτικής ικανότητας βάσει του αριθμού πλήρων κύκλων φόρτισης $\left(\frac{x_{n,t} \bar{P}_n}{E_n}\right)$

$$c_{n:n \in N_{STO}}(x_n) = \sum_{t \in T} \left(d \left(\frac{x_{n,t} \bar{P}_n}{E_n} \right)^2 - \tilde{p}_t(x_{n,t} \bar{P}_n) \right) \quad (13)$$

Τα EVs έχουν την ίδια μορφή συνάρτησης κόστους με τις αποθηκευτικές μονάδες. Στο κόστος αυτό όμως προστίθεται ένας όρος τιμωρίας ανάλογα με το πότε φορτίστηκαν σε σχέση με τον επιθυμητό χρόνο αναχώρησης

$$c_{n:n \in N_{EVS}}(x_n) = \sum_{t \in T} \left(d \left(\frac{x_{n,t} \bar{P}_n}{\bar{C}_n} \right)^2 - \tilde{p}_t(x_{n,t} \bar{P}_n) \right) + w_1 y_n \quad (14)$$

Τέλος οι καταναλωτές έχουν ένα επίπεδο ζήτησης $\widetilde{D}_{n,t}$ σε κάθε timeslot t , άρα η συνάρτηση κόστους τους προκύπτει από το

$$c_{n:n \in N_{Cons}}(x_n) = \sum_{t \in T} \left(\tilde{p}_t(x_{n,t} \bar{P}_n) + w_2 (\widetilde{D}_{n,t} - x_{n,t} \bar{P}_n)^2 \right) + w_3 \left(\sum_{t \in T} \widetilde{D}_{n,t} - \sum_{t \in T} x_{n,t} \bar{P}_n \right)^2 \quad (15)$$

4.1.1 Δίκτυο διανομής

Οι ενεργειακοί πόροι είναι συνδεδεμένοι με ένα radial δίκτυο που έχει σύνολο κόμβων B και γραμμές που τους συνδέουν. Για κάθε $b \in B$, δηλώνουμε τους πόρους που συνδέονται στο b σαν N_b , τον κόμβο πατέρα τους σαν ζ_b και το σύνολο των παιδιών του κόμβου με C_b . Η ισορροπία ενεργούς ισχυος επιτυγχάνεται με την τήρηση της

$$P_{\zeta_b b} + \sum_{n \in N_b} x_{n,t} \bar{P}_n - \sum_{c \in C_b} (P_{bc,t} + R_{bc} I_{bc,t}^{sqr}) = 0 \quad (16)$$

Και για την ισορροπία αέργου ισχύς η

$$Q_{\zeta_b b} + \sum_{n \in N_b} Q_{n,t} - \sum_{c \in C_b} (Q_{bc,t} + X_{bc} I_{bc,t}^{sqr}) = 0 \quad (17)$$

Το μέτρο της τάσης μεταξύ των κόμβων του ζ_b και b παρουσιάζεται σαν

$$V_{\zeta_b, t}^{sqr} - 2 \left(R_{\zeta_b} P_{\zeta_b} b, t + X_{\zeta_b} Q_{\zeta_b} b, t \right) - \left(R_{\zeta_b}^2 b + X_{\zeta_b}^2 b \right) I_{\zeta_b, t}^{sqr} \quad (18)$$

Καθώς η διαχυόμενη ισχύς στις γραμμές υπολογίζεται από την ανισότητα

$$V_{b,t} I_{\zeta_b} b, t \leq R_{\zeta_b}^2 b, t + Q_{\zeta_b}^2 b, t, \quad \forall b \in B, t \in T. \quad (19)$$

Τέλος υπάρχουν άνω και κάτω όρια για το μέτρο της τάσης και της έντασης.

$$\underline{V} \leq V_{b,t} \leq \bar{V} \quad \forall b \in B, t \in T. \quad (20)$$

$$0 \leq I_{\zeta_b} b, t \leq \bar{I}_{\zeta_b} b \quad \forall b \in B, t \in T. \quad (21)$$

4.2 Παρουσίαση μοντελοποίησης προβλήματος

Όταν δεν υπάρχει αβεβαιότητα η λύση μπορεί να δοθεί από το πρόβλημα βέλτιστης ροής

$$\min_u \sum_{n \in N} c_n(x_n) \quad (22)$$

$$s. t. (1) - (21),$$

Όπου οι μεταβλητές απόφασης είναι

$$u = \{(x_{n,t}, Q_{n,t})_{n \in N, t \in T}, (y_n)_{n \in N}, (V_{b,t}, P_{\zeta_b} b, t, Q_{\zeta_b} b, t, I_{\zeta_b} b, t)_{b \in B, t \in T}\}.$$

$$W_t = \{(\tilde{G}_{n,t})_{n \in N_{RES}}, (\tilde{D}_{n,t})_{n \in N_{CONS}}, \tilde{p}_t\} \cup (E_n)_{n \in N_{EV}: t_n^{arr} = t} \quad (23)$$

4.3 Αλγόριθμοι σύγκρισης

Για να συγκρίνουμε την απόδοση του αλγορίθμου, τον συγκρίναμε με τους εξής αλγόριθμους:

1. Επαναληπτικός Στοχαστικός προγραμματισμός: Υπάρχουν πολλά σενάρια που μπορούν να συμβούν στο τέλος. Η πληροφορία για τις άγνωστες μεταβλητές σε ένα timeslot παρουσιάζεται στον αλγόριθμο μόνο όταν βρίσκεται σε εκείνο το timeslot.
2. Learn to optimize: Εκπαιδεύουμε ένα νευρωνικό δίκτυο ώστε να παίρνει σαν είσοδο την κατάσταση του συστήματος στον παρόν timeslot και να βγάξει σαν έξοδο τις μεταβλητές απόφασης του αλγορίθμου.
3. Το ντετερμινιστικό πρόβλημα, χωρίς αβεβαιότητα

4.4 Προτεινόμενος αλγόριθμος

Καταρχάς θεωρούμε ένα κατανεμημένο αλγόριθμο έχοντας τέλειες προβλέψεις. Θεωρούμε τις μεταβλητές της ισορροπίας ενεργούς και άεργους ισχύος, για κάθε κόμβο και κάθε timeslot.

$$\delta_{b,t}^p = P_{\zeta_{bb}} + \sum_{n \in N_b} x_{n,t} \bar{P}_n - \sum_{c \in C_b} (P_{bc,t} + R_{bc} I_{bc,t}^{sqr})$$

Και

$$\delta_{b,t}^q = Q_{\zeta_{bb}} + \sum_{n \in N_b} Q_{n,t} - \sum_{c \in C_b} (Q_{bc,t} + X_{bc} I_{bc,t}^{sqr})$$

Χαλαρώνοντας τους πιο πάνω περιορισμούς, μπορούμε να γράψουμε το χαλαρωμένο Lagrangian όρο

$$L = \sum_{n \in N} c_n(x_n) - \sum_{t \in T} \sum_{b \in B} \left(\lambda_{b,t} \delta_{b,t}^p - \frac{\rho}{2} (\delta_{b,t}^p)^2 \right) - \sum_{t \in T} \sum_{b \in B} \left(\lambda_{b,t} \delta_{b,t}^q - \frac{\rho}{2} (\delta_{b,t}^q)^2 \right)$$

όπου $\lambda_{b,t}, \mu_{b,t}$ είναι οι δυικοί πολλαπλασιαστές των περιορισμών ισορροπίας ενεργούς και άεργους ισχύος για b, t . Δεδομένου ότι η Langrangian εξίσωση είναι διαχωρίσιμη για κάθε κόμβο, το πρόβλημα (22) μπορεί να γίνει αποκεντρωμένο, και έτσι ,για κάθε επανάληψη i , κάθε κόμβος ενημερώνει τις μεταβλητές απόφασης $U_b = ((x_{n,t}, Q_{n,t})_{n \in N_b, t \in T} (y_n)_{n \in N_b})$ έτσι ώστε

$$U_b^i \in \operatorname{argmin}_U \{L\} \quad (35)$$

$$\text{s. t. (1) – (15)}$$

$$U_{DSO} = U_{DSO}^{t-1}$$

Και για τον διαχειριστή δικτύου(DSO) η ενημέρωση των μεταβλητών απόφασης γίνεται:

$$U_b^i \in \operatorname{argmin}_U \{L\} \quad (36)$$

$$s. t. (18) - (21)$$

$$U_b = U_b^{t-1} \forall b \in B$$

Παίρνοντας τις ενημερώσεις μεταβλητών στα (35),(36) οι πολλαπλασιαστές Lagrange ενημερώνονται βάση του αλγορίθμου ADMM:

$$\lambda_{b,t}^{(i)} = \lambda_{b,t}^{(i-1)} + \rho \delta_{b,t}^p (U_b^{(i)}, U_{DSO}^{(i)}) \forall b \in B, t \in T$$

$$\mu_{b,t}^{(i)} = \mu_{b,t}^{(i-1)} + \rho \delta_{b,t}^q (U_b^{(i)}, U_{DSO}^{(i)}) \forall b \in B, t \in T$$

Για την επίλυση του πιο πάνω προβλήματος, γίνεται χρήση ενός νευρωσικού δικτύου για την, για την πρόβλεψη των δυικών πολλαπλασιαστών. Έτσι αν είμαστε στο timeslot $\tau \in T$, το νευρωνικό επιστρέφει τιμές $(\lambda_{b,t}^*, \mu_{b,t}^*)_{t \in [\tau+1, |T|]}$ που αντιστοιχούν στους περιορισμούς (16), (17) για το πρόβλημα (22). Έτσι στο timeslot τ της λειτουργίας του αλγορίθμου το νευρωνικό δίκτυο παίρνει σαν είσοδο το state

$$I_r = \{\tau, (x_{n,\tau-1})_{n \in N_{Gen}}, (E_{n,\tau-1})_{n \in N_{EVs} \cup N_{sto} \cup N_{Ncons}}, (W_t)_{t \in [1,\tau]}\}$$

Μαζί με τις αποφασισμένες δυικές μεταβλητές του προηγούμενου timeslot και προβλέπει τους μελλοντικούς δυικούς πολλαπλασιαστές. Έπειτα εκτελείτε ο ADMM, ενημερώνοντας μόνο τους πολλαπλασιαστές του τωρινού timeslot, ενώ οι υπόλοιποι μελλοντικοί πολλαπλασιαστές μένουν σταθεροί. Σαν αποτέλεσμα, ο αλγόριθμος συγκλίνει σε αποτέλεσμα που σέβεται τους περιορισμούς συστήματος και παράλληλα υπολογίζει την αβεβαιότητα. Ο ακριβής αλγόριθμος είναι

Algorithm 1 The proposed policy for the data-driven, distributed sequential OPF problem at stage τ .

