

**ΟΙΚΟΝΟΜΙΚΟ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΑΘΗΝΩΝ**



**ATHENS UNIVERSITY
OF ECONOMICS
AND BUSINESS**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΜΕΤΑΠΤΥΧΙΑΚΟ ΔΙΠΛΩΜΑ
ΕΙΔΙΚΕΥΣΗΣ (MSc)
στην ΑΝΑΠΤΥΞΗ & ΑΣΦΑΛΕΙΑ
ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**“Αλγόριθμοι απόφασης πολλαπλών νικητών για μια
ευέλικτη αποκεντρωμένη αγορά ενέργειας”**

ΜΙΧΑΣ ΣΠΥΡΙΔΩΝ

P3312116

ΑΘΗΝΑ, ΣΕΠΤΕΜΒΡΙΟΣ 2023

Περιεχόμενα

1.	Εισαγωγή	14
1.1	Στόχος της διπλωματικής εργασίας.....	14
1.2	Δομή της εργασίας	14
2.	Αγορά της Ενέργειας.....	17
2.1	Μεταφορά ηλεκτρικής ενέργειας και ο ρόλος του διαχειριστή	23
2.2	Αιτίες δημιουργίας ανισορροπιών στο δίκτυο.....	24
2.3	Τρόποι εξισορρόπησης δικτύου.....	25
2.4	Διαχείριση συμφόρησης δικτύου (αλγόριθμοι case studies χωρών).....	27
3.	Ηλεκτροκίνηση, Decentralized Energy Flexibility Marketplace και Unit Commitment πρόβλημα.....	30
3.1	Ηλεκτροκίνηση σήμερα και παρουσίαση Ευέλικτης Αποκεντρωμένης Αγοράς.....	30
3.2	Έννοιες και μεταβλητές προβλήματος UC.....	36
3.3	Αντικειμενική συνάρτηση προβλήματος UC και αλγόριθμοι επίλυσης.....	38
3.3.1	Dynamic Programming (DP).....	41
3.3.2	Mixed- Integer Linear Programming (MILP).....	41
3.3.3	Particle Swarm Optimization (PSO)	42
3.3.4	Lagrangian Relaxation (LR).....	44
4.	Παρουσίαση δεδομένων	47
4.1	Δεδομένα σταθμών φόρτισης ηλεκτρικών αυτοκινήτων	47
4.2	Ηλεκτρικά οχήματα στόλων	48
4.3	Προβλέψεις διαχειριστή δικτύου	49
4.4	Δημοπρασίες.....	50
4.5	Προσφορές EVC aggregator (bids).....	51
5.	Αποτελέσματα αλγορίθμων	53
5.1	Dynamic Programming (DP).....	53
5.1.1	Επίλυση με χρήση του αλγορίθμου knapsack – Dynamic Programming (DP)	54
5.1.2	Επίλυση με χρήση του αλγορίθμου knapsack με βάρη -Backtracking (BR)	56
5.2	Mixed-integer Linear Programming (MILP).....	58
5.3	Particle Swarm Optimization (PSO)	60
5.4	Lagrangian Relaxation (LR).....	64
6.	Συμπεράσματα	68

6.1	Σύγκριση επιδόσεων των αλγορίθμων σχετικά με την αθροιστική παραγωγή και τα κόστη των λύσεων.....	68
6.2	Σύγκριση επιδόσεων των αλγορίθμων σχετικά με τον χρόνο εκτέλεσής τους 72	
6.3	Τελικά συμπεράσματα.....	73
7.	Μελλοντική εργασία	74
	Βιβλιογραφία	76

Περίληψη

Η ηλεκτρική ενέργεια αποτελεί ένα από τα σημαντικότερα αγαθά της σημερινής ζωής, ενώ η διαμόρφωση της αγοράς ηλεκτρικής ενέργειας, έγινε αρκετά πρόσφατα. Συγκεκριμένα, η Ευρωπαϊκή Ένωση προχώρησε στην σταδιακή απελευθέρωση της αγοράς ηλεκτρικής ενέργειας, εισάγοντας και εφαρμόζοντας τις 3 ενεργειακές δέσμες, το 1996, 2003 και 2009 αντίστοιχα. Οι δέσμες αυτές εκτός από την απελευθέρωση της αγοράς ηλεκτρικής ενέργειας, αφορούν και την απελευθέρωση της αγοράς φυσικού αερίου.

Βάσει του 3^{ου} ενεργειακού πακέτου, το οποίο έχει ως στόχους την εξασφάλιση του ενεργειακού εφοδιασμού μεταξύ κρατών-μελών και την ενοποίηση των Ευρωπαϊκών αγορών ενέργειας, γίνεται εισαγωγή του Χρηματιστηρίου Ενέργειας. Οι συμμετέχοντες στο Χρηματιστήριο ενέργειας χωρίζονται σε προμηθευτές και παραγωγούς, με τους παραγωγούς να διακρίνονται, βάσει ΑΔΜΗΕ, σε συμβατικούς παραγωγούς και παραγωγούς ενέργειας από ΑΠΕ. Το ποσοστό, δε, της συμμετοχής των ΑΠΕ στο ενεργειακό μείγμα διαρκώς αυξάνεται τα τελευταία χρόνια, κάνοντας εμφανή την πολιτική πρωτοβουλία σε αυτήν την κατεύθυνση.

Ο προμηθευτής αποτελεί τον ενδιάμεσο φορέα μεταξύ Χρηματιστηρίου Ενέργειας και τελικών καταναλωτών, εκπροσωπώντας σωρευτικά χαρτοφυλάκια. Αντίστοιχα για την εκπροσώπηση των ΑΠΕ στο Χρηματιστήριο Ενέργειας, που χαρακτηρίζονται από διεσπαρμένη παραγωγή και στοχαστικότητα σε αυτήν, δημιουργείται η νομική οντότητα του ΦοΣΕ, φορέας μέσω του οποίου εξασφαλίζονται πλεονεκτήματα περιορισμού των αποκλίσεων προβλέψεων-παραγωγής για το σύστημα, ενώ παράλληλα διευκολύνεται η καθολική πρόσβαση των παραγωγών ΑΠΕ (μικρών-μεγάλων) στις διαδικασίες του Χρηματιστηρίου Ενέργειας.

Το μοντέλο αγοράς βάσει του οποίου λειτουργεί η Ευρωπαϊκή Αγορά Ενέργειας και το παραπάνω χρηματιστήριο, ονομάζεται Target Model (Μοντέλο Στόχος). Οι αγορές που υφίστανται, βάσει του Target Model είναι η Αγορά της Επομένης Μέρας, οι Ενδοημερήσιες Δημοπρασίες (συνολικά τρεις), η Ενδοημερήσια Συναλλαγή και η Αγορά Εξισορρόπησης, με βασικές διαφορές μεταξύ τους να είναι οι ώρες διεξαγωγής τους και τα χρονικά διαστήματα αγοραπωλησιών που αφορούν. Τέλος, υπάρχουν και χρηματοπιστωτικά προϊόντα, που διευρύνουν τις δυνατότητες του αγοραστή τους.

Η διαφορετική δομή των αγορών του Χρηματιστηρίου Ενέργειας σε σχέση με το χρηματιστήριο αξιών, οφείλεται στη φύση του αγαθού, καθώς η παραγωγή και η κατανάλωση της ενέργειας πρέπει να γίνεται ταυτόχρονα για την ασφάλεια και τη σταθερότητα του δικτύου ηλεκτρικής ενέργειας. Η ακρίβεια της πρόβλεψης παραγωγής και κατανάλωσης ενέργειας παρουσιάζει αρκετές δυσκολίες και γι' αυτόν το λόγο κρίνεται αναγκαία η ύπαρξη διαχειριστών που θα είναι αρμόδιοι για την ασφάλεια και την εξισορρόπηση των δικτύων. Οι διαχειριστές αυτοί ονομάζονται Διαχειριστές Δικτύων Μεταφοράς (TSO).

Αναλυτικότερα, το δίκτυο ηλεκτρικής ενέργειας διακρίνεται σε δίκτυο μεταφοράς και δίκτυο διανομής, με τους διαχειριστές των δύο να είναι ο TSO και DSO αντίστοιχα.

Στην Ελλάδα, διαχειριστής του δικτύου μεταφοράς είναι ο ΑΔΜΗΕ και διανομής ο ΔΕΔΔΗΕ.

Η είσοδος των ΑΠΕ στην παραγωγή ενέργειας εισήγαγε νέες δυσκολίες ως προς την επίτευξη σταθερότητας στο δίκτυο. Αυτό γιατί σε αντίθεση με τις συμβατικές μονάδες παραγωγής, που αυξομειώνουν την ενέργεια που παράγουν, ανάλογα με τις ανάγκες, οι ΑΠΕ δημιουργούν αβεβαιότητα για την παραγόμενη ενέργεια κάθε στιγμή, καθώς οι καιρικές συνθήκες δεν είναι δεδομένες, ούτε επ' ακριβώς προβλεπόμενες. Η αβεβαιότητα αυτή δημιουργεί, λοιπόν, ανωμαλίες οι οποίες μπορούν να ταξινομηθούν στις εξής: α) Ανωμαλία συχνότητας και τάσης, β) Υπερφόρτωση γραμμών δικτύου και γ) Ανισορροπία ζήτησης και προσφοράς.

Οι τρόποι εξισορρόπησης του δικτύου από την πλευρά του TSO είναι τέσσερις. Πιο συγκεκριμένα είναι α) η αυξομείωση της παραγωγής και της κατανάλωσης, β) η απόρριψη φορτίου, γ) η αποθήκευση ενέργειας και δ) η εισαγωγή και εξαγωγή ενέργειας σε δίκτυα διασυνδεδεμένων χωρών.

Υπάρχουν επίσης, τρόποι διαχείρισης της συμφόρησης του δικτύου, που μπορεί να έχουν στόχο την διαχείριση προβλεπόμενης συμφόρησης ή την πραγματική αποσυμφόρηση του δικτύου ή απλά την κατανομή της ενέργειας σε καταναλωτές. Οι μέθοδοι αυτοί μπορούν να συνοψιστούν στις εξής: α) first in, first serve, β) use it or lose it, γ) pro-rata, δ) market splitting και ε) redispatching.

Ένας από τους βασικούς παράγοντες συγκέντρωσης διοξειδίου του άνθρακα στην ατμόσφαιρα είναι οι μεταφορές. Γι' αυτό η ΕΕ ασκεί πολιτικές προώθησης της ηλεκτροκίνησης με προγράμματα ανάπτυξης δικτύου σταθμών φόρτισης ηλεκτρικών αυτοκινήτων στα κράτη-μέλη αλλά και στόχους όπως ο μηδενισμός των νέων οχημάτων με κινητήρες εσωτερικής καύσης έως το 2035. Στροφή προς την ηλεκτροκίνηση έχουν πραγματοποιήσει και οι μεγάλες αυτοκινητοβιομηχανίες με πολλά από τα νέα μοντέλα να είναι είτε υβριδικά είτε αμιγώς ηλεκτρικά. Τα μελλοντικά σχέδια των αυτοκινητοβιομηχανιών, μάλιστα, σχετικά με την προώθηση και την ανάπτυξη των ηλεκτρικών αυτοκινήτων και οχημάτων, είναι ακόμη πιο φιλόδοξα όπως προκύπτει από τα χρονοδιαγράμματα και τους στόχους που έχουν θέσει ως το 2030 και το 2050.

Από τα παραπάνω γίνεται σαφές ότι η ανάπτυξη της ηλεκτροκίνησης αποτελεί ήδη ένα φλέγον ζήτημα, με προβλήματα, όπως η δυσκολία αντικατάστασης του στόλου αυτοκινήτων εσωτερικής καύσης με ηλεκτροκίνητα, η ανάπτυξη επαρκούς δικτύου φόρτισης ηλεκτρικών αυτοκινήτων και η αύξηση της αυτονομίας τους, να παίρνουν υψηλή θέση στην ιεραρχία αντιμετώπισής τους.

Έτσι, προκύπτει μια νέα ανάγκη, αυτή της διαχείρισης του φορτίου των ηλεκτρικών αυτοκινήτων ώστε να εξασφαλίζεται η ασφάλεια του δικτύου. Πολλές ενέργειες πραγματοποιούνται, λοιπόν, προς αυτή τη κατεύθυνση με ιδιαίτερα αξιοσημείωτο να είναι το πρόγραμμα Decentralized Energy Flexibility Marketplace του SOFIE. Συνοπτικά, το πρόγραμμα αυτό βασίζεται σε blockchain λύσεις και επιτηρείται από τον Διαχειριστή Δικτύου (TSO ή DSO), ο οποίος πραγματοποιεί δημοπρασίες σε χρονικά διαστήματα που γνωρίζει ότι θα δημιουργηθεί αστάθεια στο δίκτυο, λόγω μεγαλύτερης παραγωγής ηλεκτρικής ενέργειας σε σχέση με την κατανάλωση. Οι δημοπρασίες αυτές

απευθύνονται σε διαχειριστές στόλων ηλεκτρικών αυτοκινήτων, οι οποίοι αν κερδίσουν τη δημοπρασία καλούνται να συντονίσουν το στόλο οχημάτων τους, ώστε να φορτίσει στο αντίστοιχο χρονικό διάστημα.

Ένα πρόβλημα της παραπάνω διαδικασίας είναι η ύπαρξη περισσότερων του ενός νικητών σε κάθε δημοπρασία, καθώς σπάνια θα μπορούν να καλυφθούν συνολικά οι ανάγκες με μόνο ένα νικητή, ενώ παράλληλα οι δυνατότητες φόρτισης των φορτιστών είναι αρκετά περιορισμένες ως προς την ισχύ που δύνανται να παρέχουν. Το πρόβλημα λοιπόν που προκύπτει είναι ότι αφενός οι ανάγκες του δικτύου για την αντιμετώπιση της ανισορροπίας είναι μεγαλύτερης τάξης μεγέθους από την ισχύ φόρτισης των φορτιστών, ενώ ταυτόχρονα ο διαχειριστής καλείται να διαλέξει τους φορτιστές και τα αυτοκίνητα που θα χρησιμοποιηθούν επιλέγοντας τις καλύτερες δυνατές προσφορές. Η λύση της παραπάνω εξίσωσης αποτελεί μια αρκετά πολύπλοκη διαδικασία, ενώ το ίδιο το πρόβλημα θα μπορούσε να κατηγοριοποιηθεί βάσει βιβλιογραφίας ως ένα Unit Commitment Problem (UC).

Τα UC στην παραγωγή ηλεκτρικής ενέργειας αποτελούν μια οικογένεια μαθηματικών προβλημάτων βελτιστοποίησης με βασικό στόχο την εξισορρόπηση της ζήτησης ηλεκτρικής ενέργειας με τις παραγωγικές δυνατότητες των γεννητριών και την ταυτόχρονη ελαχιστοποίηση του κόστους παραγωγής ηλεκτρικής ενέργειας. Μια σημαντική δυσκολία στην παραπάνω διαδικασία είναι η συνεχής μεταβολή της ζήτησης ενώ μπορεί εύκολα να γίνει η σύνδεση των στόχων ενός UC με τους στόχους ενός DSO για εξισορρόπηση μέσω ενός στόλου ηλεκτρικών αυτοκινήτων.

Ορισμένες από τις μεταβλητές που χρειάζεται κάποιος να κατανοεί για να εξετάσει τις παραπάνω μεθόδους που χρησιμοποιούν οι διαχειριστές για την εξισορρόπηση του δικτύου και την επίλυση του πολύπλοκου προβλήματος, είναι οι ακόλουθες: a) Binary Commitment Variables (On/off status), b) Power Output Variables (kW, MW), c) Ramp Up and Ramp Down Variables (kW, MW), d) Start-Up and Shut-Down Variables (0/1), e) Reserve Variables (kW, MW), f) Demand Variables (kW, MW), g) Generation Cost Variables.

Όπως αναφέρθηκε και παραπάνω στόχος μιας τέτοιας UC επίλυσης είναι η κάλυψη της μέγιστης δυνατής ζήτησης ενέργειας με το χαμηλότερο δυνατό κόστος παραγωγής. Μια εξίσωση που μπορεί να χρησιμοποιηθεί προς αυτήν την κατεύθυνση και λαμβάνοντας υπόψη κάποιους περιορισμούς, είναι η ακόλουθη:

$$F(P_i^t, U_{i,t}) = \sum_{t=1}^T \sum_{i=1}^N [F_i(P_i^t) + S_{i,t}(1 - U_{i,t-1})] U_{i,t}$$

Στη δική μας περίπτωση υπάρχουν κάποιες αλλαγές όμως σε σχέση με ένα τυπικό UC πρόβλημα. Πιο συγκεκριμένα επειδή δεν υπάρχουν κόστη εκκίνησης για έναν σταθμό φόρτισης, αίρεται ο αντίστοιχος περιορισμός, ενώ θα πρέπει να προστεθεί ένας περιορισμός ακόμη καθώς μας ενδιαφέρει πόσα tokens είναι διαθέσιμα από το Διαχειριστή. Εκτός αυτού, κάθε δημοπρασία αντιμετωπίζεται ως ξεχωριστό time slot, χωρίς να εξετάζεται η συνολική βέλτιστη αλληλουχία όπως σε ένα κλασσικό UC.

Για την επίλυσή του, λοιπόν, μπορεί να γίνει χρήση δυναμικού προγραμματισμού, δηλαδή μια μέθοδος η οποία μπορεί να επιλύσει προβλήματα με αλληλουχία βημάτων, λαμβάνοντας υπόψη κάθε φορά τις αποφάσεις που ελήφθησαν στο προηγούμενο βήμα.

Έτσι, μέσω του δυναμικού προγραμματισμού το βασικό πρόβλημα αναλύεται σε μικρότερα και απλούστερα.

Επίσης, υπάρχει η μέθοδος του Mixed- Integer Linear Programming,, ο στόχος του οποίου είναι να βρεθούν οι βέλτιστες τιμές για τις μεταβλητές απόφασης που ικανοποιούν γραμμικούς περιορισμούς, ενώ, ταυτόχρονα, ελαχιστοποιούν ή μεγιστοποιούν μια αντικειμενική συνάρτηση. Η διαφοροποίηση του από το δυναμικό προγραμματισμό στην προσέγγιση του UC έγκειται στο τρόπο επίλυσης του. Το MILP χρησιμοποιεί ταυτόχρονα διακριτές και συνεχείς μεταβλητές για τη βελτιστοποίηση του προβλήματος, ενώ στο δυναμικό προγραμματισμό στόχος είναι η διάσπαση του προβλήματος σε πολλά μικρότερα επικαλυπτόμενα προβλήματα, η εύρεση της λύσης στο κάθε υποπρόβλημα, η αποθήκευση αυτής της λύσης και, εν τέλει, η εύρεση της καλύτερης λύσης.

Ένας άλλος αλγόριθμος που χρησιμοποιείται για την επίλυση του προβλήματος είναι ο Particle Swarm Optimization (PSO). Ο PSO είναι ένας μετα-ευρεστικής βελτιστοποίησης αλγόριθμος, που λειτουργεί έχοντας μια ομάδα σωματιδίων (ή λύσεων) μέσα σε ένα χώρο αναζήτησης. Κάθε σωματίδιο έχει μια «fitness» τιμή, η οποία είναι ένας δείκτης του πόσο καλή είναι η λύση. Τα σωματίδια κινούνται προς τις καλύτερες λύσεις που έχουν βρει μέχρι τώρα, και ταυτόχρονα κινούνται προς την καλύτερη λύση που έχει βρει όλο το σμήνος μέχρι τώρα. Αυτή η διαδικασία επαναλαμβάνεται μέχρι το σμήνος να συγκλίνει σε μια καλή λύση.

Τέλος, άλλη μια μέθοδος επίλυσης είναι η Lagrangian Relaxation, που μοιάζει με το δυναμικό προγραμματισμό ως προς την αναζήτηση της βέλτιστης λύσης, η διαδικασία όμως εδώ είναι διαφορετική. Με αυτή την μέθοδο, αναζητείται τρόπος άρσης ή ελαχιστοποίησης της επίδρασης των αρχικών περιορισμών, έτσι ώστε να διαμορφωθεί ένα αρκετά ευκολότερο αρχικό πρόβλημα, ενώ στη συνέχεια αυτό το πρόβλημα αναλύεται σε μικρότερα προβλήματα για την ευκολότερη επίλυσή του.

Τα δεδομένα που λήφθηκαν υπόψη για την εκτέλεση των παραπάνω αλγοριθμικών μεθόδων και την επίλυση του προβλήματος, αφορούσαν: α) τους σταθμούς φόρτισης των ηλεκτρικών αυτοκινήτων, β) τα ηλεκτρικά αυτοκίνητα των στόλων, γ) τις προβλέψεις που κάνει ο διαχειριστής για τις ανάγκες δ) τις δημοπρασίες και ε) τις προσφορές των διαχειριστών στόλων ηλεκτρικών οχημάτων και περιγράφονται αναλυτικά στο κεφάλαιο 4.

Με χρήση γλώσσας προγραμματισμού Python και βάσης δεδομένων PostgreSQL, η εκτέλεση των αλγορίθμων οδήγησε στα εξής αποτελέσματα:

1. Οι αλγόριθμοι knapsack με βάρη και PSO βρήκαν στο 100% των δημοπρασιών λύσεις που να τηρούν τον περιορισμό που έχει τεθεί, δηλαδή το άθροισμα της ενέργειας της λύσης να μην υπερβαίνει τη ζητούμενη ενέργεια. Ο αλγόριθμος με τις λιγότερες λύσεις ήταν ο LR
2. Οι δύο αλγόριθμοι, που βρήκαν τις λιγότερες αποδεκτές λύσεις (DP και LR), έχουν το μικρότερο ποσοστό υπολειπόμενης ενέργειας. Αυτό υποδεικνύει ότι, όταν ένας από τους δύο αλγόριθμους καταφέρει να βρει λύση, πιθανότατα θα δώσει και καλύτερη λύση από τους υπόλοιπους

3. Ο PSO αλγόριθμος βρίσκει καλύτερες λύσεις σε δημοπρασίες που έχουν μεγαλύτερη δημοπρατούμενη ενέργεια και περισσότερες προσφορές, ενώ βρίσκει χειρότερες λύσεις σε δημοπρασίες που η δημοπρατούμενη ενέργεια είναι μικρή και ο ανταγωνισμός φτωχός.
4. Ο DP έχει την καλύτερη απόδοση από όλους τους αλγόριθμους, ενώ ο LR έχει τις περισσότερες δημοπρασίες στις οποίες οι ενέργειες υπερκαλύπτονται.
5. Ο αλγόριθμος DP έχει με διαφορά την ακριβότερη λύση. Συγκεκριμένα, θα χρειαζόμασταν 109.3% tokens επιπλέον από τα ήδη διαθέσιμα για να πληρώσουμε τις αποδεκτές λύσεις.
6. Ο αλγόριθμος LR αποτελεί την πιο οικονομικά συμφέρουσα λύση
7. αν ο αλγόριθμος LR δώσει λύση εντός των περιορισμών που έχουν τεθεί, θα δώσει πιθανότατα και την καλύτερη λύση

Κλείνοντας, τα γενικά συμπεράσματα που προκύπτουν από την εκτέλεση των αλγορίθμων και τα αποτελέσματά τους είναι ότι αφενός α) με τις κατάλληλες προσαρμογές στην αντικειμενική συνάρτηση του UC προβλήματος, μπορούν να χρησιμοποιηθούν οι ίδιες τεχνικές και αλγόριθμοι για την επίλυση μιας δημοπρασίας με υψηλή αποτελεσματικότητα και αφετέρου β) ότι για την καλύτερη επίλυση της δημοπρασίας, μπορεί να γίνει χρήση διαφορετικών αλγορίθμων, καθώς η αποτελεσματικότητα του καθενός εξαρτάται από τις αντικειμενικές παραμέτρους της δημοπρασίας.