- 1: Feed input to the NN, and obtain $\lambda_{b,t}^*, \mu_{b,t}^*$, for each $b \in \mathcal{B}, t \in [\tau + 1, |\mathcal{T}|]$
- 2: Set $\lambda_{b,t}, \mu_{b,t} = \lambda_{b,t}^*, \mu_{b,t}^*, \forall b \in \mathcal{B}, t \in [\tau + 1, |\mathcal{T}|]$.
- 3: Initialize $i = 0, \lambda_{b,\tau}^{(0)}, \mu_{b,\tau}^{(0)} = 0, \forall b \in \mathcal{B}$.
- 4: **repeat**:
- 5: $i = i + 1$
- 6: **for** $b \in \mathcal{B}$:
- 7: Update $\mathcal{U}_b^{(i)}$ by solving (35)
- 8: Update $\mathcal{U}_{\text{DSO}}^{(i)}$ by solving (36)
- 9: Set:
 $\lambda_{b,\tau}^{(i)} = \lambda_{b,\tau}^{(i-1)} + \rho \delta_{b,\tau}^{\text{P}}(\mathcal{U}_b^{(i)}, \mathcal{U}_{\text{DSO}}^{(i)}), \forall b \in \mathcal{B},$
 $\mu_{b,\tau}^{(i)} = \mu_{b,\tau}^{(i-1)} + \rho \delta_{b,\tau}^{\text{Q}}(\mathcal{U}_b^{(i)}, \mathcal{U}_{\text{DSO}}^{(i)}) \forall b \in \mathcal{B}.$
- 10: **until** $\max_{b \in \mathcal{B}} \{\lambda_{b,\tau}^{(i)} - \lambda_{b,\tau}^{(i-1)}\} < \varepsilon$
AND
 $\max_{b \in \mathcal{B}} \{\mu_{b,\tau}^{(i)} - \mu_{b,\tau}^{(i-1)}\} < \varepsilon$
- 11: **apply**: $(x_{n,\tau}^{(i)}, Q_{n,\tau}^{(i)})_{n \in \mathcal{N}_i}$

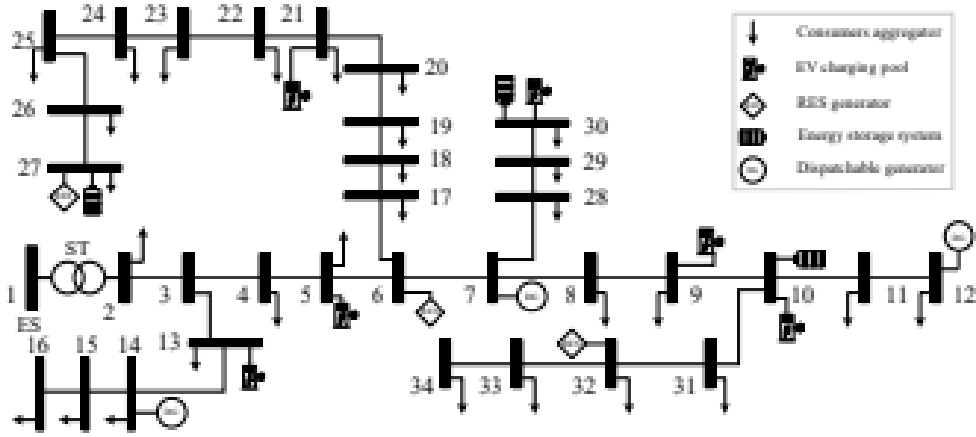


Fig. 1: 34-nodes test system including four charging pools.

4.5 Πειραματική εκτίμηση

Σχεδίαση πειράματος

Οι παρουσιασμένες πολιτικές εκτιμώνται σε ορίζοντα 24 timeslots, σε ένα σύστημα διανομής 11 kV MV. Η τάση υποσταθμού είναι ίση με 1 p.u. και τα όρια τάσης είναι $\bar{V} = 1.05$, $\underline{V} = 0.95$. Στους κόμβους 7,12 και 14 έχουμε τους γεννήτριες, στους κόμβους 6,27 και 32 υπάρχουν αποκεντρωμένοι ενεργειακοί πόροι. Επίσης στους κόμβους 10,27 και 30 υπάρχουν αποθηκευτικές εγκαταστάσεις και στους 5,9,10,13,21 και 30. Τέλος οι καταναλωτές βρίσκονται σε όλους τους κόμβους του δικτύου.

Τα datasets που χρησιμοποιήθηκαν είναι \cite

Για το πείραμα αυτό, έχει χρησιμοποιηθεί Νευρωνικό δίκτυο Γράφων(GNN), για να υπολογίσουμε τις δυικές μεταβλητές. Τα GNNs είναι ευέλικτα στην αλλαγή κόμβων και ακμών καθώς μπορεί να προβλέψει τις τιμές λαμβάνοντας υπόψη τις χωρικές εξαρτήσεις. Συγκεκριμένα υλοποιήσαμε Νευρωνικά δίκτυα συνέλιξης γράφων με 5 επίπεδα, και εκπαιδεύτηκε για 300 εποχές.