Summary

Electricity is one of the most important commodities in modern life, and the restructuring of the electricity market has occurred relatively recently. Specifically, the European Union initiated the gradual liberalization of the electricity market by introducing and implementing the three energy packages in 1996, 2003, and 2009, respectively. These packages not only pertain to the liberalization of the electricity market but also involve the liberalization of the natural gas market.

Under the third energy package, which aims to ensure energy supply between member states and unify European energy markets, the Energy Exchange is introduced. Participants in the Energy Exchange are divided into suppliers and producers, with producers further categorized as conventional producers and producers of energy from renewable sources (RES). The percentage of RES participation in the energy mix has been steadily increasing in recent years, highlighting the policy initiative in this direction.

The supplier serves as the intermediary between the Energy Exchange and end consumers, representing cumulative portfolios. Similarly, for the representation of Renewable Energy Sources (RES) in the Energy Exchange, characterized by distributed production and stochasticity, the legal entity of FoSE (Facility of Stochastic Energy) is established. Through FoSE, advantages in prediction-production deviation reduction for the system are secured, while facilitating universal access for RES producers (both small and large) to the processes of the Energy Exchange.

The market model under which the European Energy Market operates, including the aforementioned Energy Exchange, is called the Target Model. The markets that exist under the Target Model are the Day-Ahead Market, Intraday Auctions (three in total), the Intraday Market, and the Balancing Market, with significant differences in their trading hours and the time intervals they involve for buying and selling. Finally, there are financial products that expand the options available to buyers.

The different structure of the Energy Exchange markets compared to the stock market is due to the nature of the commodity, as the production and consumption of energy must occur simultaneously for the security and stability of the electrical grid. The accuracy of energy production and consumption forecasts presents several challenges, which is why the presence of network operators responsible for the security and balance of the grids is deemed necessary. These operators are called Transmission System Operators (TSOs).

In more detail, the electrical energy network is divided into the transmission network and the distribution network, with the operators being TSO and DSO, respectively. In Greece, the TSO is ADMIE for the transmission network, and the DSO is HEDNO (Hellenic Electricity Distribution Network Operator) for the distribution network.

The integration of Renewable Energy Sources (RES) into energy production introduced new challenges regarding network stability. This is because, unlike conventional production units that can adjust their energy output according to demand, RES create

uncertainty in energy generation at any given moment due to unpredictable weather conditions. This uncertainty leads to anomalies that can be classified into the following categories: a) Frequency and voltage anomalies, b) Network line overloads, and c) Demand and supply imbalance.

Transmission System Operators (TSOs) have four methods for network balancing. Specifically, these methods include a) adjusting production and consumption, b) shedding load, c) energy storage, and d) energy import/export through interconnected networks between countries.

There are also methods for managing network congestion, which may aim to predict and alleviate congestion or simply distribute energy to consumers. These methods can be summarized as follows: a) first in, first serve, b) use it or lose it, c) pro-rata, d) market splitting, and e) redispatching.

One of the key contributors to carbon dioxide emissions in the atmosphere is transportation. Therefore, the European Union (EU) promotes electric mobility policies, including the development of charging station networks in member states and targets such as phasing out new vehicles with internal combustion engines by 2035. Major automakers have also shifted towards electrification, with many new models being either hybrid or fully electric. Future plans of automakers regarding the promotion and development of electric vehicles are even more ambitious, as seen in their schedules and goals set for 2030 and 2050.

From the above, it is clear that the development of electric mobility is already a pressing issue, with challenges such as the difficulty of replacing the internal combustion engine vehicle fleet with electric ones, expanding an adequate electric vehicle charging infrastructure, and increasing vehicle autonomy ranking high in the hierarchy of challenges.

As a result, there is a new need for managing the load of electric vehicles to ensure network safety. Many actions are being taken in this direction, with notable mention of the SOFIE program, the Decentralized Energy Flexibility Marketplace. In summary, this program is based on blockchain solutions and is overseen by the Network Operator (TSO or DSO), who conducts auctions during time intervals where network instability is expected due to excess electricity production compared to consumption. These auctions target fleet managers of electric vehicles, who, if they win the auction, are responsible for coordinating their vehicle fleet to charge during the specified time frame.

The problem with the above process is the existence of more than one winner in each auction, as rarely can the network's needs be fully met with only one winner, while at the same time, the charging capabilities of the chargers are quite limited in terms of the power they can provide. The problem that arises is that, on the one hand, the network's needs to address the imbalance are orders of magnitude higher than the charging power of the chargers, while at the same time, the administrator is called upon to select the chargers and cars to be used by choosing the best possible offers. The solution to the above equation is a rather complex process, and the problem itself could be categorized based on the literature as a Unit Commitment Problem (UC).

UCs in electricity production are a family of mathematical optimization problems with the primary goal of balancing the demand for electricity with the generation capabilities of the generators and simultaneously minimizing the cost of electricity production. A significant challenge in the above process is the continuous variation in demand, while it can be easily connected the objectives of a UC with the objectives of a DSO for balance through a fleet of electric vehicles.

Some of the variables one needs to understand to examine the above methods used by administrators to balance the network and solve the complex problem include: a) Binary Commitment Variables (On/off status), b) Power Output Variables (kW, MW), c) Ramp Up and Ramp Down Variables (kW, MW), d) Start-Up and Shut-Down Variables (0/1), e) Reserve Variables (kW, MW), f) Demand Variables (kW, MW), g) Generation Cost Variables.

As mentioned above, the goal of such a UC solution is to cover the maximum demand for energy with the lowest possible production cost. An equation that can be used in this direction, taking into account certain constraints, is as follows:

$$F(P_i^t, U_{i,t}) = \sum_{t=1}^T \sum_{i=1}^N [F_i(P_i^t) + S_{i,t}(1 - U_{i,t-1})] U_{i,t}$$

In our specific case, there are some differences compared to a typical Unit Commitment (UC) problem. More specifically, because there are no startup costs for a charging station, that constraint is removed, and an additional constraint should be added as we are interested in how many tokens are available from the Administrator. Furthermore, each auction is treated as a separate time slot, without considering the overall optimal sequence as in a classic UC problem.

To solve this, dynamic programming can be used, which is a method that can solve problems step by step, taking into account the decisions made in the previous step.

In our case, there are some differences compared to a typical Unit Commitment (UC) problem. Specifically, because there are no startup costs for a charging station, that constraint is removed, and an additional constraint should be added as we are interested in how many tokens are available from the Administrator. Furthermore, each auction is treated as a separate time slot, without considering the overall optimal sequence as in a classic UC problem.

To solve it, dynamic programming can be used, which is a method that can solve problems step by step, taking into account the decisions made in the previous step.

Another method is Mixed-Integer Linear Programming (MILP), the goal of which is to find the optimal values for decision variables that satisfy linear constraints while simultaneously minimizing or maximizing an objective function. The difference from dynamic programming in the UC approach lies in the way it is solved. MILP simultaneously uses discrete and continuous variables to optimize the problem, while dynamic programming aims to break down the problem into many smaller overlapping subproblems, find the solution to each subproblem, store that solution, and ultimately find the best solution.

Another algorithm used to solve the problem is Particle Swarm Optimization (PSO). PSO is a metaheuristic optimization algorithm that operates with a group of particles (or solutions) within a search space. Each particle has a "fitness" value, which is an indicator of how good the solution is. Particles move towards the best solutions they have found so far and, at the same time, move towards the best solution the entire swarm has found. This process is repeated until the swarm converges to a good solution.

Finally, another solution method is Lagrangian Relaxation, which is similar to dynamic programming in terms of finding the optimal solution but has a different approach. With this method, the goal is to find a way to relax or minimize the impact of the initial constraints, making the initial problem significantly easier to solve. Then, this problem is further broken down into smaller subproblems for easier resolution.

The data considered for executing these algorithmic methods and solving the problem included: a) electric vehicle charging stations, b) fleet electric cars, c) predictions made by the administrator for needs, d) auctions, and e) bids from the administrator. These are described in detail in Chapter 4.

Using the Python programming language and a PostgreSQL database, the execution of the algorithms led to the following results:

1. The knapsack algorithms with weights and PSO found solutions that meet the specified constraint, i.e., the sum of the solution's energy does not exceed the required energy, in 100% of auctions. The algorithm with the fewest solutions was LR.
2. The two algorithms that found the fewest acceptable solutions (DP and PSO) have the lowest remaining energy percentage. This suggests that when one of the two algorithms finds a solution, it is likely to provide a better solution than the others.
3. The PSO algorithm finds better solutions in auctions with higher auctioned energy and more bids, while it finds worse solutions in auctions with low auctioned energy and poor competition.
4. DP has the best performance among all algorithms, while LR has the most auctions where the energy exceeds the requirements.
5. DP has by far the most expensive solution. Specifically, it would require 109.3% additional tokens beyond those already available to pay for the acceptable solutions.
6. The LR algorithm represents the most economically favorable solution.
7. If the LR algorithm provides a solution within the specified constraints, it will likely provide the best solution.

In conclusion, the general conclusions that emerge from the execution of the algorithms and their results are that, on the one hand, a) with appropriate adjustments to the objective function of the UC problem, the same techniques and algorithms can be used to efficiently solve an auction, and on the other hand, b) for the best auction solution, different algorithms can be used, as the effectiveness of each depends on the objective parameters of the auction."

ΚΕΦΑΛΑΙΟ 1
ΕΙΣΑΓΩΓΗ

1. Εισαγωγή

1.1 Στόχος της διπλωματικής εργασίας

Ο ενεργειακός τομέας είναι στο επίκεντρο μιας σημαντικής αλλαγής, που προκαλείται εξαιτίας της ταχύρρυθμης τεχνολογικής προόδου και της περιβαλλοντικής βιωσιμότητας. Τα ηλεκτρικά οχήματα δεν προκαλούν μεγάλες αλλαγές μόνο στα μέσα μεταφοράς και τα οδικά δίκτυα, αλλά και στο χώρο της ηλεκτρικής ενέργειας. Ενώ αυξάνεται η δημοτικότητα και πολλαπλασιάζεται ο αριθμός των ηλεκτρικών οχημάτων, αυξάνεται και η ζήτηση για αποδοτικές υποδομές φόρτισης και μηχανισμούς για την ενσωμάτωσή τους στο ευρύτερο ενεργειακό δίκτυο χωρίς να προκαλούνται ανισορροπίες.

Κεντρικό στοιχείο αυτής της πρόκλησης είναι οι Διαχειριστές Συστημάτων Μεταφοράς (ΔΣΜ ή TSO), υπεύθυνοι για τη διασφάλιση της σταθερότητας και της αξιοπιστίας των δικτύων ηλεκτρικής ενέργειας. Μια καινοτόμος προσέγγιση σε αυτήν την πρόκληση έγκειται στην ιδέα της δημοπρασίας της ενεργειακής ζήτησης του δικτύου που παρουσιάζεται στο έργο H2020 SOFIE. Μέσω αυτών των δημοπρασιών, οι διαχειριστές δικτύων δίνουν την ευκαιρία σε στόλους ηλεκτρικών οχημάτων να συμμετέχουν ενεργά στις αγορές ενέργειας, όχι απλώς ως καταναλωτές, αλλά ως βασικοί παίκτες, που μπορούν να υποβάλουν προσφορά για να βοηθήσουν στην εξισορρόπηση της ζήτησης του δικτύου.

Αυτή η διπλωματική επιχειρεί να παρουσιάσει ενεργειακά θέματα, όπως οι αγορές ηλεκτρικής ενέργειας, το πρόβλημα της δέσμευσης μονάδων παραγωγής ενέργειας (Unit Commitment, UC) για την εξισορρόπηση του δικτύου και τη τεχνική ανάλυση της φόρτισης των ηλεκτρικών οχημάτων. Αφού αναλυθούν τα προαναφερθέντα, καθώς και ο τρόπος λειτουργίας των δημοπρασιών που προτείνει το έργο H2020 SOFIE, εξετάζεται πως, και αν μπορούν, να εφαρμοστούν οι αλγόριθμοι και οι τεχνικές που ήδη χρησιμοποιούνται για την επίλυση του προβλήματος UC για την εύρεση του βέλτιστου συνδυασμού νικητών μιας δημοπρασίας, με στόχο την εξισορρόπηση του δικτύου. Μετά και την εφαρμογή διάφορων αλγορίθμων, παρουσιάζονται και συγκρίνονται τα αποτελέσματά τους.

1.2 Δομή της εργασίας

Στο ΚΕΦΑΛΑΙΟ 2, αναλύεται ο τρόπος λειτουργίας του χρηματιστηρίου ενέργειας, καθώς και οι αγορές του, δηλαδή η Αγορά Επόμενης Μέρας, οι Ενδοημερήσιες

δημοπρασίες, οι Ενδοημερήσιες συνεχείς συναλλαγές, η Αγορά εξισορρόπησης και η Ενεργειακή χρηματοπιστωτική αγορά. Επιπλέον, εξηγούμε το πως μεταφέρεται η ηλεκτρική ενέργεια μέσω ενός δικτύου και το ρόλο και τις ευθύνες του διαχειριστή του δικτύου, τις αιτίες δημιουργίας ανισορροπιών στο δίκτυο και τον τρόπο εξισορρόπησης του, καθώς και αποσυμφόρησής του.

Στο ΚΕΦΑΛΑΙΟ 3, γίνεται μια εισαγωγή στον κλάδο της ηλεκτροκίνησης και τους μελλοντικούς της στόχους και παρουσιάζεται το πρόβλημα της δέσμευσης μονάδων (UC). Τέλος, γίνεται μια μαθηματική αποτύπωση του προβλήματος και αναλύονται μερικοί αλγόριθμοι που προσεγγίζουν τη λύση του συγκεκριμένου προβλήματος.

Στο ΚΕΦΑΛΑΙΟ 4, παρουσιάζονται τα δεδομένα που θα χρησιμοποιηθούν στη διπλωματική. Συγκεκριμένα, με αυτά τα δεδομένα θα χρησιμοποιήσουμε τους αλγόριθμους του προηγούμενου κεφαλαίου, κάνοντας πάντα τις απαραίτητες μετατροπές.

Στο ΚΕΦΑΛΑΙΟ 5, παρουσιάζεται ο τρόπος εκτέλεσης των αλγορίθμων.

Στο ΚΕΦΑΛΑΙΟ 6, αναλύονται τα αποτελέσματα των αλγορίθμων και συγκρίνουμε την απόδοσή τους, καταλήγοντας σε συμπεράσματα για τον, αν μπορούν οι ίδιοι αλγόριθμοι που «λύνουν» το πρόβλημα UC, να βρουν και τους βέλτιστους «νικητές» μιας δημοπρασίας για την εξισορρόπηση του δικτύου.

Στο ΚΕΦΑΛΑΙΟ 7, παρουσιάζονται οι προοπτικές που προκύπτουν από τη διπλωματική και προτείνονται τρόποι εξέλιξης των δημοπρασιών που παρουσιάζονται από το έργο H2020 SOFIE.

ΚΕΦΑΛΑΙΟ 2
ΑΓΟΡΑ ΕΝΕΡΓΕΙΑΣ

2. Αγορά της Ενέργειας

Η ηλεκτρική ενέργεια αποτελεί ένα από τα βασικά αγαθά της ανθρώπινης ζωής. Ωστόσο, η λειτουργία και η αγορά του χρηματιστήριου ενέργειας είναι νέα σε σχέση με την αγορά άλλων υλικών αγαθών, όπως αυτή του αργού πετρελαίου. Παρόλο που παρουσιάζονται ομοιότητες ως προς τη χρήση τους, υπάρχουν αρκετοί παράγοντες που το διαφοροποιούν.

Η Ευρωπαϊκή Ένωση ξεκίνησε την προσπάθεια για την απελευθέρωση της αγοράς ηλεκτρικής ενέργειας το 1996 με το πρώτο ενεργειακό πακέτο. Το πακέτο αυτό περιλάμβανε την πρώτη οδηγία σχετικά με την εσωτερική αγορά ηλεκτρικής ενέργειας. Το 1998, το ίδιο πακέτο τροποποιήθηκε για να συμπεριλάβει την πρώτη οδηγία για την απελευθέρωση της αγοράς φυσικού αερίου. Το 2003, το δεύτερο ενεργειακό πακέτο τροποποίησε και αντικατέστησε τις δύο οδηγίες με δύο οδηγίες και δύο κανονισμούς. Το 2009, το τρίτο ενεργειακό πακέτο αντικατέστησε τα προηγούμενα πακέτα με δύο οδηγίες και τρεις κανονισμούς οι οποίοι ισχύουν μέχρι και σήμερα.

Τα ενεργειακά πακέτα αυτά της Ευρωπαϊκής Ένωσης είχαν ως στόχο την δημιουργία μιας ενιαίας αγοράς ηλεκτρικής ενέργειας και φυσικού αερίου που επιτρέπει στους καταναλωτές να έχουν μεγαλύτερη επιλογή προμηθευτών και να πληρώνουν χαμηλότερες τιμές. Τα ενεργειακά πακέτα βοηθούν επίσης στην αύξηση της ανταγωνιστικότητας της αγοράς και στη βελτίωση της ασφάλειας του εφοδιασμού.

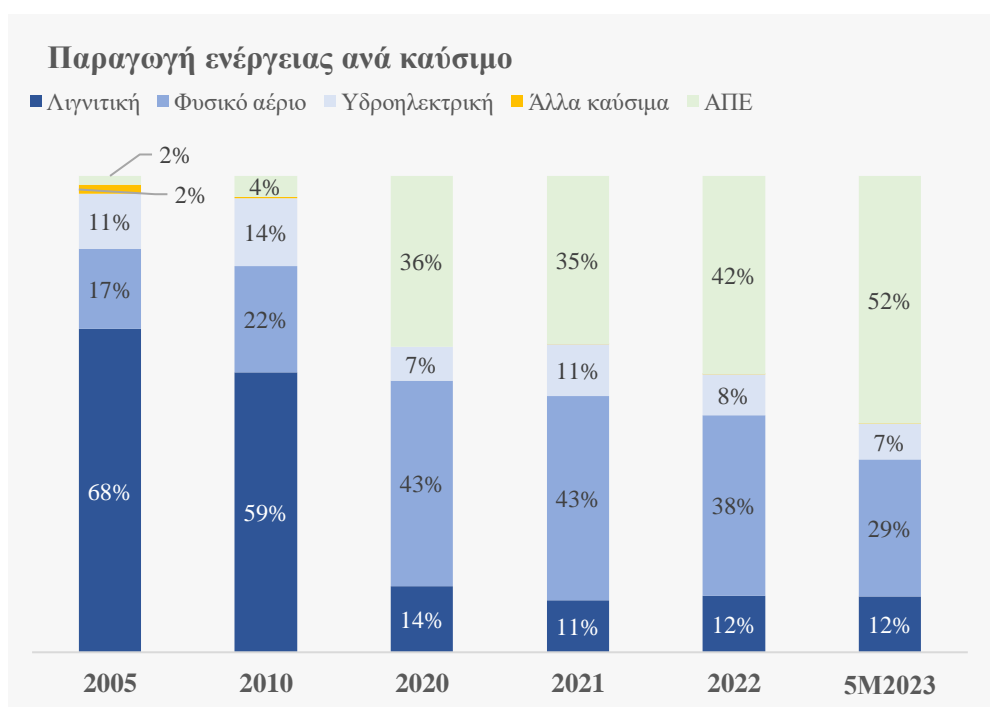
Οι συμμετέχοντες στο χρηματιστήριο ενέργειας χωρίζονται σε 2 βασικές κατηγορίες: τους παραγωγούς ενέργειας, οι οποίοι πωλούν ενέργεια (φυσικά ή νομικά πρόσωπα που παράγουν ενέργεια με θερμοηλεκτρικά εργοστάσια παραγωγής, ανανεώσιμες πηγές ενέργειας, θερμοϋδροηλεκτρικά κα), και τους προμηθευτές ενέργειας, οι οποίοι αγοράζουν ενέργεια, (νομικά πρόσωπα που έχουν άδεια προμήθειας ενέργειας). Στη συνέχεια οι καταναλωτές ενέργειας, συνήθως αποκαλούμενοι ως τελικοί καταναλωτές, όπως τα εργοστάσια και τα νοικοκυριά, συνάπτουν σύμβαση με έναν προμηθευτή ενέργειας, ώστε να καταναλώνουν την ενέργεια που χρειάζονται.

Το μείγμα της ενεργειακής παραγωγής χωρίζεται, σύμφωνα με τη κατηγοριοποίηση του ΑΔΜΗΕ, σε συμβατική παραγωγή και παραγωγή ΑΠΕ. Η συμβατική παραγωγή απαρτίζεται από παραγωγή ενέργειας από λιγνίτη, φυσικό αέριο, μεγάλα

υδροηλεκτρικά, και άλλα καύσιμα. Η παραγωγή των ΑΠΕ προέρχεται από αιολικά και φωτοβολταϊκά πάρκα, μικρές υδροηλεκτρικές μονάδες και ΣΗΘΥΑ (Συμπααραγωγή Ηλεκτρισμού-Θερμότητας Υψηλής Απόδοσης).

Το ποσοστό συμβατικής παραγωγής ως προς το σύνολο βαίνει μειούμενο τις τελευταίες δύο δεκαετίες, από το 98% της συνολικής παραγωγής το 2005 στο 48% το πρώτο 5μηνο του 2023. Αναλύοντας τις επιμέρους κατηγορίες είναι ξεκάθαρη η στρατηγική μείωσης της εξάρτησης σε ορυκτά καύσιμα και συγκεκριμένα στο λιγνίτη, που από το 68% της συνολικής παραγωγής έχει πέσει στο 12%. Την θέση του λιγνίτη ως το κύριο καύσιμο συμβατικής παραγωγής έχει πάρει πλέον το φυσικό αέριο το οποίο απαρτίζει το 29% της συνολικής παραγωγής.

Τα ΑΠΕ για πρώτη φορά το 2023 απαρτίζουν περισσότερο από 50% της συνολικής παραγωγής στο σύστημα και αγγίζουν το 43.5% της συνολικής ζήτησης. Συγκριτικά, ο στόχος της Ευρωπαϊκής Ένωσης είναι στο 42.5% έως το 2030.



Ο ρόλος των προμηθευτών είναι να δρουν ως τον ενδιάμεσο ανάμεσα στο χρηματιστήριο ενέργειας και τον τελικό καταναλωτή. Αναλαμβάνουν τον ρόλο του αγοραστή στο χρηματιστήριο σε επίπεδο χονδρικής και μετέπειτα τον πωλητή στην λιανική αγορά. Από όταν απελευθερώθηκε η αγορά προμήθειας και η ΔΕΗ έχασε τον μονοπωλιακό της ρόλο έχουν συσταθεί δεκάδες εταιρίες προμήθειας. Παρόλα αυτά, παραμένει ο κυρίαρχος προμηθευτής σε όλες τις τάσεις, με συνολικό μερίδιο αγοράς

60% για το 5μηνο του 2023. Διατηρεί επίσης υψηλό μερίδιο 72% στην Υψηλή τάση, η οποία απευθύνεται αποκλειστικά σε βιομηχανικούς καταναλωτές.