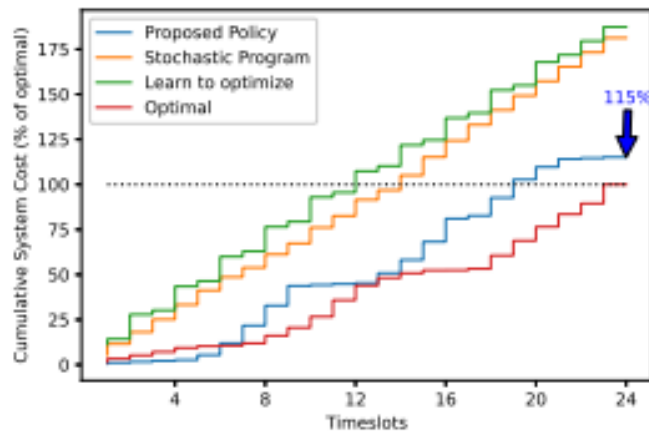


Fig. 2: Comparison of the three policies with the optimal-in-hindsight solution.

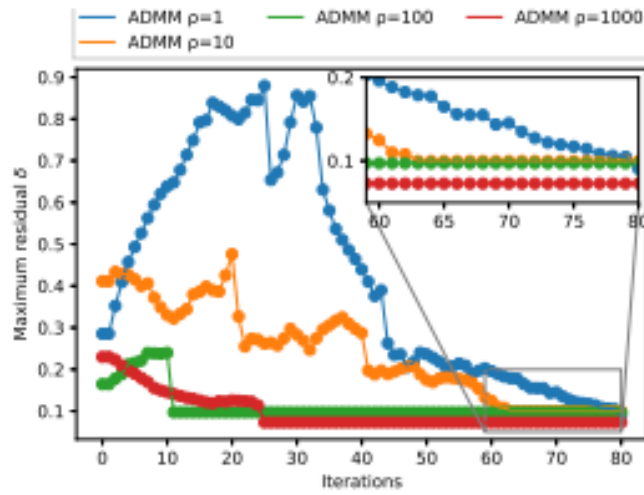


Fig. 3: Convergence behavior of the proposed policy for one decision and for different values of parameter ρ .

4.6 Εκτίμηση Αποτελεσμάτων

Το κύριο αποτέλεσμα είναι η αντιπαραβολή της προτεινομένης πολιτικής του προτεινόμενου αλγορίθμου ενάντια της βέλτιστης εκ των υστέρων λύσης του προβλήματος και ενάντια στις 2 πολιτικές αναφοράς(επαναληπτικού ορίζοντα στοχαστικός προγραμματισμός and the learn to optimize πολιτική). Η παρουσιαζόμενη σαν βέλτιστη λύση μας παρέχει την βέλτιστη τιμή η οποία μπορεί μόνο να επιτευχθεί αν όλες οι πληροφορίες είναι γνωστές εκ των προτέρων και αν δεν είναι εφικτό στη πράξη. Παρόλα αυτά , αφυπηρετούν ως θεωρητικά σημεία αναφοράς έναντι του οποίου μπορούμε να υποθέσουμε τις επιδόσεις των 3 πολιτικών. Στη γραφική παράσταση 2 παρουσιάζεται το κόστος συστήματος κάθε πολιτικής που συσσωρεύεται κατά του χρονικού ορίζοντα, όπου η τιμή στην τελευταία χρονική στιγμή($t=24$) εκφράζει την ολική επίδοση της πολιτικής ως ποσοστό του κόστους της βέλτιστης εκ των υστέρων λύσης. Όπως μπορεί να παρατηρηθεί η προτεινομένη πολιτική υπερτερεί σημαντικά των 2 δεικτών αναφοράς επιτυγχάνοντας κόστος που είναι μόνο 15% μεγαλύτερο από αυτή της τέλει πληροφοριακής υπόθεσης.

Η επίδοση που απεικονίζεται στην γραφική παράσταση 2 επιτυγχάνεται για την προτεινομένη πολιτική θέτοντας ρ ίσο με 1. Ο απαιτούμενος υπολογιστικός χρόνος που απαιτείται για να παρθούν αποφάσεις ήταν της τάξεως του ενός λεπτού η οποία επικυρώνει την ακαταλληλότητα της πολιτικής για αποφάσεις σε πραγματικό χρόνο. Μεγαλύτερες τιμές του ρ παρέχουν ακόμα πιο γρήγορους χρόνους σύγκλισης όπως παρατηρείται στη γραφική παράσταση 3, παρόλη την υψηλότερη δαπάνη του κόστους του συστήματος (απώλεια απόδοσης) η οποία είναι αξιοσημείωτη όπως φαίνεται στη γραφική παράσταση 4.

Οι ελκυστικές επιδόσεις της πολιτικής μπορούν να αποδοθούν στον συνδυασμό της μονάδας πρόβλεψης GNN και της προσαρμοστικής εννοίας του αλγόριθμου ADMM επαναλαμβανομένου ορίζοντα. Η γραφική παράσταση 5 αποδεικνύει αυτήν την εξήγηση συγκρίνοντας τους πολλαπλασιαστές βέλτιστης προοπτικής με τους πολλαπλασιαστές για timeslots που προβλέπονται μόνο από το GNN καθώς και με τους τελικούς πολλαπλασιαστές που προκύπτουν από τον Αλγόριθμο 1. Όπως φαίνεται οι προτεινόμενοι πολλαπλασιαστές της πολιτικής είναι αρκετά κοντά στους πολλαπλασιαστές βέλτιστης προοπτικής. Η έξοδος GNN βοηθά σε αυτό αποτυπώνοντας το μοτίβο των μελλοντικών πολλαπλασιαστών, αν και με μια τάση υποτίμησής σε αυτό το παράδειγμα.

Η τάση υποτίμησης την πολλαπλασιαστών μπορεί να δικαιολογηθεί από τα κατασκευαστικά δεδομένα που δημιουργούνται από την μέθοδο παραγωγής δεδομένων. Όπως φαίνεται στην γραφική παράσταση 6 , η μέθοδος παραγωγής δεδομένων αποτυπώνει τις κυρίες στατιστικές ιδιότητες της πραγματικής κατανομής δεδομένων(για παραγωγή RSE), αλλά αποτυχαίνει να αποτυπώσει τις ακραίες τιμές της υψηλής πλευράς. Αυτό σημαίνει ότι το σύνολο των δεδομένων εκπαίδευσης GNN αγνοούσε ως επί το πλείστον αυτά τα σχετικά σπάνια αλλά πιθανά σενάρια.

Στην συνέχεια εκτιμούμε την ικανότητα των GNN να προβλέπουν προκατειλημμένα δεδομένα. Γι' αυτόν το σκοπό οι επιδόσεις των GNN εκτιμήθηκαν για υποθέσεις κατανομής δεδομένων που διαφέρουν από την κατανομής δεδομένων εκπαίδευσης κατά εάν ορισμένο ποσοστό(από -15% έως +15%). Το παραγόμενο μέσο τετραγωνισμένο σφάλμα φαίνεται στη γραφική παράσταση 8.

Τέλος, αποδεικνύουμε εμπειρικά μια παραβίαση περιορισμών που μπορεί να προκύψει από την πολιτική εκμάθησης βελτιστοποίησης. Η γραφική παράσταση 9 παρουσιάζει την τάση του κόμβου 27 για την προτεινόμενη και learn-to-optimize πολιτικής για ένα τυπικό επιχειρησιακό σενάριο. Η προτεινόμενη πολιτική κρατά την τάση μέσα σε ασφαλή όρια, ενώ η learn-to-optimize πολιτική μπορεί να παραβιάσει τα όρια σημαντικά.

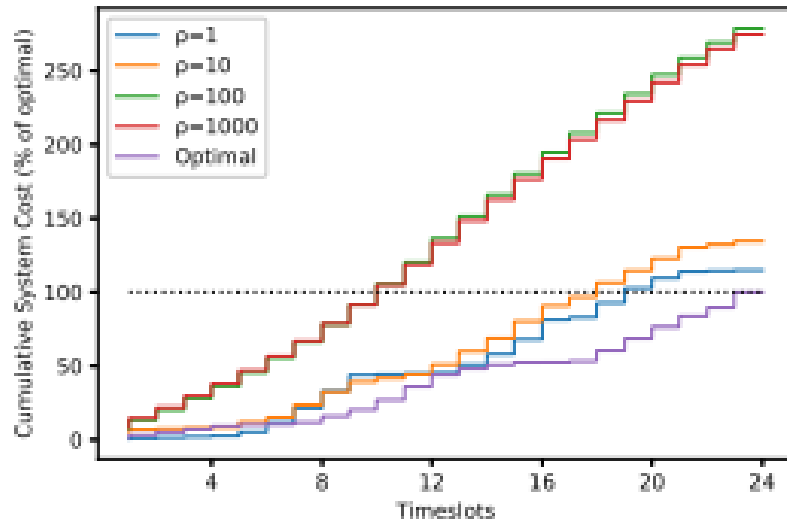


Fig. 4: Efficiency loss for different values of parameter ρ .

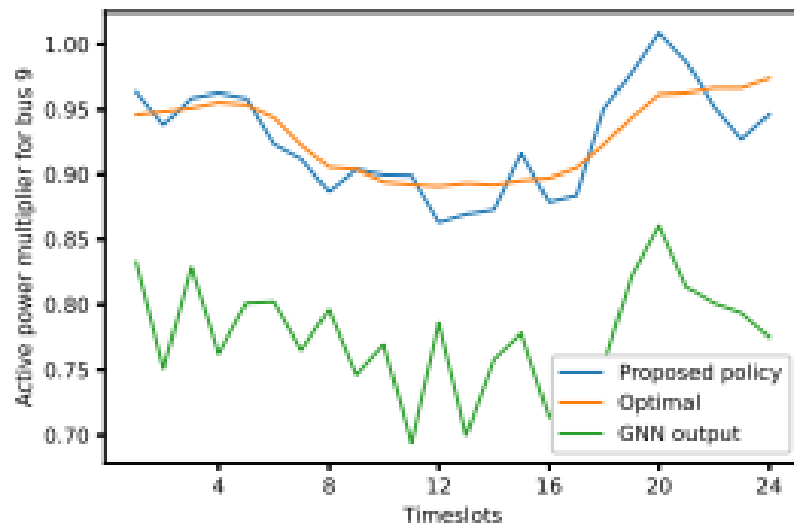


Fig. 5: Active power multipliers of the proposed policy, compared to the optimal-in-hindsight multipliers and the GNN output.