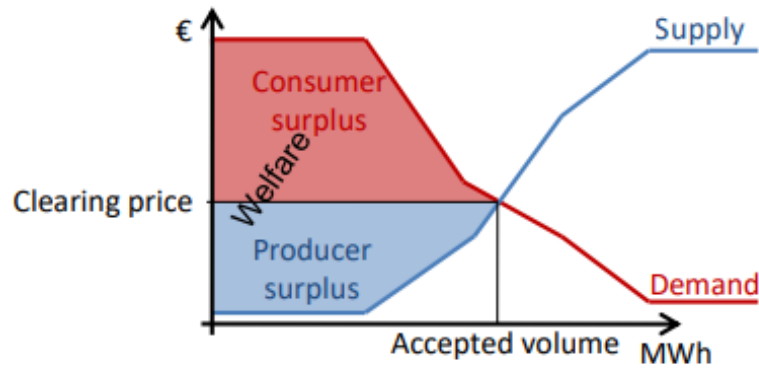
Οι Φορείς Σωρευτικής Εκπροσώπησης (ΦοΣΕ) είναι εταιρίες που αναλαμβάνουν τον ρόλο του ενδιάμεσου ανάμεσα σε παραγωγούς και προμηθευτές ενέργειας στο χρηματιστήριο ενέργειας. Ο ρόλος αυτός είναι καίριας σημασίας καθώς κάνει ευκολότερη την πρόσβαση στις αγορές για τους μικρούς παραγωγούς και αυξάνει σημαντικά την ικανότητα του συστήματος να ανταποκριθεί στις αυξομειώσεις της ζήτησης. Ένας παραγωγός με την εκπροσώπηση του από ένα ΦοΣΕ απαλλάσσεται από πολύπλοκες διαδικασίες όπως η πρόβλεψη της παραγωγής και η κατάθεση προσφορών. Επιπλέον, αυτές οι διαδικασίες είναι σημαντικά πιο αποτελεσματικές όταν εφαρμόζονται στα χαρτοφυλάκια που εκπροσωπούν οι ΦοΣΕ. Οι ΦοΣΕ εκπροσωπούν μεγάλο όγκο ΑΠΕ σε όλη την επικράτεια και έτσι καθίσταται πολύ ευκολότερη η πρόβλεψη της παραγωγής και η ανταπόκριση σε τυχόν αποκλίσεις.

Στο χρηματιστήριο ενέργειας, όλες οι αγορές που το απαρτίζουν, αναφέρονται σε μελλοντικό χρόνο. Πιο συγκεκριμένα, το target model της Ευρώπης για το χρηματιστήριο ενέργειας αποτελείται από τις αγορές: *Αγορά Επόμενης Ημέρας*, *Ενδοημερήσιες δημοπρασίες*, *Ενδοημερήσια συνεχής αγορά*, *Αγορά Εξισορρόπησης*.

I. Αγορά επόμενη ημέρας

Η αγορά επόμενης ημέρας αφορά συναλλαγές αγοράς και πώλησης ηλεκτρικής ενέργειας για φυσική παράδοση την επόμενη ημέρα. Συγκεκριμένα, για την ημέρα (D) οι συμμετέχοντες της αγοράς υποβάλλουν προσφορές κατανάλωσης ή παραγωγής ηλεκτρικής ενέργειας την ημέρα (D-1). Υπάρχουν διαθέσιμα 24 χρονικά slots (ένα για κάθε ώρα), για τα οποία μπορούν να γίνουν προσφορές. Σημειώνεται ότι ένας συμμετέχων, του οποίου το χαρτοφυλάκιο αποτελείται μόνο από ενεργειακά assets παραγωγής ενέργειας δεν μπορεί να έχει αρνητικό Net Position, και αντίστροφα. Με τον όρο Net Position εννοούμε τη θέση του συμμετέχοντα που προκύπτει από το άθροισμα όλων των προσφορών του στις αγορές ενέργειας. Η αγορά αυτή ανοίγει 2 ημέρες πριν τη ημέρα της φυσικής παράδοσης (D-2) και κλείνει στις 12:00 CET την ημέρα D-1 (ώρα Ελλάδος 13:00). Κάθε χρονικό slot αποτελεί μια δημοπρασία, για την οποία ο αλγόριθμος EUPHEMIA αποφασίζει την τιμή της και τις αποδεκτές προσφορές. Πρόκειται για έναν αλγόριθμο που βασίζεται το Ευρωπαϊκό target model, που στοχεύει στην

εύρεση ανταγωνιστικής τιμής, στην αύξηση της συνολικής ευημερίας και στην αποτελεσματική κατανομή της χωρητικότητας. Τα αποτελέσματα του αλγορίθμου κοινοποιούνται 1 ώρα μετά το κλείσιμο της αγοράς.



II. Ενδοημερήσια δημοπρασία

Η λειτουργία της Ενδοημερήσιας δημοπρασίας είναι πανομοιότυπη με αυτή της Αγοράς επόμενης ημέρας. Πρόκειται πάλι για δημοπρασίες, οι οποίες ανοίγουν πάντα μετά το κλείσιμο της αγοράς επόμενης ημέρας και βρίσκονται χρονικά πιο κοντά στη φυσική παράδοση. Η κάθε χώρα ανάλογα τις ανάγκες της έχει διαφορετικό αριθμό τέτοιων δημοπρασιών. Στην Ελλάδα υπάρχουν 3 τέτοιες δημοπρασίες για τις οποίες ισχύουν τα παρακάτω:

- Συμπληρωματική Περιφερειακή Ενδοημερήσια Δημοπρασία 1(CRIDA 1): ανοίγει στις 14:30 και κλείνει στις 16:00 την ημέρα D-1 και αφορά τις 24 ώρες της ημέρας D
- Συμπληρωματική Περιφερειακή Ενδοημερήσια Δημοπρασία 2(CRIDA 2): ανοίγει στις 16:30 και κλείνει στις 23:00 την ημέρα D-1 και αφορά τις 24 ώρες της ημέρας D
- Συμπληρωματική Περιφερειακή Ενδοημερήσια Δημοπρασία 3(CRIDA 3): ανοίγει στις 23:30 την ημέρα D και κλείνει στις 11:00 την ημέρα D και αφορά τις 12 τελευταίες ώρες της ημέρας D

Ο λόγος που δημιουργήθηκαν είναι για να δώσουν τη δυνατότητα στους συμμετέχοντες να διορθώσουν τις προσφορές τους. Επειδή στην αγορά επόμενης ημέρας οι προσφορές αναφέρονται σε μελλοντικό χρόνο, μέχρι τη στιγμή της

φυσικής παράδοσης, μπορούν να προκληθούν αλλαγές στο scheduling της παραγωγής ή της κατανάλωσης, όπως η αύξηση της ζήτησης ενός προϊόντος που απαιτεί ενέργεια για την παραγωγή του ή βλάβη σε κάποιο ενεργειακό μέσο παραγωγής ενέργειας. Έτσι, σε αυτές τις αγορές, οι συμμετέχοντες μπορούν να «διορθώσουν» τη θέση τους κάνοντας τις αντίστοιχες προσφορές. Τα αποτελέσματα αυτών των δημοπρασιών προκύπτουν από τον αλγόριθμο EUPHEMIA.

III. Ενδοημερήσια Συνεχής αγορά

Η Ενδοημερήσια συνεχής αγορά δημιουργήθηκε για το λόγο που δημιουργήθηκαν και οι Ενδοημερήσιες δημοπρασίες, ωστόσο η λειτουργία τους διαφέρει αρκετά. Η αγορά αυτή κλείνει μία ώρα πριν από τη φυσική παράδοση του χρονικού slot για το οποίο υποβάλλονται προσφορές. Τα χρονικά slots σε αυτές διαφέρουν ανάλογα με τη χώρα, με αποτέλεσμα, εκτός από τα ωριαία, να υπάρχουν και 30λεπτα και 15λεπτα χρονικά slots. Ακόμη, δεν είναι μια αγορά δημοπρασιών, αλλά μια συνεχής αγορά, σαν αυτή του συναλλάγματος. Οι συμμετέχοντες μπορούν να αγοράζουν και να πωλούν ενέργεια σε πραγματικό χρόνο. Αποτελεί πολύ σημαντική για την ασφάλεια και την αξιοπιστία του ενεργειακού συστήματος, καθώς επιτρέπει την ανταπόκριση σε αλλαγές στη ζήτηση και στη προσφορά κατά τη διάρκεια της ημέρας, μέχρι και 1 ώρα πριν τη φυσική παράδοση.

IV. Αγορά εξισορρόπησης

Η αγορά εξισορρόπησης απαρτίζεται από την αγορά ισχύος εξισορρόπησης, την αγορά ενέργειας εξισορρόπησης και την εκκαθάριση αποκλίσεων. Τις συγκεκριμένες αγορές τις διαχειρίζεται ο Ανεξάρτητος Διαχειριστής Μεταφοράς Ηλεκτρικής Ενέργειας (ΑΔΜΗΕ), σε αντίθεση με τις προηγούμενες ο οποίες ήταν υπό την διαχείριση του Ελληνικού Χρηματιστηρίου Ενέργειας. Ο ρόλος των τριών αυτών αγορών είναι να ανταποκριθούν στις αποκλίσεις που παρουσιάζονται στο σύστημα και να διασφαλίσουν την σταθερότητα του. Με τον όρο «αστάθεια» αναφερόμαστε ουσιαστικά σε ανισορροπία κατανάλωσης και παραγωγής στο σύστημα.

Οι συμβατικές μονάδες παραγωγής συμμετέχουν υποχρεωτικά και στις τρεις αγορές εξισορρόπησης ενώ οι υπόλοιπες μονάδες συμμετέχουν υποχρεωτικά μόνο στην εκκαθάριση αποκλίσεων. Οι συμμετέχοντες υποβάλλουν προσφορές από τις

14:00 έως και τις 16:00 της προηγούμενης μέρας (D-1). Μετέπειτα, ο διαχειριστής εκτελεί την διαδικασία βελτιστοποίησης του συστήματος, δηλαδή ταύτισης της ζήτησης με την προσφορά, λαμβάνοντας υπόψιν τυχόν περιορισμούς του συστήματος. Η συγκεκριμένη διαδικασία ονομάζεται διαδικασία ενοποιημένου προγραμματισμού (ΔΕΠ) και διενεργείται 3 φορές (στις 17:00 την D-1, στις 00:00 και στις 10:30 την D).

Σε περίπτωση σημαντικών αποκλίσεων, ο διαχειριστής μπορεί να χρειαστεί να εκτελέσει ενέργειες εξισορρόπησης του συστήματος. Πρακτικά αυτό σημαίνει να αυξομειωθεί η παραγωγή ή η ζήτηση από συγκεκριμένους παραγωγούς ή καταναλωτές που έχουν τέτοια δυνατότητα. Αυτό γεννά την ανάγκη να αποζημιωθούν αυτοί οι συμμετέχοντες για την «διευκόλυνση» που παρείχαν στο σύστημα. Για να αποζημιωθούν, ο διαχειριστής χρεώνει ανάλογα τους υπαίτιους για την δημιουργία των αποκλίσεων. Ο προσδιορισμός του κόστους εξισορρόπησης, των αποκλίσεων και ο καταμερισμός χρεώσεων και πιστώσεων στους αντίστοιχους συμμετέχοντες είναι ευθύνη της αγοράς εκκαθάρισης αποκλίσεων.

V. Ενεργειακή χρηματοπιστωτική αγορά

Η χρηματοπιστωτική αγορά ενέργειας προσφέρει την δυνατότητα στους συμμετέχοντες να συναλλαχθούν σε παράγωγα προϊόντα με σκοπό την διαχείριση ρίσκου ή την κερδοσκοπία. Τα διαθέσιμα προϊόντα είναι τα Συμβόλαια Μελλοντικής Εκπλήρωσης (ΣΜΕ) και τα Δικαιώματα Προαίρεσης.

Τα ΣΜΕ πρόκειται για συμβόλαια που υποχρεώνουν τα δύο μέρη να συναλλάξουν ενέργεια σε μελλοντικό χρόνο σε προκαθορισμένη τιμή. Αυτό δίνει πρακτικά την δυνατότητα σε συμμετέχοντες να «κλειδώσουν» την τιμή της ενέργειας που θα παράξουν ή θα καταναλώσουν και να εξαλείψουν έτσι το κίνδυνο από την μεταβλητότητα της τιμής της ενέργειας. Τα συμβόλαια έχουν την δυνατότητα να εκκαθαριστούν είτε με φυσική παράδοση της ενέργειας είτε με χρηματικό διακανονισμό.

Τα Δικαιώματα Προαίρεσης δίνουν το δικαίωμα, αλλά όχι την υποχρέωση, στον αγοραστή του δικαιώματος να αγοράσει ή να πουλήσει ενέργεια σε μελλοντικό χρόνο σε προκαθορισμένη τιμή. Ο αγοραστής του δικαιώματος πληρώνει

αντάλλαγμα στον πωλητή του δικαιώματος, καθώς ο δεύτερος είναι υποχρεωμένος να συναλλαχθεί εάν ο αγοραστής το επιλέξει.

2.1 Μεταφορά ηλεκτρικής ενέργειας και ο ρόλος του διαχειριστή

Η βασική ιδιαιτερότητα του χρηματιστηρίου ενέργειας οφείλεται στη φύση του αγαθού που αγοραπωλείται, καθώς η παραγωγή και η κατανάλωση του πρέπει να γίνεται ταυτόχρονα. Για να κατανοήσουμε γιατί ο λόγος αυτός αποτελεί μεγάλη διαφορά, με αποτέλεσμα να δημιουργείται μια αγορά εντελώς διαφορετική από αυτές που γνωρίζουμε, όπως το χρηματιστήριο αξιών, θα πρέπει να αναλύσουμε συνοπτικά βασικές τεχνικές προδιαγραφές της αγοράς και της ηλεκτρικής ενέργειας γενικότερα.

Η φύση της ηλεκτρικής ενέργειας απαιτεί να καταναλωθεί άμεσα ή να αποθηκευτεί. Η αποθήκευση της, όμως, είναι μια ακριβή λύση και, μέχρι πρότινος απαγορευτική. Μόνο τα τελευταία χρόνια τα κράτη-μέλη της ευρωπαϊκής ένωσης προώθησαν την αποθήκευση ηλεκτρικής ενέργειας, εξαιτίας της ακρίβειας του ηλεκτρικού ρεύματος, με αποτέλεσμα η αγορά της αποθήκευσης να βρίσκεται ακόμα σε σχετικά πρώιμο στάδιο. Αυτό οδηγεί στο συμπέρασμα ότι η ηλεκτρική ενέργεια που παράγεται, μέχρι και σήμερα, πρέπει να καταναλώνεται άμεσα. Ποιο είναι το πρόβλημα, όμως, που δημιουργείται;

Για να απαντήσουμε στο παραπάνω ερώτημα, θα πρέπει να αναφερθούμε στη μεταφορά της ηλεκτρικής ενέργειας. Η μεταφορά της ηλεκτρικής ενέργειας γίνεται μέσω ενός δικτύου. Το σημείο ισορροπίας του δικτύου της ηλεκτρικής ενέργειας στο ευρωπαϊκό δίκτυο είναι 50 Hertz, στο οποίο συμμετέχουν οντότητες με ευθύνες εξισορρόπησης και οφείλουν να διατηρούν αυτή τη συχνότητα κάθε στιγμή πολύ κοντά σε αυτή τη τιμή (± 0.50 Hertz), ώστε να μην υπάρξει διακοπή ρεύματος. Η συχνότητα αυτή αυξάνεται όταν η παραγωγή ηλεκτρικής ενέργειας είναι πολύ μεγαλύτερη από την κατανάλωση της, ενώ μειώνεται, όταν ισχύει το αντίστροφο. Η πρόβλεψη της κατανάλωσης και της παραγωγής ενέργειας δεν είναι εύκολη. Οι αποκλίσεις των προβλέψεων που συμβαίνουν για λόγους που θα εξεταστούν στην επόμενη ενότητα, οδηγούν σε ανισορροπία του δικτύου ηλεκτρικής ενέργειας. Για το λόγο αυτό κάθε δίκτυο ελέγχεται από ένα Διαχειριστή Συστήματος Μεταφοράς (TSO), ο οποίος είναι υπεύθυνος για την εξισορρόπηση του δικτύου.

Ο Διαχειριστής Συστήματος Μεταφοράς (TSO) είναι υπεύθυνος για την μεταφορά της ενέργειας στην υψηλή και μέση τάση. Στην πράξη αφορά την μεταφορά της ενέργειας, για μεγάλες αποστάσεις, από τους παραγωγούς στα αστικά κέντρα. Το δίκτυο συμπληρώνει ο Διαχειριστής Διανομής της Ηλεκτρικής Ενέργειας (DSO) οποίος είναι υπεύθυνος για την διανομή της ενέργειας στη χαμηλή τάση. Ουσιαστικά ο DSO παραλαμβάνει την ενέργεια από τον TSO και την διανέμει στα νοικοκυριά. Η αποτελεσματική συνεργασία των δύο εξασφαλίζει την σταθερότητα όλου του δικτύου μεταφοράς και διανομής. Πιο συγκεκριμένα, ο TSO αναπτύσσει, συντηρεί και προστατεύει όλες τις σημαντικές υποδομές του δικτύου μεταφοράς ενέργειας. Συλλέγει και διανέμει πληροφορίες για τα προγράμματα παραγωγής και κατανάλωσης, προγραμματισμένες περιόδους διακοπών λειτουργίας, και δεδομένων εξισορρόπησης. Τέλος, ο TSO είναι υπεύθυνος για την εξισορρόπηση του συστήματος λαμβάνοντας δράσεις αποσυμφόρησης του δικτύου, σε συνεργασία με τον DSO, που θα περιγραφούν παρακάτω. Σε αυτό το σημείο αξίζει να σημειωθεί ότι τον ρόλο του TSO στην Ελλάδα αναλαμβάνει ο ΑΔΜΗΕ, ενώ του DSO ο ΔΕΔΔΗΕ.

2.2 Αιτίες δημιουργίας ανισορροπιών στο δίκτυο

Η ενεργειακή μετάβαση από καύσιμα σε ανανεώσιμες πηγές ενέργειας έχει προκαλέσει νέες απαιτήσεις από τον διαχειριστή του δικτύου όσον αφορά την σταθερότητα του δικτύου. Ένα δίκτυο που βασίζεται σε συμβατική παραγωγή μπορεί να ανταπεξέλθει σε μεταβολές της ζήτησης σχετικά εύκολα αλλάζοντας το επίπεδο της παραγωγής κατά το δοκούν. Εφόσον η παραγωγή σε τέτοια δίκτυα είναι συγκεντρωμένη και ελέγχεται από τον διαχειριστή, η εξισορρόπηση με αυτή την μέθοδο είναι αποτελεσματική. Αντιθέτως, η εισαγωγή των ΑΠΕ στο δίκτυο δημιουργεί αβεβαιότητα για το συνολικό επίπεδο παραγωγής κάθε δεδομένη στιγμή. Αυτό προκύπτει καθώς η ηλιοφάνεια για ένα φωτοβολταϊκό πάρκο και ο αέρας για ένα αιολικό δεν είναι δεδομένα, ούτε ελεγχόμενα. Τα προβλήματα που προκαλούν αυτές οι ανωμαλίες της παραγωγής των ΑΠΕ συνοψίζονται σε τρεις πυλώνες.

1 Ανωμαλία συχνότητας και τάσης

Οι μεταβολές στην ηλιοφάνεια ή την ένταση του αέρα δημιουργούν διακύμανση στην συχνότητα και την τάση στην οποία παράγουν τα ΑΠΕ. Οι υπάρχοντες μετασχηματιστές δεν επαρκούν και αυτό σε συνδυασμό με τις εναλλασσόμενες καιρικές συνθήκες επηρεάζουν την λειτουργία του δικτύου.

2 Υπερφόρτωση γραμμών δικτύου

Υπάρχουν ώρες αιχμής όπου οι παραγωγοί είναι πιθανό να παράγουν υπερβάλλουσα ενέργεια δημιουργώντας συμφόρηση στις γραμμές του δικτύου.

3 Ανισορροπία ζήτησης και προσφοράς

Είτε υπερβάλλουσα είτε ελλιπής παραγωγή από τους παραγωγούς ΑΠΕ δημιουργεί άμεσα πλεόνασμα ή έλλειμμα στο δίκτυο. Αυτό δημιουργεί την ανάγκη για εξισορρόπηση του δικτύου και κόστη για τον διαχειριστή.

2.3 Τρόποι εξισορρόπησης δικτύου

Η εξισορρόπηση του δικτύου, όπως αναλύσαμε προηγουμένως, είναι αναγκαία για την παροχή ηλεκτρικής ενέργειας κάθε χρονική στιγμή. Για το λόγο αυτό, έχουν δημιουργηθεί διάφορες μέθοδοι που επιδιώκουν τη διόρθωση ανισορροπιών που δημιουργούνται εξαιτίας σφαλμάτων στην πρόβλεψη κατανάλωσης και παραγωγής.

A) Αυξομειώσεις στην παραγωγή και κατανάλωση της ηλεκτρικής ενέργειας

Ο Διαχειριστής του δικτύου (TSO) μπορεί να ζητήσει την αύξηση ή τη μείωση παραγωγής ηλεκτρικής ενέργειας, καθώς και την αύξηση ή τη μείωση της κατανάλωσης ενός τελικού καταναλωτή, συνήθως μεγάλα εργοστάσια που καταναλώνουν μεγάλες ποσότητες ενέργειας. Αυτό δημιουργεί την ανάγκη μιας αγοράς, στην οποία παραγωγικές μονάδες, οι οποίες μπορούν να ελέγξουν την παραγωγή τους άμεσα, όπως τα θερμοηλεκτρικά, οι ανεμογεννήτριες με Pitch control συστήματα, συμμετέχουν και δηλώνουν τα διαθέσιμα αποθέματα ενέργειας που μπορούν να αυξήσουν ή να μειώσουν σε περίπτωση ανισορροπίας του δικτύου. Αντίστοιχα, «μεγάλοι καταναλωτές» δηλώνουν κατά πόσο μπορούν να αυξήσουν ή να μειώσουν την κατανάλωση τους. Μέχρι σήμερα, με τον όρο «μεγάλοι καταναλωτές» εννοούμε εργοστάσια παραγωγής διάφορων αγαθών και υπηρεσιών, οντότητες,

δηλαδή, που έχουν τη δυνατότητα να καταναλώσουν μεγάλο φορτίο ενέργειας. Η ηλεκτροκίνηση, ωστόσο, η οποία προωθείται από πολλά κράτη και αγορές, δημιουργεί το ενδεχόμενο ενός μεγάλου κατανεμημένου φορτίου, με την προϋπόθεση ότι θα υπάρξει συντονισμός μεταξύ των διάφορων ηλεκτρικών στόλων αυτοκινήτων, ικανού να καλύψει τις ανάγκες του διαχειριστή για την εξισορρόπηση του δικτύου.