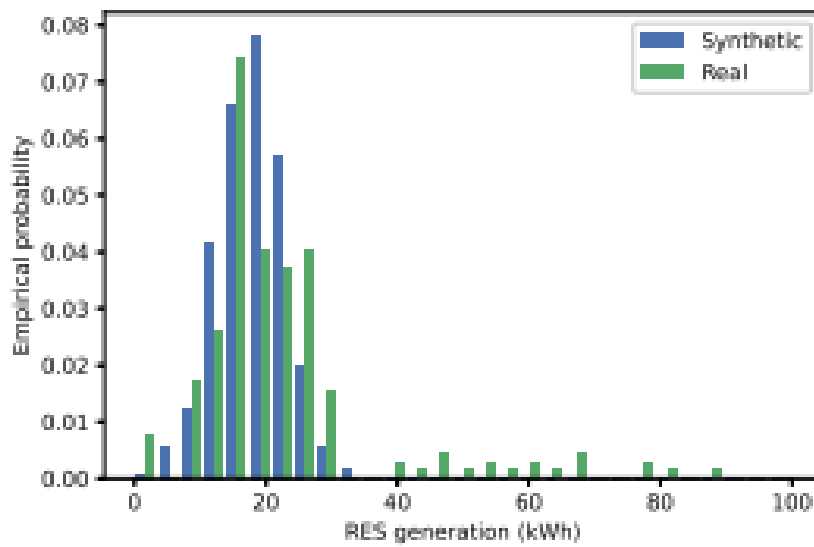


Fig. 6: Fitted distribution of synthetic data vs the empirical distribution of real data for RES generation

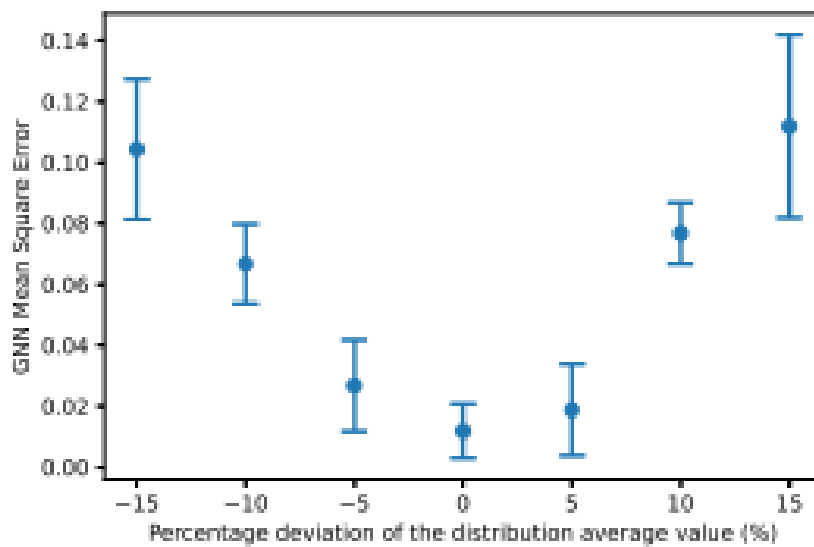


Fig. 7: Mean Square Error of the GNN as a function of the degree of bias in the training dataset.

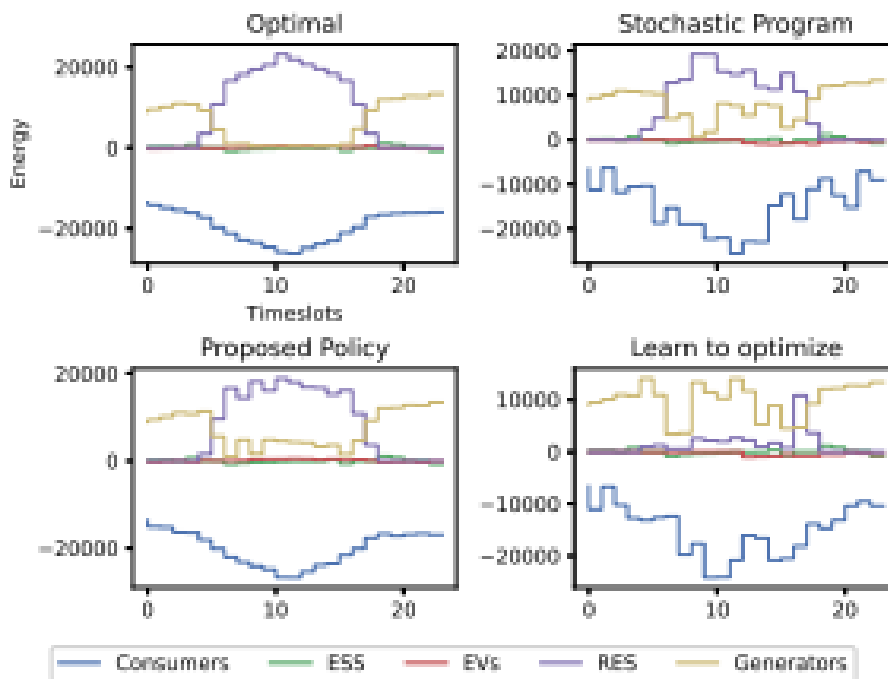


Fig. 8: Mean Square Error of the GNN as a function of the degree of bias in the training dataset.

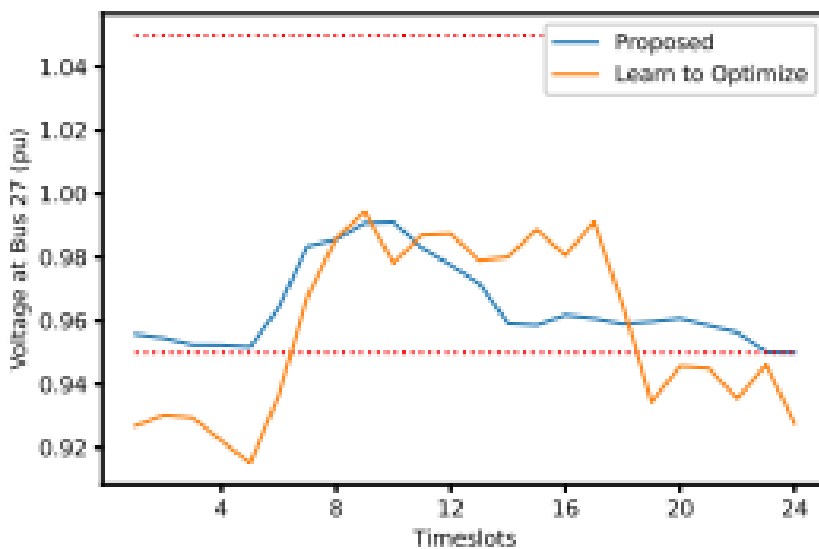


Fig. 9: Voltage of node 27 for the proposed and the learn-to-optimize policies.

III. ΕΠΙΛΟΓΟΣ

5. ΕΠΙΛΟΓΟΣ

5.1 Συμπεράσματα

Ο αλγόριθμος που παρουσιάστηκε εφαρμόστηκε σε 2 ξεχωριστά προβλήματα στον ενεργειακό τομέα. Το ένα αφορά το Peer-to-Peer Energy και το άλλο ένα δίκτυο διανομής ηλεκτρικής ενέργειας με υψηλή διείσδυση από ευέλικτους κατανεμημένους πόρους. Αποδεικνύεται πειραματικά ότι ο αλγόριθμος είναι ευέλικτος στις αλλαγές των παραμέτρων και των κατανομών των τυχαίων μεταβλητών. Επίσης φαίνεται ότι η πολιτική μας δίνει λύσεις κοντά στις βέλτιστες, σε πραγματικό χρόνο. Αυτό επιτυγχάνεται με την μείωση της αβεβαιότητας προβλέποντας τις τιμές των δικών πολλαπλασιαστών και όχι των μεταβλητών απόφασης. Τέλος ο προτεινόμενος αλγόριθμος αποδεικνύεται πολύ πιο γρήγορος από τις σύγχρονες μεθόδους που χρησιμοποιούνται όπως ο στοχαστικός προγραμματισμός [18], δίνοντας παράλληλα αποτελέσματα που σέβονται τους περιορισμούς του συστήματος. Το γεγονός αυτό τον κάνει ικανό να υλοποιηθεί και να δοκιμαστεί σε πραγματικά συστήματα διανομής ηλεκτρικής ενέργειας.

5.2 Μελλοντικές Επεκτάσεις

Στο μέλλον θα μπορούσε να διερευνηθεί καταρχάς η χρήση πιο εξελιγμένων αλγορίθμων ADMM όπως αυτούς που παρουσιάζονται στα [19], [20] καθώς υπάρχουν αλγόριθμοι που παρουσιάζουν βελτιωμένους ρυθμούς σύγκλισης με περισσότερη ακρίβεια. Συμπληρωματικά μπορούν να χρησιμοποιηθούν πιο εξελιγμένα μοντέλα νευρωνικών δικτύων όπως δυναμικά νευρωνικά δίκτυα γράφων [21], ώστε το σύστημα να είναι πιο ευέλικτο στις αλλαγές των παραμέτρων. Επίσης θα μπορούσε να διερευνηθεί διαφορετική εκπαίδευση των μοντέλων νευρωνικών δικτύων γι' αυτό τον σκοπό όπως στο [22]. Τέλος η χρήση του αλγορίθμου σε πραγματικό σύστημα θα βοηθούσε στην περαιτέρω κατανόηση της δυναμικής του αλγορίθμου και στην πιο λεπτομερή ρύθμισή του ώστε να αποδίδει βέλτιστα σε οποιοσδήποτε συνθήκες.

Βιβλιογραφία

- [1] “Mathematical Optimization.”
<https://web.stanford.edu/group/sisl/k12/optimization/#!index.md> (accessed Jan. 07, 2023).
- [2] “Convex Optimization,” IEEE Trans Automat Contr, vol. 51, no. 11, pp. 1859–1859, Nov. 2006, doi: 10.1109/TAC.2006.884922.
- [3] L. Lasdon, “Duality and Decomposition in Mathematical Programming,” IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 86–100, 1968, doi: 10.1109/TSSC.1968.300135.
- [4] S. Boyd, “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers,” Foundations and Trends® in Machine Learning, vol. 3, no. 1, pp. 1–122, 2010, doi: 10.1561/22000000016.
- [5] S. Walczak and N. Cerpa, “Artificial Neural Networks,” Encyclopedia of Physical Science and Technology, pp. 631–645, 2003, doi: 10.1016/B0-12-227410-5/00837-1.
- [6] E. Wilson and D. W. Tufts, “Multilayer perceptron design algorithm,” in Proceedings of IEEE Workshop on Neural Networks for Signal Processing, pp. 61–68. doi: 10.1109/NNSP.1994.366063.
- [7] F. Scarselli, M. Gori, Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini, “The Graph Neural Network Model,” IEEE Trans Neural Netw, vol. 20, no. 1, pp. 61–80, Jan. 2009, doi: 10.1109/TNN.2008.2005605.
- [8] W. B. Powell, “Sequential Decision Analytics and Modeling: Modeling with Python - Part I,” Foundations and Trends® in Technology, Information and Operations Management, vol. 15, no. 4, pp. 325–456, Nov. 2022, doi: 10.1561/0200000103-I.
- [9] W. B. Powell, “Sequential Decision Analytics and Modeling: Modeling with Python - Part II,” Foundations and Trends® in Technology, Information and Operations Management, vol. 16, no. 1–2, pp. 1–176, Nov. 2022, doi: 10.1561/0200000103-II.