B) Απόρριψη φορτίου

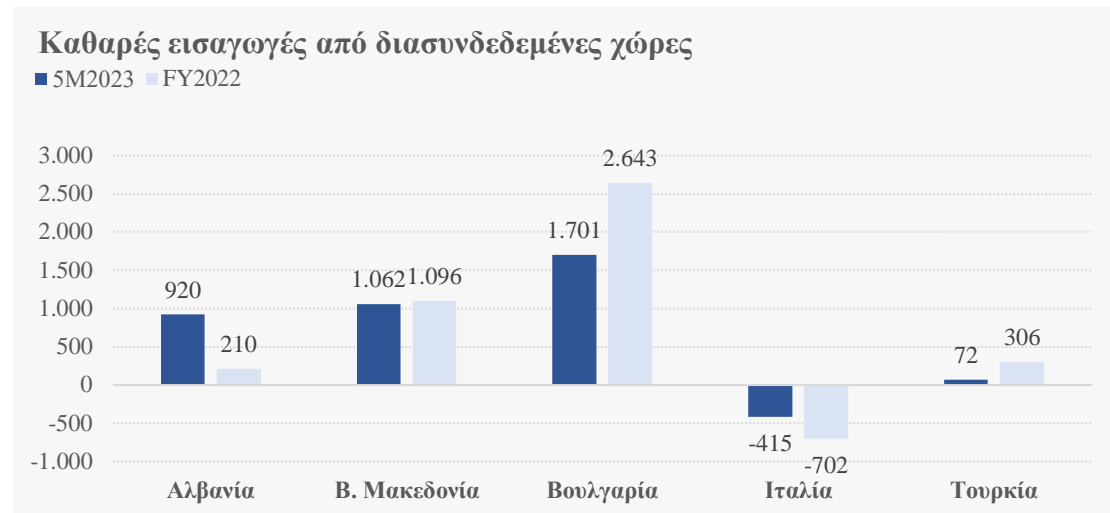
Η απόρριψη φορτίου (load shedding) είναι ένας τρόπος κατανομής της ζήτησης ηλεκτρικής ενέργειας και χρησιμοποιείται, όταν η κατανάλωση ηλεκτρικής ενέργειας είναι πολύ μεγαλύτερη από την παραγωγή της. Ο Διαχειριστής του δικτύου έχει την επιλογή να κόβει την παροχή ρεύματος σε συγκεκριμένους καταναλωτές, μέχρι το σύστημα να έρθει πάλι σε ισορροπία. Είναι ένας μηχανισμός που μοιάζει με τον προηγούμενο, ωστόσο σε αυτή την περίπτωση η διακοπή του ρεύματος γίνεται βεβιασμένα από το διαχειριστή. Για το λόγο αυτό, είναι ένας μηχανισμός, ο οποίος ενεργοποιείται μόνο σε περιπτώσεις κρίσιμης ανάγκης, όπως φυσικές καταστροφές.

Γ) Αποθήκευση ηλεκτρικής ενέργειας

Η αποθήκευση ηλεκτρικής ενέργειας αποτελεί ένα νέο μηχανισμό. Βασίζεται στην λογική του (Α) και αποτελεί ένα τρόπο αύξησης των αποθεμάτων ηλεκτρικής ενέργειας τόσο για σκοπούς κατανάλωσης, όσο και για σκοπούς παραγωγής.

Δ) Εισαγωγή/Εξαγωγή ηλεκτρικής ενέργειας

Ο Διαχειριστής έχει τη δυνατότητα να εισάγει/εξάγει ενέργεια από άλλα «γειτονικά» δίκτυα μέσω δικτύων υψηλής τάσης, με σκοπό να εξισορροπηθεί τόσο το δίκτυο που διαχειρίζεται, όσο και των «γειτονικών» δικτύων. Για την Ελλάδα, διασυνδεδεμένες χώρες είναι η Αλβανία, η Βόρεια Μακεδονία, η Βουλγαρία, η Τουρκία και η Ιταλία. Για το 5μηνο του 2023 εμφανίζονται καθαρές εισαγωγές για την χώρα μας από όλες τις διασυνδεδεμένες χώρες με εξαίρεση την Ιταλία, όπου η Ελλάδα έχει καθαρές εξαγωγές.



2.4 Διαχείριση συμφόρησης δικτύου (αλγόριθμοι case studies χωρών)

Οι τρόποι διαχείρισης της συμφόρησης του δικτύου μπορεί να έχουν στόχο την διαχείριση προβλεπόμενης συμφόρησης, να έχουν στόχο την πραγματική αποσυμφόρηση του δικτύου, ή απλά την κατανομή της ενέργειας σε καταναλωτές. Παρακάτω συνοψίζονται μέθοδοι αποσυμφόρησης του δικτύου με παραδείγματα από χώρες που τα εφαρμόζουν.

Η πρώτη συμβατική μέθοδος αποσυμφόρησης βασίζεται στην εξυπηρέτηση της ζήτησης με χρονική σειρά προτεραιότητας, δηλαδή οι ζητήσεις εξυπηρετούνται με βάση το ποια ήρθε πρώτη. Από αυτό το χαρακτηριστικό η μέθοδος λαμβάνει και το όνομά της, “first come, first serve”. Το πλεονέκτημα της μεθόδου είναι ότι για να εξασφαλίσουν οι καταναλωτές ότι θα εξυπηρετηθούν αναγκάζονται να κάνουν πιο μακροχρόνιες προβλέψεις. Για να αποφευχθεί η περίπτωση όπου οι καταναλωτές υπερεκτιμούν τις ανάγκες τους ώστε να εξυπηρετηθούν σίγουρα, η Ιαπωνία χρησιμοποιεί αυτή την μέθοδο μαζί με την “use it or lose it”. Σε αυτή, όταν ένας καταναλωτής δεν χρησιμοποιήσει την ενέργεια που αγόρασε, την «χάνει», δηλαδή την πληρώνει έτσι κι αλλιώς. Με αυτό το κίνητρο οι προβλέψεις συστηματικά βελτιώνονται και ο διαχειριστής έχει την δυνατότητα να σχεδιάσει καλύτερα την προσφορά. Παρόλα αυτά, ένα τέτοιο μοντέλο αποσυμφόρησης δεν δίνει λύση σε αγορές όπως η day ahead και η συνεχής διαπραγμάτευση. Παρομοίως απλοϊκή είναι και η μέθοδος “pro-rata” κατά την οποία η ενέργεια και τυχόν αποκλίσεις μοιράζονται στους καταναλωτές

αναλογικά με τις ανάγκες τους. Είναι εύκολο να καταλάβει κανείς πως ένα τέτοιο σύστημα είναι δύσκολο να εφαρμοστεί στην πράξη και να έχει βέλτιστο κοινωνικό όφελος.

Η Νορβηγία ακολουθεί την μέθοδο του “market splitting”. Σε αυτή την μέθοδο ο διαχειριστής σχεδιάζει αρχικά την προσφορά χωρίς περιορισμούς. Αμέσως μετά αναγνωρίζει τα σημεία του συστήματος που προκαλούν συμφόρηση και τα «διαχωρίζει». Τέλος, διαχειρίζεται τα προβληματικά σημεία «μετακινώντας» διαθέσιμη ενέργεια από το ένα στο άλλο. Επιπλέον, γενικά στις σκανδιναβικές χώρες χρησιμοποιείται και η μέθοδος του “redispatching” όπου ορισμένοι παραγωγοί μειώνουν την παραγωγή τους ενώ άλλοι την αυξάνουν, κατ’ εντολή του διαχειριστή. Με αυτό το τρόπο δεν αλλάζει το συνολικό φορτίο αλλά επιτυγχάνεται η αποσυμφόρηση σε τοπικά μέρη του δικτύου. Αυτή μέθοδος δημιουργεί κόστος για τον διαχειριστή διότι πρέπει να αποζημιώσει του παραγωγούς για την μείωση της παραγωγής τους. Αυτά τα κόστη καταλήγουν στους καταναλωτές μέσω των χρεώσεων του διαχειριστή.

ΚΕΦΑΛΑΙΟ 3

ΗΛΕΚΤΡΟΚΙΝΗΣΗ, DECENTRALIZED ENERGY
FLEXIBILITY MARKETPLACE, UNIT
COMMITMENT ΠΡΟΒΛΗΜΑ

3. Ηλεκτροκίνηση, Decentralized Energy Flexibility

Marketplace και Unit Commitment πρόβλημα

3.1 Ηλεκτροκίνηση σήμερα και παρουσίαση Ευέλικτης Αποκεντρωμένης Αγοράς

Όπως προαναφέραμε, η Ευρωπαϊκή ένωση, τα τελευταία χρόνια, προωθεί και στηρίζει την ηλεκτροκίνηση θεσπίζοντας διάφορους κανονισμούς και οδηγίες. Είναι αποδεδειγμένο ότι ο τομέας των μεταφορών αποτελεί έναν από τους κυριότερους λόγους για το φαινόμενο του θερμοκηπίου. Σύμφωνα με πρόσφατο κανονισμό της Ευρωπαϊκής Ένωσης που στοχεύει στη μείωση των εκπομπών διοξειδίου του άνθρακα, η ένωση θέτει τους παρακάτω στόχους:

- το αργότερο μέχρι το 2035 το ποσοστό των νέων αυτοκινήτων που θα έχουν μηδενικές εκπομπές ρύπων θα πρέπει να είναι 100%, με στόχο την πλήρη απαλλαγή των οδικών μεταφορών από τις εκπομπές άνθρακα έως το 2050. Η Ελλάδα καθιέρωσε το πρόγραμμα «ΚΙΝΟΥΜΑΙ ΗΛΕΚΤΡΙΚΑ», το οποίο επιδοτεί αγορές ηλεκτρικών οχημάτων με ποσοστά, τα οποία είναι ικανά να ανταγωνιστούν τις μικρότερες τιμές των οχημάτων με κινητήρες εσωτερικής καύσεως.
- ανάπτυξη δικτύου φορτιστών ηλεκτρικών οχημάτων, ικανού να καλύψει όλες τις απαραίτητες ανάγκες των ηλεκτρικών οχημάτων. Στο πλαίσιο αυτό, οι δήμοι της Ελλάδας είχαν την υποχρέωση υλοποίησης ενός ΣΦΗΟ (Σχέδιο Φόρτισης Ηλεκτρικών Οχημάτων), λαμβάνοντας παραμέτρους κυκλοφοριακούς, χωροταξικούς και πολεοδομικούς, με σκοπό την εγκατάσταση φορτιστών ηλεκτρικών οχημάτων, ανάλογα με την αναμενομένη διείσδυση ηλεκτρικών οχημάτων. Σε κάθε περίπτωση, θα πρέπει ένας ηλεκτρικός φορτιστής να αντιστοιχεί το πολύ σε 1000 κατοίκους.

Πλην της Ευρωπαϊκής Ένωσης, όμως, μεγάλες βιομηχανίες οχημάτων έχουν θέσει και αυτοί με τη σειρά τους στόχους για την πλήρη απεξάρτηση από αυτοκίνητα με κινητήρες εσωτερικής καύσης. Ο παρακάτω πίνακας (Πίνακας 1) δείχνει συνοπτικά μερικούς από αυτούς τους στόχους.

Όπως μπορούμε να δούμε, οι περισσότερες αυτοκινητοβιομηχανίες σχεδιάζουν να καταργήσουν τα οχήματα με κινητήρα εσωτερικής καύσης τις επόμενες δεκαετίες.

Ορισμένες αυτοκινητοβιομηχανίες, όπως η Bentley, η Mercedes-Benz και η Jaguar, έχουν ήδη ορίσει ημερομηνίες για το πότε θα πωλούν μόνο plug-in υβριδικά ηλεκτρικά οχήματα (PHEV) και οχήματα με μπαταρία (BEV). Άλλες αυτοκινητοβιομηχανίες έχουν ορίσει πιο γενικούς στόχους, όπως η Toyota θέλει μέχρι το 2030 να έχει πουλήσει 8 εκατομμύρια υβριδικά αυτοκίνητα.

Ακόμη, στον πίνακα εντοπίζεται και η διάθεση των αυτοκινητοβιομηχανιών για χρηματική επένδυση στην ηλεκτροκίνηση. Συγκεκριμένα, μέχρι το 2025, η General Motors, η Mercedes-Benz και η Volkswagen Group έχουν στόχο να επενδύσουν στην ηλεκτροκίνηση τα μεγάλα ποσά των 35, 47 και 86 δισεκατομμυρίων δολαρίων αντίστοιχα. Ωστόσο, αυτή η συμπεριφορά δεν παρατηρείται από όλες τις εταιρείες. Συγκεκριμένα, εταιρείες που έχουν την έδρα τους στην Ασία (Mazda, Honda, Nissan) κρατάνε μια στάση πιο επιφυλακτική προς την ηλεκτροκίνηση. Κύριο επιχείρημα τους είναι πως μια βιαστική μετατροπή όλων των αυτοκινήτων σε ηλεκτρικά θα δημιουργήσει ένα φορτίο ρεύματος τόσο μεγάλο, που οι ενεργειακές ανάγκες δε θα μπορούν να εξυπηρετηθούν από ανανεώσιμες πηγές μόνο, με αποτέλεσμα η καύση πετρελαίου και φυσικού αερίου για παραγωγή ενέργειας για την στήριξη του παγκόσμιου στόλου αυτοκινήτων να δημιουργεί μεγαλύτερο αποτύπωμα άνθρακα στο περιβάλλον. Προτείνουν, λοιπόν, την παράλληλη ανάπτυξη τεχνολογιών Ανανεώσιμων Πηγών Ενέργειας.

Συνολικά, ωστόσο, φαίνεται η αυτοκινητοβιομηχανία να δεσμεύεται για την ηλεκτροκίνηση, θέτοντας φιλόδοξους στόχους και επενδύοντας σε νέες τεχνολογίες.

Σχέδια αυτοκινητοβιομηχανίας για ηλεκτροκίνηση

Bentley	By 2030	By 2030	-	First battery EV in 2025
BMW	-	100% renewable energy by 2050	\$6.5 billion (2019 figure)	2 million fully electric vehicles by the end of 2025
Ford	-	By 2050	\$22 billion through 2025	76% carbon emissions reduction by 2035

Cadillac	By 2030	-	-	No new models with gas engines now
General Motors	By 2035	By 2040	\$35 billion through 2025	Electric Hummers and Silverado coming
Honda	2022 (Europe), 2040 (North America)	By 2050	-	2 new EVs in 2024 to be built by GM
Hyundai-Kia	-	Undated pledge	\$7.4 billion in the U.S. by 2025	23 types of EVs and hydrogen cars by 2025
Jaguar Land Rover	By 2030	By 2039	\$3.5 billion annually	6 electric Land Rovers over the next 5 years
Mazda	-	By 2050	-	First battery EV is the MX-30 in the fall
Mercedes-Benz	All new platforms EV-only in 2025 2030 with caveats in some markets	By 2039	\$47 billion between 2022 and 2030	EQS luxury sedan on sale this fall
Mitsubishi	-	25% carbon reduction by 2030	-	A plug-in hybrid focus. The Airtrek EV shown for the Chinese market.
Nissan	-	By 2050	\$1.3 billion on EV hub in England	8 EVs on the road by the end of 2023
Rolls-Royce	EVs will be available in 90% of segments	By 2030	-	Silent Shadow is in development, using BMW technology
Stellantis	70% of European sales, 40% of North American sales electrified in 4 years	-	\$35.5 billion in EV spending through 2025	55 electrified cars and trucks for sale in the U.S. and Europe by 2025

Toyota	8 million electrified vehicles by 2030	By 2050	-	70 electrified models by 2025, 15 of them battery EVs
Volkswagen Group	50% fully electric sales in U.S. by 2030 Last new combustion platform in 2026	By 2050	\$86 billion through 2025	The VW Group has 70 new electrified models in the pipeline
Volvo	By 2025, half of global sales fully electric	By 2040	\$1 billion annually on electrification and autonomy	All fully electric models will be available online only

Συμπεραίνουμε, λοιπόν, ότι η ηλεκτροκίνηση είναι ένα πολύ φλέγον ζήτημα που θα απασχολεί, όχι μόνο την Ευρώπη, αλλά όλο τον κόσμο, την επόμενη δεκαετία. Η μετάβαση, όμως, από αυτοκίνητα με κινητήρες εσωτερικής καύσης σε κινητήρες ηλεκτρικούς δεν είναι εύκολη. Σύμφωνα με έρευνα της Hedges Company, μέχρι το 2022 υπολογίστηκε ότι υπήρχαν περίπου 1.5 δισεκατομμύρια αυτοκίνητα. Η συντριπτική πλειοψηφία αυτών είναι αυτοκίνητα με κινητήρες εσωτερικής καύσης. Καταλαβαίνουμε, λοιπόν, ότι η μετατροπή ενός τέτοιου τεράστιου στόλου είναι ένα δύσκολο και απαιτητικό πρόβλημα. Εκτός από την μετατροπή, θα πρέπει να ληφθούν υπόψιν και άλλοι παράγοντες, όπως η κατασκευή επαρκών υποδομών για τη φόρτιση των ηλεκτρικών οχημάτων, αλλά και η βελτίωση της τεχνολογίας τους, ώστε να προσφέρουν δυνατότητες ισοδύναμες με αυτές των κινητήρων εσωτερικής καύσης, όπως επαρκή χιλιομετρική αυτονομία. Τέλος, το τεράστιο πλήθος ηλεκτρικών οχημάτων θα οδηγήσει και στην αύξηση των ενεργειακών αναγκών, καθώς δημιουργείται ένα τεράστιο φορτίο ενέργειας.

Για το λόγο αυτό, πολλές δράσεις έχουν ξεκινήσει για την ομαλή μετάβαση στην ηλεκτροκίνηση. Μία από αυτές είναι το πιλοτικό πρόγραμμα Decentralized Energy Flexibility Marketplace (Ευέλικτη Αποκεντρωμένη Αγορά) του SOFIE. Το πρόγραμμα αυτό επιδιώκει την εφαρμογή λύσεων που βασίζονται σε blockchain στο πλαίσιο των ενεργειακών αγορών, χρησιμοποιώντας τεχνολογίες όπως τα «έξυπνα συμβόλαια» και τα ανταλλάξιμα κρυπτονομίσματα. Πιο συγκεκριμένα, το πρόγραμμα αφορά μια αγορά

την οποία ελέγχει ένας Διαχειριστής Ηλεκτρικού Δικτύου (DSO, TSO) και στην οποία δημιουργεί δημοπρασίες. Αυτές οι δημοπρασίες αφορούν χρονικά slots με προσφερόμενη ενέργεια προς κατανάλωση, κατά τα οποία, ο Διαχειριστής έχει προβλέψει ότι η διοχετευόμενη ενέργεια στο δίκτυο θα είναι μεγαλύτερη από την απορροφημένη, με αποτέλεσμα να δημιουργείται αστάθεια στο δίκτυο. Σε αυτές τις δημοπρασίες, μπορούν να συμμετέχουν μόνο διαχειριστές στόλων ηλεκτρικών αυτοκινήτων, οι οποίοι προσφέρονται να φορτίσουν το συγκεκριμένο χρονικό slot μέρος του στόλου τους. Ο Διαχειριστής του δικτύου προσφέρει στη δημοπρασία που δημιούργησε ένα συγκεκριμένο αριθμό token, τα οποία έχουν συγκεκριμένη οικονομική αξία (π.χ. 0,002euro/Watt), και οι διαχειριστές των στόλων κάνουν προσφορές για να κερδίσουν αυτά τα tokens. Ο Διαχειριστής του δικτύου, τέλος, αποφασίζει ποιοι στόλοι αυτοκινήτων πρέπει να φορτίσουν το κάθε χρονικό slot και, επομένως, να αποκομίσουν τα tokens. Όλες αυτές οι συναλλαγές, γίνονται με τη χρήση blockchain τεχνολογιών, ώστε να εξασφαλίζεται έτσι η ασφάλεια τους, αλλά κυρίως η διαφάνεια και η αμεροληψία του Διαχειριστή Δικτύου.

Το πρόβλημα των «νικητών» μιας δημοπρασίας, όμως, έχει πολλές ιδιαιτερότητες. Μία από αυτές είναι ότι στην κάθε δημοπρασία, μπορεί να υπάρξουν πολλοί νικητές, καθώς εκτιμάται ότι οι φορές που ένας στόλος αυτοκινήτων θα είναι ικανός να καλύψει μόνος του τις ανάγκες του δικτύου για να εξισορροπηθεί είναι απειροελάχιστες. Ακόμη, πρέπει να ληφθεί υπόψιν ότι οι φορτιστές ηλεκτρικών οχημάτων έχουν περιορισμένες δυνατότητες, ανάλογα πάντα και της τεχνολογίας του, όπως και οι μπαταρίες των ηλεκτρικών αυτοκινήτων. Για παράδειγμα, υπάρχουν διάφοροι τύποι φορτιστών ως προς τη ταχύτητα της φόρτισης του. Οι πιο συνηθισμένοι φορτιστές έχουν ισχύ 7kW, 11kW και 22kW, γεγονός που σημαίνει ότι μπορούν να φορτίσουν ένα όχημα με 7kWh, 11kWh και 22kWh αντίστοιχα στη διάρκεια μίας ώρας. Αν αναλογιστούμε ότι τα χρονικά slot που δημοπρατούνται αφορούν διάρκεια 15 λεπτών, τότε η ενέργεια που μπορούν να διοχετεύσουν σε ένα ηλεκτρικό όχημα είναι ακόμα μικρότερη. Συγκεκριμένα, οι προαναφερθέντες φορτιστές θα μπορούσαν να φορτίσουν 1.75kWh, 2.75kWh και 5.5kWh αντίστοιχα. Γιατί όμως αυτό αποτελεί πρόβλημα;

Αν το δίκτυο βρίσκεται σε κατάσταση ανισορροπίας, η ενέργεια που θα απαιτείται για να βρεθεί σε ισορροπία θα είναι συνήθως διαφορετική τάξη μεγέθους. Για παράδειγμα, ένα εργοστάσιο μεγάλης παραγωγής (επομένως και κατανάλωσης) σε ώρες λειτουργίας καταναλώνει μέχρι και 200kWh την ώρα (50kWh σε ένα τέταρτο), όση ενέργεια

περίπου καταναλώνουν 9 φορτιστές των 22kW. Υπάρχουν βέβαια και οι ταχυφορτιστές οι οποίοι μπορούν να φτάσουν αυτές τις καταναλώσεις, αλλά, τόσο η εγκατάσταση τους, όσο και το κόστος του, κάνουν αυτή τη λύση αρκετά λιγότερο προσιτή. Οι αποκλίσεις που φέρνουν το δίκτυο σε ανισορροπία είναι της τάξεως των MWh στην Ελλάδα και επομένως, μπορούμε εύκολα, πλέον, να κατανοήσουμε γιατί σε μια τέτοια πιθανή δημοπρασία ενός DSO οι νικητές πρέπει να είναι πολλοί. Ο DSO καλείται, δηλαδή, να επιλέξει ποιοι φορτιστές πρέπει να «ενεργοποιηθούν», ώστε να ισορροπήσει το δίκτυο, και, στη συνέχεια, να επιλέξει ποια αμάξια πρέπει να φορτίσουν, ανάλογα με την καλύτερη προσφορά. Καλύτερες προσφορές ανά σταθμό φόρτισης ηλεκτρικών οχημάτων θεωρούνται αυτές, όπου ο συνδυασμός τους καλύπτει την ανάγκη του δικτύου για ισορροπία ως προς την προσφερόμενη διαθέσιμη ισχύ και επιτυγχάνει το μικρότερο δυνατό κόστος για το DSO ως προς την τιμή. Σε περίπτωση ίδιων προσφορών, οι κανόνες του χρηματιστήριου μπορούν να εφαρμοστούν, όπου η προσφορά, η οποία έγινε νωρίτερα χρονικά, κερδίζει.