- [10] W. B. Powell, “From Reinforcement Learning to Optimal Control: A Unified Framework for Sequential Decisions,” *Studies in Systems, Decision and Control*, vol. 325, pp. 29–74, 2021, doi: 10.1007/978-3-030-60990-0_3.
- [11] T. Halder, “A smart grid,” in 2014 6th IEEE Power India International Conference (PIICON), Dec. 2014, pp. 1–6. doi: 10.1109/POWERI.2014.7117674.
- [12] “Europe - Frequently Asked Questions.” https://intellectual-property-helpdesk.ec.europa.eu/regional-helpdesks/european-ip-helpdesk/europe-frequently-asked-questions_en (accessed Jan. 07, 2023).
- [13] “Grid Modernization and the Smart Grid | Department of Energy.” <https://www.energy.gov/oe/activities/technology-development/grid-modernization-and-smart-grid> (accessed Jan. 07, 2023).
- [14] J. C. Stephens, E. J. Wilson, and T. R. Peterson, “Smart grid (R)evolution: Electric power struggles,” *Smart Grid (R)Evolution: Electric Power Struggles*, pp. 1–198, Jan. 2014, doi: 10.1017/CBO9781107239029.
- [15] A. Paudel, L. P. M. I. Sampath, J. Yang, and H. B. Gooi, “Peer-to-Peer Energy Trading in Smart Grid Considering Power Losses and Network Fees,” *IEEE Trans Smart Grid*, vol. 11, no. 6, pp. 4727–4737, Nov. 2020, doi: 10.1109/TSG.2020.2997956.
- [16] W. Tushar, T. K. Saha, C. Yuen, D. Smith, and H. V. Poor, “Peer-to-Peer Trading in Electricity Networks: An Overview,” *IEEE Trans Smart Grid*, vol. 11, no. 4, pp. 3185–3200, Jul. 2020, doi: 10.1109/TSG.2020.2969657.
- [17] D. P. Bertsekas, Bertsekas, and D. P., *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.
- [18] L. A. Roald, D. Pozo, A. Papavasiliou, D. K. Molzahn, J. Kazempour, and A. Conejo, “Power systems optimization under uncertainty: A review of methods and applications,” *Electric Power Systems Research*, vol. 214, Jan. 2023, doi: 10.1016/j.epsr.2022.108725.
- [19] C. Song, S. Yoon, and V. Pavlovic, “Fast ADMM Algorithm for Distributed Optimization with Adaptive Penalty,” 30th AAAI Conference on Artificial Intelligence, AAAI 2016, pp. 753–759, Jun. 2015, doi: 10.48550/arxiv.1506.08928.

- [20] Y. Yang, X. Guan, Q.-S. Jia, L. Yu, B. Xu, and C. J. Spanos, “A Survey of ADMM Variants for Distributed Optimization: Problems, Algorithms and Features,” Aug. 2022, Accessed: Jan. 07, 2023. [Online]. Available: <https://arxiv.org/abs/2208.03700v2>
- [21] J. Xie, Y. Liu, and Y. Shen, “Explaining Dynamic Graph Neural Networks via Relevance Back-propagation,” Jul. 2022, doi: 10.48550/arxiv.2207.11175.
- [22] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, “Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization,” Nov. 2019, doi: 10.48550/arxiv.1911.08731.
- [23] “Results of the greek balancing market,” <https://www.admie.gr/en/market/market-statistics/detail-data>, accessed: 10 - 2021.
- [24] “Vimsen project repository,” [https://github.com/vimsen/dss/tree/master/data points](https://github.com/vimsen/dss/tree/master/data%20points), accessed: 10 - 2021.
- [25] “Adaptive charging network,” <https://ev.caltech.edu/dataset>, accessed: 10- 2021.

Απόδοση ξενόγλωσσων όρων

- Smart Grid – Έξυπνο Δίκτυο
- Multi-Layer Perceptron (MLP) - Πολυστρωματικό Νευρωνικό Δίκτυο
- Graph Neural Networks (GNN) – Νευρωνικό Δίκτυο γράφων
- Stochastic Programming – Στοχαστικός προγραμματισμός
- Duality – Διαικότητα
- Convex – Κυρτό
- Alternating Direction Method of Multipliers (ADMM) – Μέθοδος αποσύνθεσης προβλημάτων βελτιστοποίησης.
- Policy – Πολιτική
- Learn to optimize – Μέθοδος κατά την οποία ένα μαύρο κουτί δέχεται σαν είσοδο την κατάσταση του συστήματος και εξάγει τις απαιτούμενες ενέργειες ελέγχου.