Καταλαβαίνουμε, τελικά, ότι η λύση μια τέτοιας δημοπρασίας αποτελεί ένα πολύπλοκο πρόβλημα. Το πρόβλημα αυτό θα μπορούσε να χαρακτηριστεί σαν ένα πρόβλημα Unit Commitment.

Το πρόβλημα Unit Commitment (UC) στην παραγωγή ηλεκτρικής ενέργειας είναι μια μεγάλη οικογένεια μαθηματικών προβλημάτων βελτιστοποίησης, όπου η παραγωγή ενός συνόλου ηλεκτρικών γεννητριών συντονίζεται για να επιτευχθεί κάποιος κοινός στόχος. Ο κυριότερος βασικός στόχος του UC είναι η εξισορρόπηση της ζήτησης ηλεκτρικής ενέργειας με τη διαθέσιμη παραγωγική ικανότητα των γεννητριών και η ελαχιστοποίηση του κόστους παραγωγής ηλεκτρικής ενέργειας. Επιπλέον, άλλος συχνός στόχος είναι η μεγιστοποίηση των εσόδων από την παραγωγή ηλεκτρικής ενέργειας με την πώληση στο δίκτυο.

Το πρόβλημα αποτελεί πρόκληση λόγω της συνεχούς μεταβολής της ζήτησης, των διαφορετικών τύπων γεννητριών με διαφορετικά χαρακτηριστικά και των περιορισμών που πρέπει να ληφθούν υπόψη, όπως οι περιορισμοί στην παραγωγική ικανότητα και οι περιορισμοί των εκπομπών αερίων.

Ήδη από την απόδοση του ορισμού του προβλήματος, γίνονται φανερές οι ομοιότητες με το πρόβλημα της λύσης μιας δημοπρασίας από έναν DSO για την εξισορρόπηση του δικτύου. Ο κυριότερος στόχος του UC είναι όμοιος με το στόχο του DSO, αν

αναλογιστούμε ότι, αντί για τη διαθέσιμη παραγωγική ικανότητα των γεννητριών, μάς απασχολεί η διαθέσιμη ισχύς των αυτοκινήτων και η μέγιστη ισχύς των φορτιστών.

Βασιζόμενοι στο παραπάνω συμπέρασμα, γεννιούνται τα εξής ερωτήματα:

1. Μπορούν να εφαρμοστούν αλγόριθμοι που λύνουν ένα πρόβλημα UC για την εύρεση νικητών σε μια δημοπρασία ενός DSO, που προσφέρει ενέργεια σε στόλους ηλεκτρικών οχημάτων για την εξισορρόπηση του δικτύου;
2. Αν ναι, τι συμπεράσματα προκύπτουν;

3.2 Έννοιες και μεταβλητές προβλήματος UC

Σκοπός της ενότητας αυτής είναι να παρουσιαστούν οι πιο γνωστές μαθηματικές και προγραμματιστικές μέθοδοι, οι οποίες χρησιμοποιούν αυτή τη στιγμή διάφοροι Διαχειριστές Δικτύων για την εξισορρόπηση του. Επιπροσθέτως, θα παρουσιαστεί το πως αυτοί οι αλγόριθμοι μπορούν να τροποποιηθούν με τέτοιο τρόπο, ώστε να εφαρμοστούν στη δική μας περίπτωση μελέτης. Πριν, όμως, αναφερθούμε στους αλγορίθμους, παρακάτω παρουσιάζονται μερικές μεταβλητές, οι οποίες χρειάζονται για να κατανοήσουμε καλύτερα τις μεθόδους που χρησιμοποιούνται για τη βελτιστοποίηση του προβλήματος του UC:

1. Binary Commitment Variables (On/off status) (0/1):

Αυτές οι δυαδικές μεταβλητές αντιπροσωπεύουν την κατάσταση δέσμευσης κάθε μονάδας παραγωγής ενέργειας. Λαμβάνουν την τιμή 1 εάν η μονάδα έχει δεσμευτεί να λειτουργεί και 0 εάν είναι εκτός σύνδεσης.

Μελέτη Περίπτωσης: Θεωρούμε ότι αν γίνει αποδεκτή μια προσφορά ενός διαχειριστή στόλου ηλεκτρικών αυτοκινήτων (από εδώ και στο εξής αποκαλούμενος ως EVC aggregator) σε ένα συγκεκριμένο σταθμό φόρτισης, τότε αυτός ο σταθμός θα παίρνει την τιμή 1, διαφορετικά, αν δεν έχει γίνει καμιά αποδεκτή προσφορά σε ένα συγκεκριμένο σταθμό, τότε αυτός παίρνει την τιμή 0.

2. Power Output Variables (kW, MW):

Αυτές οι μεταβλητές αντιπροσωπεύουν την ισχύ εξόδου κάθε δεσμευμένης μονάδας παραγωγής. Είναι συνεχείς μεταβλητές και υποδεικνύουν την ποσότητα ισχύος που παράγεται από κάθε μονάδα σε κάθε χρονικό διάστημα.

Μελέτη Περίπτωσης: Θεωρούμε την ισχύ του κάθε φορτιστή σε κάθε χρονικό slot. Εάν μια αποδεκτή προσφορά από ένα EVC aggregator ή ένας συνδυασμός αποδεκτών προσφορών από ένα EVC aggregator στον ίδιο φορτιστή, είναι μεγαλύτερη η ποσότητα ισχύς που προσφέρουν από την ισχύ του φορτιστή, τότε αυτή είτε δεν γίνεται αποδεκτή, είτε γίνεται μερικώς αποδεκτή, δηλαδή γίνεται αποδεκτό το μέρος της προσφοράς της ισχύς που μπορεί να υποστηρίξει ο φορτιστής.

3. Ramp Up and Ramp Down Variables (kW, MW):

Ramp up/Ramp down είναι η διαδικασία αύξησης/μείωσης της ισχύος εξόδου μιας μονάδας παραγωγής σε μια συγκεκριμένη περίοδο. Αντιπροσωπεύει πόσο γρήγορα ένας σταθμός ηλεκτροπαραγωγής μπορεί να μεταβεί από ένα χαμηλότερο/υψηλότερο επίπεδο παραγωγής σε ένα υψηλότερο/χαμηλότερο επίπεδο παραγωγής. Μια υψηλή τιμή υποδηλώνει ότι η μονάδα μπορεί να αυξήσει/μείωση την απόδοση της γρηγορότερα, ενώ μια χαμηλότερη τιμή συνεπάγεται βραδύτερη αύξηση/μείωση της παραγωγής ενέργειας.

Μελέτη περίπτωσης: Αυτή η μεταβλητή δε θα μας απασχολήσει, καθώς στις περιπτώσεις των σταθμών φόρτισης δεν υπάρχει τέτοιος περιορισμός. Μπορεί εύκολα και γρήγορα να αυξομειώσει την ισχύ εξόδου του.

4. Start-Up and Shut-Down Variables (0/1):

Σε ορισμένες συνθέσεις, χρησιμοποιούνται πρόσθετες δυαδικές μεταβλητές για την παρακολούθηση της κατάστασης εκκίνησης και τερματισμού λειτουργίας των μονάδων παραγωγής ενέργειας. Χρησιμοποιούνται σε συνδυασμό με τις προηγούμενες μεταβλητές για να βοηθήσουν στη μοντελοποίηση αυτών των περιορισμών. Για παράδειγμα, πολλές μονάδες παραγωγής έχουν επιπρόσθετα κόστη για να εκκινήσουν τη λειτουργία τους ή να τη σταματήσουν εντελώς.

Μελέτη περίπτωσης: Αυτή η μεταβλητή δε θα μας απασχολήσει για τον ίδιο λόγο με το (3).

5. Reserve Variables (kW, MW):

Οι μεταβλητές αυτές χρησιμοποιούνται για να αναπαραστήσουν τη διαθέσιμη εφεδρική χωρητικότητα που μπορεί να χρησιμοποιηθεί για την κάλυψη απροσδόκητων αλλαγών στη ζήτηση ηλεκτρικής ενέργειας ή διακοπές παραγωγής.

Μελέτη περίπτωσης: Θεωρούμε σαν εφεδρική χωρητικότητα τη διαθέσιμη ισχύ που προσφέρει ο κάθε EVC aggregator.

6. Demand Variables (kW, MW):

Αυτές οι μεταβλητές αντιπροσωπεύουν τη ζήτηση ηλεκτρικής ενέργειας σε διαφορετικά χρονικά διαστήματα, η οποία απαιτείται από το DSO/TSO για την εξισορρόπηση του δικτύου.

Μελέτη περίπτωσης: Ως DV θεωρείται η ενέργεια, η οποία προσφέρεται σε κάθε δημοπρασία.

7. Generation Cost Variables:

Αυτές οι μεταβλητές αντιπροσωπεύουν το κόστος που σχετίζεται με τη λειτουργία κάθε μονάδας παραγωγής σε διαφορετικά επίπεδα παραγωγής. Ο στόχος του προβλήματος UC είναι συχνά η ελαχιστοποίηση αυτών των δαπανών.

Μελέτη περίπτωσης: Τα tokens τα οποία ζητάει ο κάθε EVC aggregator για τη διάθεση της ισχύς του στόλου του.

3.3 Αντικειμενική συνάρτηση προβλήματος UC και αλγόριθμοι επίλυσης

Συνοψίζοντας, σκοπός του προβλήματος UC είναι να μεγιστοποιήσει την κάλυψη της ζήτησης της ενέργειας ελαχιστοποιώντας ταυτόχρονα το κόστος για την μείωση της

παραγωγής (ή αύξησης της κατανάλωσης). Μια μαθηματική προσέγγιση που μπορεί να αποδοθεί στο πρόβλημα της ελαχιστοποίησης του κόστους της παραγωγής ενέργειας είναι:

$$C(P_i^t, U_{i,t}) = \sum_{t=1}^T \sum_{i=1}^N [F_i(P_i^t) + S_{i,t}(1 - U_{i,t-1})] U_{i,t}$$

Σκοπός της παραπάνω συνάρτησης είναι η ελαχιστοποίηση της και έχει τους παρακάτω περιορισμούς:

a) Περιορισμός ισορροπίας ισχύος

$$P_{load}^t - \sum_{i=1}^N P_i^t U_{i,t} = 0$$

b) Περιορισμός εφεδρείας

$$P_{load}^t + R^t - \sum_{i=1}^N P_{i,max} U_{i,t} \leq 0$$

c) Περιορισμός παραγωγής

$$P_{i,min} U_{i,t} \leq P_i^t \leq P_{i,max} U_{i,t}, \quad i = 1, 2, \dots, N$$

d) Κόστος εκκίνησης

$$ST_{i,t} = \begin{cases} HST_i, & \text{if } T_{i,down} \leq T_{i,off} \leq T_{i,cold} + T_{i,down} , \\ CST_i, & \text{if } T_{i,off} > T_{i,cold} + T_{i,down} , \end{cases}$$

όπου:

$F_i(P_i^t)$ – Κόστος της μονάδας i^{th} για παραγωγή P_i^t , τη χρονική στιγμή t . Τις περισσότερες φορές πρόκειται για ένα τετραγωνικό πολυώνυμο με συντελεστές a_i, b_i και c_i ως εξής:

$$F_i(P_i^t) = a_i + b_i P_i^t + c_i (P_i^t)^2$$

N - ο αριθμός των σταθμών παραγωγής

T - ο αριθμός των χρονικών slot

P_i^t - Η παραγωγή του σταθμού i την ώρα t

ST_i - i^{th} Κόστος εκκίνησης της μονάδας i^{th}

$U_{i,t}$ - Η on/off κατάσταση της μονάδας i^{th} τη χρονική στιγμή t , όπου $U_{i,t} = 0$ όταν είναι off (μη ενεργή) και $U_{i,t} = 1$ όταν είναι on (ενεργή).

P_{load}^t - $t^{[0B]}$ (σε MW)

R^t - $t^{[0B]}$ MW $^{[0B]}$)

$P_{i,max}$ - Μέγιστη πραγματική παραγωγή ενέργειας της μονάδας i (σε MW)

$P_{i,min}$ - Ελάχιστη πραγματική παραγωγή ενέργειας της μονάδας i (σε MW)

HST_i - Hot start-up κόστος της μονάδας i (σε ευρώ)

CST_i - Cold start-up κόστος της μονάδας i (σε ευρώ)

$T_{i,down}$ - Ελάχιστος χρόνος διακοπής λειτουργίας της μονάδας i (σε ώρες)

$T_{i,off}$ - Συνεχόμενος χρόνος εκτός λειτουργίας της μονάδας i (σε ώρες)

$T_{i,cold}$ - Cold start ώρες της μονάδας i (σε ώρες)

Μελέτη περίπτωσης:

Στην εξεταζόμενη περίπτωση, ο σκοπός του προβλήματος είναι ίδιος με αυτό του UC, αλλά η προσέγγιση του έχει κάποιες σημαντικές διαφορές. Όπως αναφέραμε προηγουμένως στο 3.2, δεν υπάρχουν κόστη εκκίνησης για τους σταθμούς φόρτισης ηλεκτρικών οχημάτων. Επομένως, ο περιορισμός (d) δεν θα εφαρμοστεί. Μας επηρεάζει, όμως, τα πόσα διαθέσιμα tokens έχει να προσφέρει ο TSO/DSO στους EVC aggregator. Επομένως θα πρέπει να προστεθεί ένας περιορισμός, ο οποίος θα εξασφαλίζει ότι το άθροισμα των προσφορών που γίνονται αποδεκτές κάθε φορά δεν είναι μεγαλύτερο από τα διαθέσιμα token της εκάστοτε δημοπρασίας. Ακόμη, η κάθε δημοπρασία θεωρείται στην μελέτη περίπτωσης ένα ξεχωριστό χρονικό slot. Ενώ στο πρόβλημα UC ψάχνουμε να βρούμε τη βέλτιστη λύση σε μια αλληλουχία χρονικών

slot, στο δικό μας πρόβλημα προσπαθούμε να εντοπίσουμε τη βέλτιστη λύση για τη κάθε δημοπρασία ξεχωριστά. Στο πρόβλημα UC έχει σημασία η διάσταση του χρόνου, καθώς πρέπει να συνυπολογίζονται τα Hot start-up και το Cold start-up κόστος. Στην περίπτωση μας, αυτά είναι μηδέν, άρα, η διάσταση του χρόνου δεν είναι σημαντική.

3.3.1 Dynamic Programming (DP)

Ο δυναμικός προγραμματισμός αναπτύχθηκε από τον Richard Bellman τη δεκαετία του 1940. Ο όρος «δυναμικός προγραμματισμός» προέρχεται από το γεγονός ότι η μέθοδος περιλαμβάνει την εύρεση της καλύτερης λύσης σε κάθε στάδιο του προβλήματος, λαμβάνοντας υπόψη τις προηγούμενες αποφάσεις που έχουν ληφθεί.

Ο δυναμικός προγραμματισμός είναι ένα ισχυρό εργαλείο που μπορεί να χρησιμοποιηθεί για την επίλυση μεγάλης ποικιλίας προβλημάτων. Είναι ιδιαίτερα χρήσιμο για προβλήματα που περιλαμβάνουν βελτιστοποίηση, όπως η εύρεση της συντομότερης διαδρομής μεταξύ δύο σημείων ή του πιο αποτελεσματικού τρόπου προγραμματισμού ενός συνόλου εργασιών.

Το πρόβλημα UC μπορεί να λυθεί χρησιμοποιώντας δυναμικό προγραμματισμό λόγω της ικανότητάς του να χειρίζεται αποτελεσματικά πολύπλοκα προβλήματα βελτιστοποίησης με επικαλυπτόμενα υποπροβλήματα. Ο δυναμικός προγραμματισμός αναλύει το βασικό πρόβλημα σε μικρότερα, πιο διαχειρίσιμα υποπροβλήματα, επιλύοντας κάθε υποπρόβλημα μόνο μία φορά και αποθηκεύοντας τη λύση του για μελλοντική αναφορά. Αυτή η προσέγγιση εξαλείφει τους περιττούς υπολογισμούς και βελτιώνει σημαντικά την αποτελεσματικότητα της διαδικασίας λύσης.

3.3.2 Mixed- Integer Linear Programming (MILP)

Στο MILP, ο στόχος είναι να βρεθούν οι βέλτιστες τιμές για τις μεταβλητές απόφασης που ικανοποιούν γραμμικούς περιορισμούς, ενώ, ταυτόχρονα, ελαχιστοποιούν ή μεγιστοποιούν μια αντικειμενική συνάρτηση. Ο όρος "μικτός ακέραιος" αναφέρεται στην παρουσία τόσο συνεχών μεταβλητών (οι οποίες μπορούν να λάβουν οποιαδήποτε πραγματική τιμή εντός μιας περιοχής) όσο και διακριτών μεταβλητών (που μπορούν να λάβουν μόνο ακέραιες τιμές) στο πρόβλημα βελτιστοποίησης. Η χρήση του MILP στο πρόβλημα UC προσφέρει μια ισχυρή και αποτελεσματική προσέγγιση για την

εύρεση του βέλτιστου χρονοδιαγράμματος παραγωγής ενέργειας των σταθμών παραγωγής. Το πρόβλημα UC περιλαμβάνει τη λήψη διακριτών αποφάσεων σχετικά με το εάν κάποια μονάδα παραγωγής ενέργειας θα πρέπει να είναι ενεργή ή όχι σε κάθε χρονικό διάστημα. Αυτές οι διακριτές αποφάσεις αντιπροσωπεύονται από δυαδικές μεταβλητές (0 για offline, 1 για online), καθιστώντας τις ακέραιες μεταβλητές. Επιπλέον, το πρόβλημα UC συχνά περιλαμβάνει συνεχείς μεταβλητές για την αναπαράσταση της ισχύος εξόδου κάθε μονάδας. Το MILP δέχεται φυσικά την παρουσία τόσο συνεχών όσο και διακριτών μεταβλητών, επιτρέποντας μια ενοποιημένη διατύπωση του προβλήματος UC. Ακόμη, οι περιορισμοί του UC μπορούν να αναπαρασταθούν σαν γραμμικοί περιορισμοί.

Η διαφοροποίηση του από το δυναμικό προγραμματισμό στην προσέγγιση του UC έγκειται στο τρόπο επίλυσης του. Το MILP χρησιμοποιεί ταυτόχρονα διακριτές και συνεχείς μεταβλητές για τη βελτιστοποίηση του προβλήματος, ενώ στο δυναμικό προγραμματισμό στόχος είναι η διάσπαση του προβλήματος σε πολλά μικρότερα επικαλυπτόμενα προβλήματα, η εύρεση της λύσης στο κάθε υποπρόβλημα, η αποθήκευση αυτής της λύσης και, εν τέλει, η εύρεση της καλύτερης λύσης. Επιπλέον, διαφέρουν και ως προς την πολυπλοκότητά τους. Η υπολογιστική πολυπλοκότητα στη χειρότερη περίπτωση του MILP είναι συνήθως εκθετική ως προς τον αριθμό των μεταβλητών και των περιορισμών, ενώ στο δυναμικό προγραμματισμό η πολυπλοκότητα ανεβαίνει όσο αυξάνεται ο αριθμός των υποπροβλημάτων και ο χρόνος επίλυσης του κάθε υποπροβλήματος.

3.3.3 Particle Swarm Optimization (PSO)

Ο Particle Swarm Optimization (PSO) είναι ένας αλγόριθμος μετα-ευρεστικής βελτιστοποίησης εμπνευσμένος από την κοινωνική συμπεριφορά ενός σμήνους πουλιών και της εκτροφής ψαριών. Ο αλγόριθμος αναπτύχθηκε από τους Kennedy και Eberhart το 1995.

Στο PSO, κάθε σωματίδιο (particle) αντιπροσωπεύει μια πιθανή λύση στο πρόβλημα βελτιστοποίησης. Τα σωματίδια κινούνται στον χώρο αναζήτησης και η κίνησή τους καθοδηγείται από δύο παράγοντες:

- Την καλύτερη θέση του ίδιου του σωματιδίου: Αυτή είναι η θέση που έχει βρει το σωματίδιο μέχρι στιγμής και έχει την καλύτερη τιμή, ώστε να ικανοποιεί την αντικειμενική συνάρτηση.
- καλύτερη θέση του σμήνους: Αυτή είναι η θέση που έχει βρει μέχρι στιγμής το καλύτερο σωματίδιο στο σμήνος.

Σε κάθε επανάληψη, κάθε σωματίδιο ενημερώνει τη θέση και την ταχύτητά του με βάση τη δική του καλύτερη θέση και την καλύτερη θέση του σμήνους. Οι εξισώσεις ενημέρωσης είναι οι εξής:

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (p_{best_i}^t - x_i^t) + c_2 r_2 (g_{best}^t - x_i^t)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

όπου:

- v : το διάνυσμα της ταχύτητας
- ω : το βάρος της αδράνειας το οποίο ελέγχει την ισορροπία μεταξύ εξερεύνησης και εκμετάλλευσης
- r_1, r_2 : τυχαία διανύσματα ομοιόμορφα κατανομημένα εντός του εύρους $[0,1]$
- c_1, c_2 : «συντελεστές επιτάχυνσης», οι οποίοι ελέγχουν πόσο το σωματίδιο ακολουθεί τη δική του καλύτερη θέση και την καλύτερη θέση του σμήνους, αντίστοιχα
- $p_{best_i}^t$: η καλύτερη θέση του σωματιδίου στην επανάληψη t
- g_{best}^t : η καλύτερη θέση του σμήνους στην επανάληψη t

Ο αλγόριθμος PSO επαναλαμβάνεται μέχρι να ικανοποιηθεί ένα κριτήριο διακοπής, όπως ένας μέγιστος αριθμός επαναλήψεων.

Συνοπτικά, ο αλγόριθμος PSO λειτουργεί έχοντας μια ομάδα σωματιδίων (ή λύσεων) μέσα σε ένα χώρο αναζήτησης. Κάθε σωματίδιο έχει μια τιμή «fitness», η οποία είναι ένα μέτρο του πόσο καλή είναι η λύση. Τα σωματίδια κινούνται προς τις καλύτερες λύσεις που έχουν βρει μέχρι τώρα, και επίσης κινούνται προς την καλύτερη λύση που

έχει βρει όλο το σμήνος μέχρι τώρα. Αυτή η διαδικασία επαναλαμβάνεται μέχρι το σμήνος να συγκλίνει σε μια καλή λύση.

Το PSO μπορεί να χρησιμοποιηθεί για την επίλυση του προβλήματος UC αναπαριστώντας κάθε μονάδα παραγωγής ως σωματίδιο. Τα «σωματίδια» θα μετακινούνται στη συνέχεια στον χώρο αναζήτησης, προσπαθώντας να βρουν έναν συνδυασμό αποφάσεων εκκίνησης και τερματισμού λειτουργίας που ελαχιστοποιεί το συνολικό κόστος παραγωγής. Τα «σωματίδια» θα καθοδηγούνται από τις δικές τους καλύτερες θέσεις και την καλύτερη θέση του «σμήνους». Έχει αποδειχθεί ότι είναι αποτελεσματικό για την επίλυση του προβλήματος UC, ειδικά όταν ο αριθμός των μονάδων παραγωγής/σωματιδίων είναι μεγάλος και έχει πολύπλοκους περιορισμούς.

3.3.4 Lagrangian Relaxation (LR)

Η Lagrangian Relaxation είναι μια μαθηματική τεχνική βελτιστοποίησης, η οποία μπορεί να θεωρηθεί ως επέκταση της μεθόδου των πολλαπλασιαστών Lagrange. Παρόμοια με το δυναμικό προγραμματισμό, η μέθοδος Lagrangian Relaxation επιδιώκει τη μετατροπή ενός σύνθετου προβλήματος βελτιστοποίησης σε μια σειρά απλούστερων υποπροβλημάτων. Η διαφορά της με το δυναμικό προγραμματισμό έγκειται στον τρόπο με τον οποίο καταφέρνει το προαναφερθέν. Σε αυτή τη μέθοδο, στόχος είναι να χαλαρώσουν ορισμένοι περιορισμοί του αρχικού προβλήματος με την εισαγωγή ενός συνόλου μεταβλητών που ονομάζονται πολλαπλασιαστές Lagrange για να δημιουργηθεί ένα πιο «χαλαρό» πρόβλημα από το αρχικό. Αυτές οι μεταβλητές χρησιμοποιούνται για να εντοπίζουν τυχόν παραβιάσεις των περιορισμών και να αυξάνουν το κόστος της εξεταζόμενης λύσης. Αφού βρεθεί μια λύση στο πιο «χαλαρό» πρόβλημα, αυτή η λύση, τίθεται ως ένα ανώτερο όριο στη λύση του αρχικού προβλήματος.

Για να χρησιμοποιηθεί η μέθοδος αυτή στο πρόβλημα UC, αρχικά, προστίθενται κάποιοι πολλαπλασιαστές Lagrange, ώστε να δημιουργηθεί ένα δεύτερο, πιο «χαλαρό», πρόβλημα, του οποίου η λύση μπορεί να βρεθεί πιο εύκολα, καθώς ορισμένοι από τους περιορισμούς του αρχικού προβλήματος έχουν «χαλαρώσει». Η λύση στο δεύτερο πρόβλημα παρέχει τις βέλτιστες τιμές για τους πολλαπλασιαστές, οι οποίοι αντιπροσωπεύουν το κόστος παραβίασης του κάθε περιορισμού.

Στη συνέχεια, δημιουργούνται υποπροβλήματα του εξεταζόμενου UC προβλήματος, τα οποία, πλέον, έχουν πιο χαλαρούς περιορισμούς, και οι λύσεις αυτών μπορούν να συνδυαστούν για να παρέχουν μια προσεγγιστική λύση για το αρχικό πρόβλημα UC.

ΚΕΦΑΛΑΙΟ 4
ΠΑΡΟΥΣΙΑΣΗ ΔΕΔΟΜΕΝΩΝ

4. Παρουσίαση δεδομένων

Στο κεφάλαιο (4), παρουσιάζονται τα δεδομένα που έχουν χρησιμοποιηθεί για την εκτέλεση των αλγορίθμων στο κεφάλαιο (5), καθώς και ο τύπος και η δομή τους, αλλά και μερικές από τις παραδοχές που χρειάστηκαν να γίνουν. Τα δεδομένα αφορούν τους σταθμούς φόρτισης ηλεκτρικών οχημάτων, τα ηλεκτρικά οχήματα των στόλων, τις προβλέψεις που κάνει ο διαχειριστής δικτύου για την κατανάλωση που χρειάζεται να ισορροπήσει το δίκτυο, τις δημοπρασίες που εκτελεί ο διαχειριστής του δικτύου και τις προσφορές που κάνει ο διαχειριστής του στόλου των ηλεκτρικών αυτοκινήτων.

Τα δεδομένα που είναι παρόμοια με αυτά που χρησιμοποιήθηκαν στο παραδοτέο (4.5) του SOFIE. Στη συνέχεια, αναλύονται για το κάθε τύπο δεδομένων τα στοιχεία που τα συγκροτούν καθώς και οι παραδοχές τους. Για τη μορφοποίηση των δεδομένων χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python και για την αποθήκευσή τους μία βάση δεδομένων PostgreSQL.

4.1 Δεδομένα σταθμών φόρτισης ηλεκτρικών αυτοκινήτων

Τα δεδομένα των φορτιστών που θα μας απασχολήσουν είναι 2, το `id` και το `Max Capacity`. Το `id` των σταθμών είναι ένας ακέραιος αριθμός, ο οποίος χρησιμοποιείται για λόγους ταυτοποίησης, ενώ το `Max Capacity` είναι ένας δεκαδικός αριθμός, ο οποίος παρουσιάζει τη μέγιστη ισχύ που μπορεί να παρέχει ένας σταθμός σε kW. Ωστόσο, θα πρέπει να σημειωθεί ότι από εδώ και στο εξής με τον όρο «σταθμό φόρτισης» εννοούμε ένα σύνολο σταθμών και το `Max Capacity` υποδηλώνει το άθροισμα της συνολικής ισχύος αυτών των σταθμών.

Για την εξέταση διάφορων περιπτώσεων, το άθροισμα της μέγιστης συνολικής ισχύος μπορεί να αλλάζει ή να θεωρείται αμελητέο. Οι μεταβλητές `X` και `Y` αφορούν τις συντεταγμένες της τοποθεσίας του σταθμού.

	X bigint	Y double precision	Max Capacity numeric	id [PK] integer
1	167	171.13333129882812	150	1
2	270	174.13333129882812	100	2
3	300	334.1333312988281	200	3

4.2 Ηλεκτρικά οχήματα στόλων

Σχετικά με τα δεδομένα των ηλεκτρικών οχημάτων, χρειάστηκαν να γίνουν αρκετές παραδοχές σχετικά με τα πρωτότυπα που παρέλαβα. Αρχικά, τα δεδομένα είχαν την παρακάτω μορφή:

MeasureID	VehicleID	Brand	Model	Battery Power	Battery Capacity	Autonomy (km)	Odometer (km)	Battery SoC	Timestamp
379	#1	Renault	ZOE	22	41	68	12641	60	10/16/2018 18:00
380	#1	Renault	ZOE	22	41	68	12641	60	10/16/2018 18:03
381	#1	Renault	ZOE	22	41	68	12641	60	10/16/2018 18:05
382	#1	Renault	ZOE	22	41	68	12641	60	10/16/2018 18:08
383	#1	Renault	ZOE	22	41	69	12641	61	10/16/2018 18:10
384	#1	Renault	ZOE	22	41	62	12645	56	10/16/2018 18:13
385	#1	Renault	ZOE	22	41	60	12649	53	10/16/2018 18:16
386	#1	Renault	ZOE	22	41	59	12652	52	10/16/2018 18:18
387	#1	Renault	ZOE	22	41	59	12653	51	10/16/2018 18:21
388	#1	Renault	ZOE	22	41	58	12653	51	10/16/2018 18:23
389	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:26
390	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:28
391	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:31
392	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:33
393	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:36
394	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:38
395	#1	Renault	ZOE	22	41	58	12654	51	10/16/2018 18:41
396	#1	Renault	ZOE	22	41	61	12654	53	10/16/2018 18:43
397	#1	Renault	ZOE	22	41	60	12656	52	10/16/2018 18:46
398	#1	Renault	ZOE	22	41	60	12659	51	10/16/2018 18:48

Το measureID αναφέρεται απλά σε ένα αναγνωριστικό μιας μέτρησης, ενώ το VehicleID αναφέρεται στο αναγνωριστικό ενός συγκεκριμένου αυτοκινήτου. Τα δεδομένα αυτά αφορούσαν τις μετρήσεις μόνο 7 οχημάτων, γεγονός που δεν ήταν αρκετό για να μπορούν να δημιουργηθούν αρκετοί στόλοι αυτοκινήτων, ώστε να υπάρξουν αρκετές διαφορετικές προσφορές κάθε χρονικό slot. Κάνουμε, λοιπόν την παραδοχή, ότι η κάθε μέτρηση αφορά ξεχωριστό αυτοκίνητο, με αποτέλεσμα να δημιουργούνται αρκετά δεδομένα. Οι στήλες Brand, Model, Autonomy(km) και Odometer(km) δε χρησιμοποιήθηκαν στην εφαρμογή των σεναρίων. Η στήλη Battery Capacity αναφέρεται στην συνολική χωρητικότητα της μπαταρίας ενός οχήματος και η στήλη Battery SoC αφορά το State of Charge της μπαταρίας, δηλαδή το ποσοστό της μπαταρίας το οποίο είναι φορτισμένο την χρονική στιγμή που αναφέρεται στη στήλη timestamp. Έτσι λοιπόν, αν θεωρήσουμε ότι ένα όχημα έχει x Battery Capacity και y SoC την χρονική στιγμή t, τότε η ισχύς που θα μπορούσε να διαθέσει το συγκεκριμένο αυτοκίνητο τη χρονική στιγμή t θα ισούται με $x * (y/100)$ kW. Τέλος, προστίθενται 2

στήλες που αφορούν το διαχειριστή του στόλου στο οποίο ανήκει το κάθε όχημα και το σταθμό στον οποίο είναι διατεθειμένοι να φορτίσουν. Οι τιμές αυτές γειμίζονται με τυχαίο τρόπο. Τα δεδομένα λοιπόν καταλήγουν να έχουν την παρακάτω μορφή:

	Datetime timestamp without time zone	Battery-Power bigint	Battery-Capacity bigint	Autonomy-[km] double precision	Odometer-[km] double precision	Battery double precision	aggregator double precision	cs double precision
1	2019-10-16 16:03:00		22	41	68	12641	60	3
2	2019-10-16 16:05:00		22	41	68	12641	60	4
3	2019-10-16 16:08:00		22	41	68	12641	60	4
4	2019-10-16 16:10:00		22	41	68	12641	60	2
5	2019-10-16 16:13:00		22	41	68	12641	60	2
6	2019-10-16 16:15:00		22	41	68	12641	60	2
7	2019-10-16 16:18:00		22	41	68	12641	60	3
8	2019-10-16 16:20:00		22	41	68	12641	60	2
9	2019-10-16 16:23:00		22	41	68	12641	60	2
10	2019-10-16 16:25:00		22	41	68	12641	60	4
11	2019-10-16 16:28:00		22	41	68	12641	60	4
12	2019-10-16 16:30:00		22	41	68	12641	60	1
13	2019-10-16 16:33:00		22	41	68	12641	60	2

4.3 Προβλέψεις διαχειριστή δικτύου

Τα δεδομένα αυτά αφορούν προβλέψεις που κάνει ο διαχειριστής του δικτύου, ώστε να καθορίσει τη διαθέσιμη δημοπρατούμενη ισχύ στο δίκτυο. Τα δεδομένα στην αρχική τους μορφή είχαν ένα timestamp το οποίο θεωρήθηκε ότι αναφερόταν σε ένα τέταρτο της ώρας. Το συμπέρασμα αυτό προέκυψε, καθώς το μέγεθος των δεδομένων ήταν 35.040 γραμμές, όσα είναι και τα 15λεπτα στη διάρκεια ενός ολόκληρου χρόνου. Σημειώνεται, ότι όλα τα χρονικά δεδομένα μετατράπηκαν, ώστε να αναφέρονται σε ένα συγκεκριμένο χρόνο. Η τελική μορφή το δεδομένων είναι η παρακάτω:

	timestamp double precision	Total_Evs_power_demand double precision	Reverse_power_flow_without_Evs_MW double precision	Reverse_power_flow_with_Evs_(MW) double precision	Datetime timestamp without time zone
1	43466	5.69841	2.1468	-3.55161	2019-01-01 00:00:00
2	43466.01	5.39446	1.72824	-3.66622	2019-01-01 00:15:00
3	43466.02	4.457515	2.08848	-2.369035	2019-01-01 00:30:00
4	43466.03	4.205455	2.71152	-1.493935	2019-01-01 00:45:00
5	43466.04	4.360345	2.6004	-1.759945	2019-01-01 01:00:00
6	43466.05	4.22587	2.17944	-2.04643	2019-01-01 01:15:00
7	43466.06	3.612115	2.8428	-0.769315	2019-01-01 01:30:00
8	43466.07	3.143535	2.84568	-0.297855	2019-01-01 01:45:00
9	43466.08	3.198385	3.40536	0.206975	2019-01-01 02:00:00
10	43466.09	4.54318	2.94048	-1.6027	2019-01-01 02:15:00

Η στήλη Total_Evs_power_demand αναφέρεται στη συνολική ζήτηση ενέργειας που χρειάζονται τα ηλεκτρικά αυτοκίνητα για να φορτίσουν πλήρως. Η στήλη Reverse_power_flow_without_Evs αναφέρεται στη συνολική κατανάλωση ενέργειας που απαιτείται για να έρθει το δίκτυο σε ισορροπία. Η προτελευταία στήλη, Reverse_power_flow_with_Evs αναφέρεται στην κατάσταση που θα έρθει το δίκτυο, αν όλα τα αυτοκίνητα φορτίσουν πλήρως τις μπαταρίες του τη δεδομένη χρονική

στιγμή. Μια αρνητική τιμή, λοιπόν στην προτελευταία στήλη, σημαίνει ότι το δίκτυο δε θα μπορέσει να εξυπηρετήσει τη συνολική ζήτηση της ενέργειας, και επομένως θα ζητήσει από κάποια μονάδα παραγωγής να ενεργοποιηθεί για την απαιτούμενη ενέργεια. Στόχος του προβλήματος μας είναι να φορτιστεί μόνο η περίσσεια ενέργεια της 3^{ης} στήλης της παραπάνω εικόνας (Reverse_power_flow_without_Evs) και όχι όλα τα ηλεκτρικά αυτοκίνητα ταυτόχρονα, ώστε το δίκτυο να βρίσκεται πάντα σε ισορροπία.

4.4 Δημοπρασίες

Τα δεδομένα αυτά δημιουργήθηκαν για την εκτέλεση διάφορων σεναρίων. Η δομή τους είναι απλή:

	Datetime timestamp without time zone	scaled_offers double precision	Quantity double precision
1	2019-01-01 00:00:00	531	2.1468
2	2019-01-01 00:15:00	427	1.72824
3	2019-01-01 00:30:00	516	2.08848
4	2019-01-01 00:45:00	670	2.71152
5	2019-01-01 01:00:00	643	2.6004
6	2019-01-01 01:15:00	539	2.17944
7	2019-01-01 01:30:00	702	2.8428
8	2019-01-01 01:45:00	703	2.84568

Οι στήλες Datetime και Quantity αναφέρονται στις στήλες Datetime και Reverse_power_flow_without_Evs των δεδομένων που παρουσιάστηκαν στην προηγούμενη ενότητα (4.3). Η στήλη scaled offers αναφέρεται στα tokens που είναι διατεθειμένος ο διαχειριστής του δικτύου να προσφέρει για να εξισορροπήσει το δίκτυο. Οι τιμές κυμαίνονται από 0 μέχρι 2500 tokens.

Ο τρόπος που ανατέθηκαν οι τιμές έγινε χρησιμοποιώντας τη συνάρτηση MinMaxScaler της βιβλιοθήκης scikit-learn. Η συνάρτηση αυτή εφαρμόστηκε πάνω στη στήλη Quantity, μετατρέποντας την σε τιμές, οι οποίες κυμαίνονται από το 0 μέχρι το 2500. Είναι συνάρτηση που χρησιμοποιείται για να ομαλύνει συνεχείς μεταβλητές, χρησιμοποιώντας το παρακάτω τύπο:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Επομένως, για τη μεγαλύτερη τιμή που βρίσκεται στη στήλη Quantity θα προσφερθούν 2500 tokens, ενώ για τη χαμηλότερη τιμή θα προσφερθούν 0 tokens.

4.5 Προσφορές EVC aggregator (bids)

Τα δεδομένα αυτά χρησιμοποιούνται για να αναπαραστήσουν τις προσφορές/bids που κάνουν για κάθε δημοπρασία οι EVC aggregators και έχουν την παρακάτω μορφή:

Datetime timestamp without time zone	aggregator double precision	cs double precision	energy double precision	bid bigint
2019-01-01 00:15:00	1	2	71	187
2019-01-01 00:15:00	4	1	71	8
2019-01-01 00:15:00	3	2	19.36	267
2019-01-01 00:15:00	2	2	112	230
2019-01-01 00:15:00	3	3	90.36	387
2019-01-01 00:15:00	3	1	38.72	251
2019-01-01 00:15:00	4	3	30	67

Για κάθε timestamp υπάρχουν προσφορές από όλους τους EVC aggregator σε όλους ή μερικούς σταθμούς. Για να δημιουργηθούν τα δεδομένα, τα δεδομένα φιλτραρίστηκαν έτσι ώστε να εξετάσουμε μόνο τα timestamps, όπου υπήρχε ένα έστω token δημοπρατούμενο. Στη συνέχεια αθροίσαμε την διαθέσιμη ισχύ ανά σταθμό και ανά aggregator. Για τις προσφορές που κάνει ο κάθε aggregator, βάζουμε ένα τυχαίο αριθμό που κυμαίνεται από 1 μέχρι τα διαθέσιμα token που δημοπρατούνται.

Ωστόσο, βρέθηκαν ακόμα αρκετές χρονικές στιγμές για τις οποίες τα δεδομένα δεν ήταν αρκετά, ώστε οι διάφοροι αλγόριθμοι, τα αποτελέσματα των οποίων θα παρουσιαστούν στο επόμενο κεφάλαιο, να πετυχαίνουν διαφορετικά αποτελέσματα. Πιο αναλυτικά, υπάρχουν αρκετά χρονικά τέταρτα στα οποία οι προσφορές είναι λίγες (<10), με αποτέλεσμα όλοι οι αλγόριθμοι να καταλήγουν σε ίδια συμπεράσματα. Για το λόγο αυτό, δημιουργήθηκε και μια συνάρτηση, η οποία δημιουργεί τυχαίες προσφορές για τα χρονικά τέταρτα στα οποία εντοπίζονται λίγες προσφορές.

ΚΕΦΑΛΑΙΟ 5
ΑΠΟΤΕΛΕΣΜΑΤΑ ΑΛΓΟΡΙΘΜΩΝ

5. Αποτελέσματα αλγορίθμων

Στο κεφάλαιο 5, θα παρουσιαστούν τα αποτελέσματα των αλγορίθμων που προέκυψαν από τη λύση όλων των δημοπρασιών στη διάρκεια ενός χρόνου. Ο κάθε αλγόριθμος αποφασίζει ποιες προσφορές γίνονται αποδεκτές στην κάθε δημοπρασία ξεχωριστά. Όπως και στο προηγούμενο κεφάλαιο, χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python και η βάση δεδομένων PostgreSQL.

5.1 Dynamic Programming (DP)

Ο δυναμικός προγραμματισμός, όπως προαναφέρθηκε στο 3.3.1, είναι μια ισχυρή μαθηματική τεχνική, η οποία διασπά ένα μεγάλο πρόβλημα σε απλά υποπροβλήματα ικανά να λυθούν και, έχοντας υπόψιν του όλες τις λύσεις των υποπροβλημάτων, καταλήγει στη βέλτιστη λύση.

Σχετικά με την περίπτωση που εξετάζουμε, θα μπορούσαμε να παρομοιάσουμε τη λύση μιας δημοπρασίας με το γνωστό πρόβλημα του σακιδίου (knapsack problem). Το πρόβλημα του σακιδίου είναι ένα ευρέως γνωστό πρόβλημα, στο οποίο το ζητούμενο είναι ο προσδιορισμός του αριθμού του κάθε στοιχείου που μπορεί να συμπεριληφθεί σε μια συλλογή, έτσι ώστε το συνολικό βάρος των στοιχείων να είναι μικρότερο από ένα δεδομένο όριο και η συνολική του αξία να είναι όσο το δυνατόν μεγαλύτερη. Το όνομα του προέρχεται από το ότι, δεδομένου ενός σακιδίου που έχει συγκεκριμένη χωρητικότητα, ποια αντικείμενα μπορούν να τοποθετηθούν μέσα, ώστε να μεγιστοποιηθεί η αξία του. Πως μπορεί, όμως, να παραλληλιστεί η λύση μιας δημοπρασίας με το πρόβλημα του σακιδίου;



Αν αναλογιστούμε, ότι κάθε δημοπρασία είναι ένας «σάκος», και συγκεκριμένα, χωρητικότητας X , όπου X η δημοπρατούμενη ενέργεια από το διαχειριστή του δικτύου, και τα «αντικείμενα», με τα οποία θέλουμε να γεμίσουμε το «σάκο», είναι οι προσφορές ενέργειας που κάνουν οι διάφοροι διαχειριστές ηλεκτρικών αυτοκινήτων, τότε θα μπορούσε να χρησιμοποιηθεί ο δυναμικός προγραμματισμός, όπως χρησιμοποιείται για την λύση του προβλήματος του σακιδίου.

Παρακάτω αναλύονται 2 υλοποιήσεις του αλγόριθμου, όπου στη μία δεν μας ενδιαφέρει ο αριθμός των tokens που ζητάει ο κάθε παραγωγός και μία στην οποία μας ενδιαφέρει.

5.1.1 Επίλυση με χρήση του αλγορίθμου knapsack – Dynamic Programming (DP)

Στο συγκεκριμένο αλγόριθμο ζητούμενο είναι η εύρεση των βέλτιστων προσφορών που θα ικανοποιήσουν τη ζήτηση ενέργειας στο μέγιστο, αδιαφορώντας για το πόσα tokens θα χρειαστεί να διαθέσει ο διαχειριστής του δικτύου. Στις σημερινές αγορές ενέργειας, υπάρχουν περιπτώσεις, όπου σε κρίσιμες καταστάσεις, όπως φυσικές καταστροφές ή η καταστροφή σημαντικού τμήματος του δικτύου, ο διαχειριστής «αδιαφορεί» για το κόστος της εξισορρόπησης του δικτύου, προκειμένου να εξασφαλίσει την παροχή ενέργειας σε κρίσιμες υποδομές ζωτικής σημασίας, όπως είναι για παράδειγμα ένα νοσοκομείο. Επομένως, η εξέταση αυτής της περίπτωσης δεν είναι άτοπη.

Στα πλαίσια αυτού του αλγορίθμου, δημιουργείται ένας διδιάστατος πίνακας (dp) μεγέθους $(n+1) \times (\text{energy_demand} + 1)$, όπου n είναι ο αριθμός των προσφορών και η ζητούμενη ενέργεια μετατρέπεται σε ακέραιο αριθμό πάντα πολλαπλασιάζοντας την με 100. Στη γραμμή i του πίνακα dp και στη στήλη w αποθηκεύεται η μέγιστη ενέργεια που μπορεί να γίνει αποδεκτή από τις πρώτες i προσφορές, έχοντας w υπολειπόμενη ενέργεια. Αν η ποσότητα της επόμενης προσφοράς ($i + 1$) είναι μεγαλύτερη από την ενέργεια w , τότε η τιμή του $\text{dp}[i+1][w]$ παραμένει ίδια με την τιμή $\text{dp}[i][w]$. Η διαδικασία αυτή ακολουθείται για όλους τους πιθανούς συνδυασμούς και αποθηκεύονται οι αντίστοιχες τιμές στον πίνακα dp.

```

n = len(self.offers)
energy_demand = int(self.energy_demand * 100)

dp = [[0] * (energy_demand + 1) for _ in range(n + 1)]

for i in range(1, n + 1):
    energy, _ = self.offers[i - 1]
    energy_in_cents = int(energy * 100)

    for w in range(1, energy_demand + 1):
        if energy_in_cents <= w:
            dp[i][w] = max(dp[i - 1][w], dp[i - 1][w - energy_in_cents] + energy_in_cents)
        else:
            dp[i][w] = dp[i - 1][w]

```

Στη συνέχεια, αφού συμπληρωθεί ο πίνακας dp, ο αλγόριθμος χρησιμοποιεί μια τεχνική για τον εντοπισμό της βέλτιστης λύσης, σύμφωνα με την οποία, ξεκινώντας από το τελευταίο κελί του πίνακα, $\text{dp}[n][\text{energy_demand}]$, διατρέχει όλο τον πίνακα και βρίσκει το βέλτιστο συνδυασμό προσφορών και το επιστρέφει:

```

best_combination = []

w = energy_demand
for i in range(n, 0, -1):
    energy, _ = self.offers[i - 1]
    energy_in_cents = int(energy * 100) # Convert float to integer for table indexing

    if dp[i][w] != dp[i - 1][w]:
        best_combination.append(energy)
        w -= energy_in_cents
best_combination = [offer for offer in best_combination]

return best_combination, dp

```

Παρακάτω απεικονίζεται η λύση που δίνει ο συγκεκριμένος αλγόριθμος στη λύση μιας δημοπρασίας. Πιο συγκεκριμένα, παρουσιάζονται οι προσφορές μιας δημοπρασίας, ο

καλύτερος συνδυασμός των αποδεκτών προσφορών που προτείνει ο αλγόριθμος, το άθροισμα της ενέργειας που καλύπτουν οι αποδεκτές προσφορές και το συνολικό κόστος των αποδεκτών προσφορών.

2019-01-01 00:00:00

Bids: [(48.54, 376), (999.91, 76), (88.18, 41), (1361.38, 308), (1161.37, 502), (787.26, 449), (1091.83, 456), (152.66, 115), (358.42, 435), (375.85, 519)]

Best Combination of Offers (energy): [358.42, 787.26, 999.91]

Total Energy: 2145.59 Demand: 2147.0

Total Price: 960 Available: 531

Παρατηρούμε ότι η ενέργεια καλύπτεται σχεδόν ολόκληρη (99.93%) σε αυτή τη δημοπρασία, αλλά το κόστος αυτός της λύσης είναι πιο ακριβό από τα διαθέσιμα tokens που, αρχικά, είχε υπολογίσει ο διαχειριστής του δικτύου να προσφέρει.

5.1.2 Επίλυση με χρήση του αλγορίθμου knapsack με βάρη - Backtracking (BR)

Παρόμοια με τον προηγούμενο αλγόριθμο, το ζητούμενο είναι η εύρεση των βέλτιστων προσφορών που θα ικανοποιήσουν τη ζήτηση ενέργειας στο μέγιστο, λαμβάνοντας ωστόσο υπόψιν και τα tokens, τα οποία ο διαχειριστής είναι διατεθειμένος να προσφέρει συνολικά. Ουσιαστικά, η αντικειμενική συνάρτηση που καλείται να λύσει ο αλγόριθμος είναι αυτή που παρουσιάζεται στο κεφάλαιο 3.3. Την ίδια αντικειμενική συνάρτηση θα κληθούν να λύσουν και όλοι οι υπόλοιποι εξεταζόμενοι αλγόριθμοι.

Παρά το γεγονός ότι πρόκειται για τον ίδιο αλγόριθμο με αυτόν του 5.1.1 έχοντας έναν επιπλέον περιορισμό, τα διαθέσιμα tokens της κάθε δημοπρασίας, προσεγγίζεται με ένα διαφορετικό τρόπο για να εξεταστεί και η απόδοση των 2 διαφορετικών αλγορίθμων. Χρησιμοποιείται μια backtracking τεχνική με αναδρομικό τρόπο. Παρόλο που είναι διαφορετική από το δυναμικό προγραμματισμό, έχει παρόμοια λογική. Ενώ ο δυναμικός προγραμματισμός προσπαθεί να βελτιστοποιήσει ένα πρόβλημα λύνοντας επικαλυπτόμενα υποπροβλήματα, το backtracking χρησιμοποιεί μια brute-force προσέγγιση, λύνοντας όλους τους πιθανούς συνδυασμούς.

Πιο συγκεκριμένα, χρησιμοποιώντας αναδρομή διατρέχει όλους τους πιθανούς συνδυασμούς και χωρίς να τους αποθηκεύει σε κάποια μεταβλητή. Αντιθέτως:

- Ελέγχει αν το άθροισμα των ενεργειών του εξεταζόμενου συνδυασμού προσφορών και το άθροισμα των τιμών είναι μεγαλύτερα από την ενέργεια και την τιμή που διαθέτει η εκάστοτε δημοπρασία, αντίστοιχα. Αν ναι, την απορρίπτει και πάει στην επόμενη.
- Ελέγχει αν το άθροισμα των ενεργειών του εξεταζόμενου συνδυασμού προσφορών είναι μεγαλύτερο από το υπάρχον άθροισμα του καλύτερου συνδυασμού. Αν ναι, θέτει αυτό τον συνδυασμό ως το καλύτερο.
- Τέλος, διατρέχει όλες τις προσφορές που δεν έχουν χρησιμοποιηθεί. Αν δεν ξεπερνάει το όριο της ενέργειας της εκάστοτε δημοπρασίας, την προσθέτει στο συνδυασμό και καλεί τη συνάρτηση αναδρομικά. Στο τέλος επιστρέφεται ο καλύτερος συνδυασμός.

```
def knapsack_backtracking(self):
    def backtrack(start, curr_energy, curr_price, combination):
        nonlocal max_energy, best_combination

        if curr_energy > self.energy_demand or curr_price > price_limit:
            return

        if curr_energy > max_energy:
            max_energy = curr_energy
            best_combination = combination.copy()

        for i in range(start, len(self.offers)):
            energy, price = self.offers[i]
            if not used[i] and (
                curr_energy + energy) <= self.energy_demand:
                used[i] = True
                combination.append((energy, price))
                backtrack(i + 1, curr_energy + energy, curr_price + price, combination)
                combination.pop()
                used[i] = False

    n = len(self.offers)
    price_limit = int(self.price_limit)

    max_energy = 0
    best_combination = []
    combination = []
    used = [False] * n

    backtrack(0, 0, 0, combination)

    return best_combination
```

Παρακάτω απεικονίζεται η λύση που δίνει ο συγκεκριμένος αλγόριθμος στη λύση μιας δημοπρασίας. Πιο συγκεκριμένα, παρουσιάζονται οι προσφορές μιας δημοπρασίας, ο

καλύτερος συνδυασμός των αποδεκτών προσφορών που προτείνει ο αλγόριθμος, το άθροισμα της ενέργειας που καλύπτουν οι αποδεκτές προσφορές και το συνολικό κόστος των αποδεκτών προσφορών.

```
2019-01-01 00:00:00
Bids: [(1470.5, 460), (210.47, 90), (872.67, 509), (571.88, 268), (1302.43, 53), (1053.94, 12), (459.05, 271), (269.7, 267), (1171.69, 161), (893.57, 507)]
Knapsack Backtracking
Best Combination of Offers (energy): [210.47, 571.88, 1302.43]
Total Energy: 2084.78 Demand: 2147.0
Total Price: 411 Available: 531
```

Παρατηρούμε ότι η ενέργεια καλύπτεται σχεδόν ολόκληρη (97%) σε αυτή τη δημοπρασία, αλλά σε μικρότερο βαθμό από τον προηγούμενο αλγόριθμο. Ωστόσο, η λύση δεν ξεπερνάει το κόστος που διατίθεται ο διαχειριστής του δικτύου να προσφέρει για την απορρόφηση της ενέργειας.

5.2 Mixed-integer Linear Programming (MILP)

Για την υλοποίηση του αλγόριθμου MILP χρησιμοποιήθηκε η βιβλιοθήκη PuLP της Python. Η βιβλιοθήκη PuLP είναι μια ευρέως χρησιμοποιούμενη βιβλιοθήκη της Python, ανοιχτού κώδικα, που παρέχει μια φιλική προς το χρήστη διεπαφή για τη διαμόρφωση και την επίλυση προβλημάτων γραμμικού προγραμματισμού (LP) και γραμμικού προγραμματισμού μικτού - ακέραιου (MILP). Η δυνατότητα της PuLP να εκφραστούν αντικειμενικές συναρτήσεις και γραμμικοί περιορισμοί με τρόπο που μοιάζει αρκετά με μια μαθηματική έκφραση είναι και ο λόγος που επιλέχθηκε.

Το πρόβλημα βελτιστοποίησής μας περιστρέφεται γύρω από την επιλογή προσφορών για τη μεγιστοποίηση της συνολικής παραγωγής ενέργειας με ταυτόχρονη τήρηση των περιορισμών. Ορίζουμε δυαδικές μεταβλητές απόφασης για να υποδείξουμε την κατάσταση επιλογής κάθε προσφοράς. Ο στόχος είναι η μεγιστοποίηση της συνολικής παραγόμενης ενέργειας, που υπολογίζεται ως το άθροισμα των ενεργειακών εκροών των επιλεγμένων προσφορών.

Στην παρακάτω εικόνα, απεικονίζεται ο κώδικας που ορίζει την αντικειμενική μας συνάρτηση:

```
model = pulp.LpProblem("Bid_Selection_Optimization", pulp.LpMaximize)

x = [pulp.LpVariable(f"x_{i}", cat=pulp.LpBinary) for i in range(len(bids))]

model += pulp.lpSum([x[i] * bid[0] for i, bid in enumerate(bids)]), "Total_Energy_Generated_Objective"
```

Στη συνέχεια, προσθέτουμε τους 2 περιορισμούς του προβλήματος μας, δηλαδή η συνολική ενέργεια και το συνολικό κόστος των αποδεκτών προσφορών να είναι μικρότερα ή ίσα από την εκάστοτε δημοπρατούμενη ενέργεια και τα δημοπρατούμενα tokens, αντίστοιχα.

```
model += pulp.lpSum([x[i] * bid[0] for i, bid in enumerate(bids)]) <= demand, "Energy_Demand_Constraint"
model += pulp.lpSum([x[i] * bid[1] for i, bid in enumerate(bids)]) <= available_tokens, "Budget_Constraint"
```

Τέλος, λύνουμε το πρόβλημα, βρίσκουμε την καλύτερη λύση και επιστρέφουμε τα αποτελέσματα της συνάρτησης.

```
# Solve the MILP problem
model.solve()

## Display the results
print("Status:", pulp.LpStatus[model.status])
print("Objective Value (Total Energy Generated):", pulp.value(model.objective))

# Display the selected bids
selected_bids = []
total_energy = 0
total_cost = 0
for i, bid in enumerate(bids):
    if pulp.value(x[i]) > 0:
        selected_bids.append((bid[0] * pulp.value(x[i]), bid[1] * pulp.value(x[i])))
        total_energy += bid[0] * pulp.value(x[i])
        total_cost += bid[1] * pulp.value(x[i])
        print(f"Bid {i + 1}: Energy={bid[0]}, Cost={bid[1]}, Selected={pulp.value(x[i])}")

return total_energy, total_cost, selected_bids
```

Παρακάτω απεικονίζεται η λύση που δίνει ο συγκεκριμένος αλγόριθμος στη λύση μιας δημοπρασίας. Πιο συγκεκριμένα, παρουσιάζονται οι προσφορές μιας δημοπρασίας, ποιες από τις προσφορές έγιναν αποδεκτές, δηλαδή τον καλύτερο συνδυασμό αποδεκτών προσφορών που προτείνει ο αλγόριθμος, το άθροισμα της ενέργειας που καλύπτουν οι αποδεκτές προσφορές και το συνολικό κόστος των αποδεκτών προσφορών.

```
2019-01-01 00:00:00
Bids: [(111.31, 328), (961.24, 67), (367.67, 443), (636.99, 226), (1214.4, 9), (17.37, 18), (1430.29, 347), (457.61, 98), (643.7, 228), (355.79, 195)]
Status: Optimal
Objective Value (Total Energy Generated): 2879.92
Bid 2: Energy=961.24, Cost=67, Selected=1.0
Bid 6: Energy=17.37, Cost=18, Selected=1.0
Bid 8: Energy=457.61, Cost=98, Selected=1.0
Bid 9: Energy=643.7, Cost=228, Selected=1.0
MILP
Best Combination of Offers (energy): [(961.24, 67.0), (17.37, 18.0), (457.61, 98.0), (643.7, 228.0)]
Total Energy: 2879.92 Demand: 2147.0
Total Price: 411.0 Available: 531
```

Στο συγκεκριμένο παράδειγμα, παρατηρούμε ότι ικανοποιούνται και οι 2 συνθήκες και πλησιάζει αρκετά κοντά στην κάλυψη όλης της ενέργειας.

5.3 Particle Swarm Optimization (PSO)

Όπως και στους 2 προηγούμενους αλγόριθμους που παρουσιάστηκαν (5.2, 5.1.2), η αντικειμενική συνάρτηση και οι περιορισμοί είναι ίδιοι. Προστίθεται, όμως, στο PSO και η έννοια του CAPACITY της κάθε μονάδας. Με τον όρο CAPACITY εννοούμε την τεχνική δυνατότητα της κάθε μονάδας, δηλαδή του κάθε συνόλου από ηλεκτρικούς φορτιστές, για το πόση ενέργεια μπορεί να προσφέρει για μια δημοπρασία.

Έχει νόημα, λοιπόν, να γίνει ένας ακόμη έλεγχος για το, εάν η ποσότητα ενέργειας μιας προσφοράς που έχει κάνει ένας διαχειριστής στόλου ηλεκτρικών οχημάτων για μια συγκεκριμένη μονάδα, μπορεί να υποστηριχθεί. Σε περίπτωση που η ποσότητα αυτή είναι μεγαλύτερη από το CAPACITY της εκάστοτε μονάδας, τότε η ποσότητα της ενέργειας που μπορεί να γίνει αποδεκτή είναι ίση με το μέγιστο CAPACITY της μονάδας.

Έτσι, όλοι οι πιθανοί συνδυασμοί προσφορών θα πρέπει να λαμβάνουν υπόψιν τους και για ποιο φορτιστή έχει γίνει η κάθε προσφορά. Για παράδειγμα:

- Έστω ότι έχουμε δύο φορτιστές, A και B, και τρεις προσφορές από διαχειριστές στόλων ηλεκτρικών οχημάτων, α_1 , α_2 και β_1 . Οι δύο πρώτες προσφορές, α_1 και α_2 , αφορούν το φορτιστή A και η τελευταία προσφορά, β_1 , το φορτιστή B. Οι πιθανές λύσεις του αλγόριθμου τότε που πρέπει να εξεταστούν είναι οι παρακάτω:

- $[\alpha_1, \beta_1]$

- $[\alpha_2, \beta_1]$

με την προϋπόθεση $\alpha_1, \alpha_2 \leq \text{CAPACITY}(A)$ και $\beta_1 \leq \text{CAPACITY}(B)$.

Χρησιμοποιώντας τη συνάρτηση `product` της βιβλιοθήκης `itertools` της Python δημιουργούνται όλοι οι πιθανοί συνδυασμοί. Για τον κάθε συνδυασμό καλείται ο αλγόριθμος PSO και επιστρέφεται η καλύτερη λύση για τον αντίστοιχο συνδυασμό.

Συγκρίνουμε την λύση t με την καλύτερη προ υπάρχουσα λύση t' και, αν η λύση t είναι καλύτερη από την λύση t' , τότε ανανεώνουμε τις αντίστοιχες τιμές. Παρακάτω απεικονίζεται ο κώδικας:

```
# For every combination of offers
for combo_idx in range(len(combos_energies)):
    energy_offers = combos_energies[combo_idx]
    bid_prices = combos_prices[combo_idx]
    # Run PSO algorithm and get the optimal solution for the combination
    best_solution, _ = particle_swarm_optimization(energy_offers, bid_prices, demand, available_tokens)
    best_cost = 0
    best_energy = 0
    # Compare if the returned solution is the best and update the corresponding variables
    for i in range(len(best_solution)):
        best_cost += best_solution[i] * bid_prices[i]
        best_energy += best_solution[i] * energy_offers[i]
    if best_energy > max_energy and best_energy <= demand:
        max_energy = best_energy
        best_solution_of_all = [(best_solution[i] * energy_offers[i], best_solution[i] * bid_prices[i]) for i
                               in range(len(best_solution))]
        best_energy_offer = energy_offers
        best_price_offer = bid_prices
        min_cost = best_cost
    elif best_energy == max_energy and best_energy <= demand:
        if best_cost < min_cost:
            max_energy = best_energy
            best_solution_of_all = [(best_solution[i] * energy_offers[i], best_solution[i] * bid_prices[i]) for i
                                   in range(len(best_solution))]
            best_energy_offer = energy_offers
            best_price_offer = bid_prices
            min_cost = best_cost
```

Στη συνέχεια, αναλύονται συνοπτικά όλες οι απαραίτητες συναρτήσεις που χρειάζονται για την ανάπτυξη του αλγορίθμου. Πιο συγκεκριμένα:

- `particle_swarm_optimization`: Αυτή είναι η κύρια συνάρτηση του αλγορίθμου. Είναι υπεύθυνη για την αρχικοποίηση των ταχυτήτων και των σωματιδίων. Για κάθε σωματίδιο, ελέγχει την καταλληλότητά του, ενημερώνει τη θέση και τη ταχύτητα του. Εν τέλει, επιστρέφει την καλύτερη «παγκόσμια» λύση.

```

# PSO algorithm
def particle_swarm_optimization(energies, bids, energy_demand, available_tokens):
    particles = [[random.choice([0, 1]) for _ in range(NUM_UNITS)] for _ in range(NUM_PARTICLES)]
    velocities = [[random.uniform(-1, 1) for _ in range(NUM_UNITS)] for _ in range(NUM_PARTICLES)]
    best_global = [0, 0, 0]
    best_global_cost = float('inf')

    for _ in range(MAX_ITERATIONS):
        for i, particle in enumerate(particles):
            cost = fitness_function(particle, energies, bids, energy_demand, available_tokens)
            if cost < best_global_cost:
                best_global = particle.copy()
                best_global_cost = cost

        best_particle = particles[i].copy()
        update_velocity(particle, velocities[i], best_particle, best_global, energies, energy_demand)
        update_position(particle, velocities[i])

    return best_global, best_global_cost

```

- `fitness_functions`: Αυτή η συνάρτηση εξετάζει την καταλληλότητα ενός σωματιδίου. Ένα σωματίδιο είναι μια αλληλουχία από 0 και 1, όπου 0 σημαίνει ότι η μονάδα δε χρησιμοποιείται και 1 ότι χρησιμοποιείται. Αν μια μονάδα χρησιμοποιείται, γίνεται ο έλεγχος αν η ενέργεια της προσφοράς είναι μεγαλύτερη από το `CAPACITY` της μονάδας και, αν ναι, κάνει την αντίστοιχη τροποποίηση. Επίσης, ελέγχει:
 - Αν όλες οι μονάδες είναι κλειστές. Αν ναι, τότε θέτει το κόστος ίσο με άπειρο.
 - Αν η συνολική ενέργεια του σωματιδίου είναι μεγαλύτερη από τη δημοπρατούμενη ενέργεια. Αν ναι, προσθέτει ένα πέναλτυ στο κόστος της λύσης ίσο με τη μέγιστη τιμή των προσφορών επί δέκα.
 - Αν το κόστος της λύσης είναι μεγαλύτερο από τα δημοπρατούμενα tokens. Αν ναι, τότε θέτει το κόστος ίσο με άπειρο.
 - Αν η συνολική ενέργεια του σωματιδίου είναι χαμηλότερη από τη δημοπρατούμενη ενέργεια. Αν ναι, τότε προσθέτει ένα πέναλτυ στο κόστος της λύσης ίσο με τη μέγιστη τιμή των προσφορών.

```

def fitness_function(particle, energy_offers, bid_prices, energy_demand, available_tokens):
    total_cost = 0
    total_energy = 0
    all_units_off = True

    for i in range(len(particle)):
        if particle[i] == 1: # If the unit is ON
            all_units_off = False
            if energy_offers[i] > MAX_CAPACITY[i]:
                total_energy += MAX_CAPACITY[i]
            else:
                total_energy += energy_offers[i]
            total_cost += bid_prices[i]

    if all_units_off:
        total_cost = float('inf')

    # Ensure total_energy does not exceed energy_demand
    if total_energy > energy_demand:
        total_cost += (total_energy - energy_demand) * max(bid_prices) * 10

    # Ensure total_energy does not exceed energy_demand
    if total_cost > available_tokens:
        total_cost = float('inf')

    # Penalty for not meeting the demand
    if total_energy < energy_demand:
        total_cost += (energy_demand - total_energy) * max(bid_prices)
    return total_cost

```

- `update_velocity`: Αυτή η συνάρτηση ενημερώνει την ταχύτητα ενός σωματιδίου. Ουσιαστικά, υπολογίζει τη συνάρτηση που παρουσιάζεται στο 3.3.3. Μετά τον υπολογισμό των ταχυτήτων, διασφαλίζει ότι η ζήτηση της ενέργειας καλύπτεται από το συγκεκριμένο σωματίδιο.

```

def update_velocity(particle, velocity, best_particle, best_global, energy_offers, energy_demand):
    for i in range(len(particle)):
        r1, r2 = random.random(), random.random()
        velocity[i] = (INERTIA_WEIGHT * velocity[i]
                      + COGNITIVE_WEIGHT * r1 * (best_particle[i] - particle[i])
                      + SOCIAL_WEIGHT * r2 * (best_global[i] - particle[i]))

    total_energy = sum(particle[i] * energy_offers[i] for i in range(len(particle)))
    if total_energy < energy_demand:
        deficit = energy_demand - total_energy
        for i in range(len(particle)):
            if particle[i] == 0 and deficit >= energy_offers[i]:
                particle[i] = 1
                deficit -= energy_offers[i]
            elif particle[i] == 0 and deficit > 0:
                particle[i] = 1
                break

```

Παρακάτω απεικονίζεται η λύση που δίνει ο συγκεκριμένος αλγόριθμος στη λύση μιας δημοπρασίας. Πιο συγκεκριμένα, παρουσιάζονται οι προσφορές μιας δημοπρασίας, ποιες από τις προσφορές έγιναν αποδεκτές, δηλαδή τον καλύτερο συνδυασμό αποδεκτών προσφορών που προτείνει ο αλγόριθμος και ποιες αποφασίζει να ενεργοποιήσει (0/1), το άθροισμα της ενέργειας που καλύπτουν οι αποδεκτές προσφορές και το συνολικό κόστος των αποδεκτών προσφορών.

```
PSO
Energy demand 2147.0
Available tokens 531
Energy offers: [(1046.6, 147), (1131.48, 463), (99.55, 280), (388.02, 175), (1190.4, 276), (476.64, 292), (693.5, 207), (1360.15, 190), (955.18, 487), (696.68, 463)]
Best solution (Unit Commitment Schedule): [(693.5, 207), (1360.15, 190), (0.0, 0)]
Best energy: 2053.65
Best cost: 397
Best combination of offers(energy): (693.5, 1360.15, 99.55)
Best combination of offers(price): (207, 190, 280)
```

Στη συγκεκριμένη δημοπρασία, παρατηρούμε ότι ικανοποιούνται και οι 2 συνθήκες και πλησιάζει αρκετά κοντά στην κάλυψη όλης της ενέργειας. Παρόλο που πετυχαίνει χειρότερα αποτελέσματα σε σύγκριση με τους προηγούμενους αλγόριθμους, παρατηρείται ότι πετυχαίνει την πιο οικονομική λύση. Ξοδεύοντας, δηλαδή ~130 tokens λιγότερα, χάνει μόνο ~70 kWh σε σχέση με τους άλλους αλγόριθμους.

5.4 Lagrangian Relaxation (LR)

Ο τελευταίος αλγόριθμος που παρουσιάζεται είναι αυτός της μεθόδου Lagrangian Relaxation. Η ιδιαιτερότητα αυτή της μεθόδου έγκειται στο γεγονός ότι μπορεί να γίνει αποδεκτό ένα ποσοστιαίο μέρος μιας προσφοράς και όχι ολόκληρη η προσφορά. Το γεγονός αυτό είναι ενδιαφέρον, αν αναλογιστούμε ότι σε πολλές χρηματιστηριακές αγορές, οι παίκτες έχουν το δικαίωμα να κάνουν διάφορους τύπους προσφορών. Πιο συγκεκριμένα, ένας παίκτης που θέλει να συμμετέχει σε μια αγορά, για παράδειγμα στην αγορά ηλεκτρικής ενέργειας, μπορεί να βάλει διάφορους περιορισμούς στην προσφορά του, όπως, να γίνει ολόκληρη η προσφορά δεκτή, αλλιώς να μη γίνει καθόλου αποδεκτή, να μπορεί να γίνει μέρος αυτής της προσφοράς αποδεκτό, να μπορεί να γίνει μια προσφορά αποδεκτή, εφόσον και μόνο, αν μια προηγούμενη προσφορά του γίνει αποδεκτή κοκ.

Για την υλοποίηση αυτού του αλγορίθμου χρησιμοποιήθηκε η βιβλιοθήκη `scipy` της Python. Αφού αρχικοποιηθούν οι απαραίτητες μεταβλητές για τον αλγόριθμο, όπως ο αριθμός των επαναλήψεων και οι τιμές των πολλαπλασιαστών Lagrange, διατρέχουμε το κύριο βρόγχο. Σε κάθε επανάληψη γίνονται τα παρακάτω βήματα:

- Ορίζεται μια συνάρτηση `relaxed_problem`, σκοπός της οποίας είναι να δημιουργήσει τη λύση στο «χαλαρό» πρόβλημα. Οι υπολογισμοί λαμβάνουν υπόψιν τη συνολική ενέργεια και κόστος των εξεταζόμενων προσφορών, καθώς και το πολλαπλασιαστή Lagrange `lambda_budget`.
- Ορίζει τους 2 περιορισμούς της δημοπρασίας, δηλαδή το άθροισμα των ενεργειών των προσφορών και το άθροισμα των τιμών των προσφορών να μην ξεπερνάνε τη δημοπρατούμενη ενέργεια και τα δημοπρατούμενα `tokens`, αντίστοιχα.
- Ορίζει το διάστημα 0 έως 1 στο οποίο μια προσφορά μπορεί να γίνει αποδεκτή. Αυτό είναι απαραίτητο, ώστε να η κάθε προσφορά να γίνεται αποδεκτή το πολύ μια φορά.
- Λύνεται το «χαλαρό» πρόβλημα χρησιμοποιώντας τη συνάρτηση `minimize` της βιβλιοθήκης `scipy`.
- Προσαρμόζεται σύμφωνα με τη λύση που βρέθηκε στο προηγούμενο βήμα η τιμή του πολλαπλασιαστή Lagrange και υπολογίζεται η παραβίαση του περιορισμού κόστους. Αν αυτή η παραβίαση είναι μικρότερη από ένα όριο, ο βρόγχος διακόπτεται.

```
def lagrangian_relaxation(bids, demand, available_tokens):

    all_bids = [bid for bid in bids]
    num_bids = len(all_bids)

    max_iterations = 100
    convergence_threshold = 1e-6

    lambda_budget = 0.0

    for iteration in range(max_iterations):
        def relaxed_problem(x):
            total_energy = sum(x[i] * all_bids[i][0] for i in range(num_bids))
            total_cost = sum(x[i] * all_bids[i][1] for i in range(num_bids))
            return -total_energy + lambda_budget * (total_cost - available_tokens)

        constraints = ({'type': 'eq', 'fun': lambda x: sum(x[i] * all_bids[i][0] for i in range(num_bids)) - demand},
                      {'type': 'ineq',
                       'fun': lambda x: sum(x[i] * all_bids[i][1] for i in range(num_bids)) - available_tokens})

        bounds = [(0, 1) for _ in range(num_bids)]

        result = minimize(relaxed_problem, np.zeros(num_bids), constraints=constraints, bounds=bounds)
        subproblem_solution = result.x

        violation_budget = sum(
            subproblem_solution[i] * all_bids[i][1] for i in range(num_bids)) - available_tokens
        lambda_budget += violation_budget

    if abs(violation_budget) < convergence_threshold:
        break
```

Παρακάτω απεικονίζεται η λύση που δίνει ο συγκεκριμένος αλγόριθμος στη λύση μιας δημοπρασίας. Πιο συγκεκριμένα, παρουσιάζονται ο αριθμός των επαναλήψεων, η τιμή του πολλαπλασιαστή Lagrange, τα ποσοστά, στα οποία γίνεται αποδεκτή η κάθε προσφορά, η συνολική ενέργεια της προτεινόμενης λύσης και το συνολικό κόστος της προτεινόμενης λύσης.

```
Lagrangian Relaxation
Iteration: 100
Lagrange Multiplier (Budget): 19.27425063744039
Subproblem Solution: [0.37651289 0.24366217 0.07493812 0.36985388 0.05050929
0.19062344 0.49000301 0.15408816 0.05449303]
Total energy generated: 2147.005 / 2147.0
Total cost: 530.94 / 531
```

Παρατηρούμε ότι το άθροισμα της αποδεκτής ενέργειας είναι μεγαλύτερο από την ζητούμενη ενέργεια κατά 0.005 kWh. Αυτό συμβαίνει, διότι οι 100 επαναλήψεις του βρόγχου, δεν ήταν αρκετές για τη συγκεκριμένη δημοπρασία. Θεωρητικά, αν είχαμε περισσότερες επαναλήψεις μια καλύτερη λύση θα είχε βρεθεί. Ωστόσο, ο αριθμός των 100 επαναλήψεων έχει χρησιμοποιηθεί, ώστε να είναι αρκετός για να βρίσκει λύση στο μεγαλύτερο μέρος των δημοπρασιών και ο αλγόριθμος να μπορεί να τρέξει ικανοποιητικά σε πραγματικές συνθήκες.

ΚΕΦΑΛΑΙΟ 6
ΣΥΜΠΕΡΑΣΜΑΤΑ

6. Συμπεράσματα

Έχοντας αναλύσει τα δεδομένα, τη δημιουργία τους και το πως οι αλγόριθμοι εφαρμόζονται, σε αυτό το κεφάλαιο, θα παρουσιαστούν τα συμπεράσματα που προκύπτουν από τα αποτελέσματα των αλγορίθμων και θα απαντηθούν τα ερωτήματα που τέθηκαν στο κεφάλαιο 3.1.

6.1 Σύγκριση επιδόσεων των αλγορίθμων σχετικά με την αθροιστική παραγωγή και τα κόστη των λύσεων

Αρχικά, εξετάζουμε το πόσες φορές μπόρεσε ο κάθε αλγόριθμος να λύσει μια δημοπρασία. Συγκεκριμένα:

- Αν μπόρεσε να βρει μια λύση, δηλαδή να γίνει αποδεκτή, έστω και μια προσφορά (Found Solution)
- Αν η λύση που βρήκε είχε συνολική ενέργεια μεγαλύτερη από τη ζητούμενη (Accepted Energy > Energy Demand)
- Αν η λύση που βρήκε ήταν μικρότερη ή ίση από τη ζητούμενη ενέργεια (Accepted Energy <= Energy Demand)

Αλγόριθμοι	Found Solution (%)	Accepted Energy > Energy Demand (%)	Accepted Energy <= Energy Demand (%)
<i>Knapsack without weights –Dynamic Programming (DP)</i>	100%	2.70%	97.30%
<i>Knapsack with weights – Backtracking (BR)</i>	100%	0.00%	100.00%
<i>Particle Swarm Optimization (PSO)</i>	100%	0.00%	100.00%
<i>Mixed Integer-Linear Programming (MILP)</i>	100%	0.04%	99.96%
<i>Lagrangian Relaxation – (LR)</i>	92%	18.50%	73.50%

Παρατηρούμε ότι μόνο δύο αλγόριθμοι, οι BR και PSO, βρήκαν στο 100% των δημοπρασιών λύσεις που να τηρούν τον περιορισμό του να γίνονται αποδεκτές προσφορές που το άθροισμα της ενέργειας του να μην είναι μεγαλύτερο από τη δημοπρατούμενη ενέργεια, ενώ πολύ καλά κοντά στο 100% έφτασαν και οι αλγόριθμοι DP και MILP. Τα χειρότερα αποτελέσματα εντοπίζονται στον αλγόριθμο LR, γεγονός αναμενόμενο, καθώς για να βρεθεί αποτέλεσμα σε μια δημοπρασία που δεν κατάφερε να βρει, θα έπρεπε να εκτελέσουμε περισσότερες επαναλήψεις, αυξάνοντας, όμως, αρκετά το χρόνο εκτέλεσης επίλυσης της.

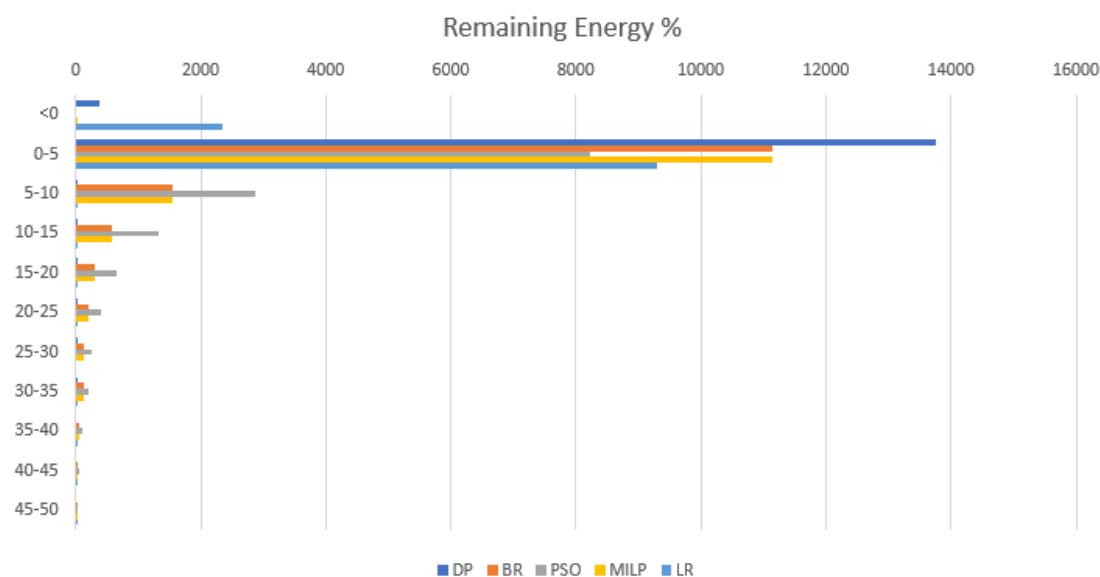
Ωστόσο, ιδιαίτερο ενδιαφέρον έχουν τα συμπεράσματα που προκύπτουν από τον παρακάτω πίνακα, στον οποίο φαίνεται το άθροισμα και το ποσοστό της υπολειπόμενης ενέργειας όλων των δημοπρασιών από τις λύσεις που βρήκαν οι αλγόριθμοι. Να σημειωθεί ότι έχουν αθροιστεί μόνο οι λύσεις που οι αλγόριθμοι έδωσαν και τηρούσαν το περιορισμό της ενέργειας.

Αλγόριθμοι	Remaining Energy (%)
<i>DP</i>	0.27%
<i>BR</i>	4.24%
<i>PSO</i>	4.21%
<i>MILP</i>	4.25%
<i>LR</i>	1.75%

Παρατηρούμε ότι οι δύο αλγόριθμοι, που βρήκαν τις λιγότερες αποδεκτές λύσεις (DP και LR), έχουν το μικρότερο ποσοστό υπολειπόμενης ενέργειας. Αυτό υποδεικνύει ότι, όταν ένας από τους δύο αλγόριθμους καταφέρει να βρει λύση, πιθανότατα θα δώσει και καλύτερη λύση από τους υπόλοιπους. Οι άλλοι αλγόριθμοι βρίσκονται στην ίδια θέση με το PSO να είναι ελάχιστα καλύτερος από τους άλλους δύο κατά 0.03%.

Σχετικά με την απόδοση του κάθε αλγόριθμου στην κάθε δημοπρασία, υπολογίσαμε, αρχικά, το ποσοστό της υπολειπόμενης ενέργειας που απομένει σε κάθε δημοπρασία, δηλαδή αν η δημοπρατούμενη ενέργεια είναι 100kWh και το άθροισμα της ενέργειας της αποδεκτής λύσης είναι 95kWh, τότε έχει επιτευχθεί ένα ποσοστό υπολειπόμενης ενέργειας 5%. Στη συνέχεια, δημιουργήθηκαν διαστήματα (bins) εύρους 5%, με σκοπό να ενταχθεί η κάθε δημοπρασία σε ένα διάστημα. Όσες δημοπρασίες είχαν ποσοστό

μικρότερο του 0%, δηλαδή γινόταν υπερκάλυψη της ενέργειας, τοποθετήθηκαν σε ένα διάστημα (<0%). Παρακάτω, απεικονίζονται τα αποτελέσματα:



Παρατηρούμε ότι, παρόλο που ο αλγόριθμος PSO έχει συνολικό άθροισμα υπολειπόμενης ενέργειας μικρότερο από τους αλγορίθμους MILP και BR, στα διαστήματα έχει χειρότερες επιδόσεις. Συγκεκριμένα, οι δημοπρασίες του PSO που εντάσσονται στο διάστημα 0-5% είναι 8.227, ενώ των άλλων δύο αλγορίθμων είναι 11.142. Ακόμη, σε όλες τις υπόλοιπα διαστήματα ο PSO αλγόριθμος έχει περισσότερες δημοπρασίες. Σύμφωνα λοιπόν με αυτά τα δεδομένα, προκύπτει το συμπέρασμα ότι ο PSO αλγόριθμος βρίσκει καλύτερες λύσεις σε δημοπρασίες που έχουν μεγαλύτερη δημοπρατούμενη ενέργεια και περισσότερες προσφορές, ενώ βρίσκει χειρότερες λύσεις σε δημοπρασίες που η δημοπρατούμενη ενέργεια είναι μικρή και ο ανταγωνισμός φτωχός. Αναμενόμενα είναι τα αποτελέσματα για τους αλγορίθμους LR και DP. Ο DP έχει την καλύτερη απόδοση από όλους τους αλγορίθμους, ενώ ο LR έχει τις περισσότερες δημοπρασίες στις οποίες οι ενέργειες υπερκαλύπτονται.

Στη συνέχεια, θα εξεταστεί και η απόδοση των αλγορίθμων ως προς το κόστος των λύσεων που προτείνουν και ποιος από αυτούς είναι ο πιο οικονομικά συμφέρων. Ο DP αλγόριθμος εξαιτίας της έλλειψης του περιορισμού των δημοπρατούμενων tokens, τον οποίο δε λαμβάνει υπόψιν, αναμένεται να προτείνει και τις πιο «ακριβές» λύσεις.

Αλγόριθμοι	Remaining tokens (%)
<i>DP</i>	-114.68%
<i>BR</i>	20.74%
<i>PSO</i>	23.4%
<i>MILP</i>	20.76%
<i>LR</i>	0.7%

Πράγματι, στον παραπάνω πίνακα, ο αλγόριθμος DP έχει με διαφορά την ακριβότερη λύση. Το αρνητικό ποσοστό υποδεικνύει ότι χρειαζόμασταν αλλά 109.3% tokens από τα ήδη διαθέσιμα για να πληρώσουμε τις αποδεκτές λύσεις. Είναι λογικό, λοιπόν, ο συγκεκριμένος αλγόριθμος να έχει την πιο ακριβή λύση, αλλά και τις καλύτερες επιδόσεις την κάλυψη της ενέργειας, όπως είδαμε προηγουμένως. Οι πιο οικονομικές λύσεις προκύπτουν από τον αλγόριθμο PSO, ο οποίος χρησιμοποιεί μόνο το 76.6% των συνολικών διαθέσιμων tokens. Οι αλγόριθμοι BR και MILP πετυχαίνουν ίδια αποτελέσματα και βρίσκονται κοντά στο PSO. Ο αλγόριθμος LR, ωστόσο, χρησιμοποιεί, σχεδόν, όλα τα διαθέσιμα tokens. Στο συγκεκριμένο πίνακα, όλες οι δημοπρασίες έχουν συμπεριληφθεί. Σημαντικός είναι και ο επόμενος πίνακας όπου θα εξετάσουμε το ποσοστό των αποδεκτών λύσεων μόνο, δηλαδή τις λύσεις που ικανοποιούν τους περιορισμούς της παραγωγής και των tokens.

Αλγόριθμοι	Remaining tokens (%)
<i>DP</i>	-109.3%
<i>BR</i>	20.74%
<i>PSO</i>	23.4%
<i>MILP</i>	20.76%
<i>LR</i>	26.65%

Παρατηρούμε ότι τα αποτελέσματα παραμένουν περίπου ίδια στους 4 από τους 5 αλγόριθμους. Ωστόσο, παρατηρείται σημαντική βελτίωση στο αλγόριθμο LR, με αποτέλεσμα να είναι ο πιο οικονομικά συμφέρων πλέον. Καταλήγουμε στο ίδιο συμπέρασμα λοιπόν με αυτό που καταλήξαμε σχετικά με τις επιδόσεις των αλγορίθμων στην κάλυψη της ενέργειας, δηλαδή ότι, αν ο αλγόριθμος LR δώσει λύση εντός των περιορισμών που έχουν τεθεί, θα δώσει πιθανότατα και την καλύτερη λύση.

6.2 Σύγκριση επιδόσεων των αλγορίθμων σχετικά με τον χρόνο εκτέλεσής τους

Σε αυτό το κεφάλαιο, θα εξεταστούν και θα συγκριθούν οι χρόνοι εκτέλεσης του κάθε αλγόριθμου. Για το συγκεκριμένο πείραμα, χρησιμοποιήθηκαν 20 δημοπρασίες με τουλάχιστον 100 διαφορετικές προσφορές στην καθεμία. Όλες οι δημοπρασίες λύθηκαν από όλους τους αλγόριθμους και τα αποτελέσματα φαίνονται στον παρακάτω πίνακα.

Αλγόριθμοι	Average Time (s)	Standard Deviation
<i>DP</i>	7.19	1.73
<i>BR</i>	1.55	0.98
<i>PSO</i>	5.99	0.88
<i>MILP</i>	1.16	1.14
<i>LR</i>	11.11	8.23

Παρατηρούμε ότι οι πιο γρήγοροι αλγόριθμοι ήταν ο BR και MILP με μέσο χρόνο λύσης μιας δημοπρασίας 1.55 και 1.16 δευτερόλεπτα αντίστοιχα. Στη συνέχεια, ακολουθούν οι αλγόριθμοι PSO και DP με 5.99 και 7.19 δευτερόλεπτα, και τέλος, ο αλγόριθμος LR με μέσο χρόνος εκτέλεσης 11.11 δευτερόλεπτα. Ωστόσο, ο LR είναι και ο μόνος αλγόριθμος, ο οποίος παρουσιάζει μεγάλη τυπική απόκλιση. Αυτό οφείλεται στο γεγονός ότι ο αλγόριθμος αυτός σταματάει αν βρει μια ικανοποιητική λύση στις πρώτες επαναλήψεις. Έτσι βρέθηκαν δημοπρασίες στις οποίες χρειάστηκαν μόνο 0.02 δευτερόλεπτα να λυθούν και άλλες που χρειαζόταν τουλάχιστον 20 δευτερόλεπτα. Αυτό οφείλεται στην πολυπλοκότητα της κάθε δημοπρασίας. Για να βεβαιωθούν τα συμπεράσματα, το συγκεκριμένο πείραμα εκτελέστηκε 4 φορές και τα αποτελέσματα παρέμειναν στα ίδια επίπεδα με ελάχιστες αποκλίσεις.

6.3 Τελικά συμπεράσματα

Λαμβάνοντας υπόψιν τα προαναφερθέντα, τα ερωτήματα που τέθηκαν στο ερώτημα

3.1 μπορούν να απαντηθούν:

- α) Μπορούν να εφαρμοστούν αλγόριθμοι που λύνουν ένα UC πρόβλημα για την εύρεση νικητών σε μια δημοπρασία ενός DSO, που προσφέρει ενέργεια σε στόλους ηλεκτρικών οχημάτων για την εξισορρόπηση του δικτύου;

Απάντηση: Παρατηρείται ότι με τις κατάλληλες προσαρμογές στην αντικειμενική συνάρτηση του UC προβλήματος, μπορούν να χρησιμοποιηθούν οι ίδιες τεχνικές και αλγόριθμοι για την επίλυση μιας δημοπρασίας, και μάλιστα, αρκετά αποτελεσματικά. Εξετάστηκαν, μάλιστα, διάφορες περιπτώσεις και χρησιμοποιώντας κάθε φορά τον κατάλληλο αλγόριθμο, η δημοπρασία μπορεί να λυθεί.

- β) Αν ναι, τι συμπεράσματα προκύπτουν;

Απάντηση: Παρατηρείται ότι ένας αλγόριθμος είναι κατάλληλος ανάλογα και με την εξεταζόμενη περίπτωση. Συγκεκριμένα, μια δημοπρασία στην οποία ο ανταγωνισμός είναι μικρός ή η δημοπρατούμενη ενέργεια είναι σχετικά λίγη, τότε ιδανικοί αλγόριθμοι για την κάλυψη της ενέργειας είναι ο MILP και BR. Οι 2 αυτοί αλγόριθμοι πρότειναν ακριβώς τις ίδιες λύσεις στο ~ 99.9% των δημοπρασιών, και επιπλέον, είχαν παρόμοιους χρόνους εκτέλεσης. Αν η δημοπρασία, όμως, που καλείται να κληθεί έχει μεγάλο ανταγωνισμό, τότε καλύτερη λύση θεωρείται ο PSO. Σε κάθε περίπτωση, όμως, καλύτερη απόδοση από όλους έχει ο LR, όταν αυτός μπορεί να λύσει μια δημοπρασία και ο χρόνος επίλυσης μιας δημοπρασίας είναι αδιάφορος, καθώς σε πολύ πολύπλοκες δημοπρασίες αυτός μπορεί να αυξηθεί σημαντικά. Σε ακραίες περιπτώσεις, όμως, όπου απαιτείται μια λύση σε σύντομο χρονικό πλαίσιο και ο διαχειριστής επιθυμεί να καλύψει την ενέργεια ανεξαρτήτως του κόστους, τότε καλύτερος θεωρείται ο DP.

ΚΕΦΑΛΑΙΟ 7 ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ

7. Μελλοντική εργασία

Η ηλεκτροκίνηση αποτελεί ένα φλέγον ζήτημα που απασχολεί και θα συνεχίσει να απασχολεί για αρκετά χρόνια, όχι μόνο τους ιδιωτικούς και δημόσιους φορείς, αλλά και εμάς τους ίδιους. Η μετάβαση, ωστόσο, από τα οχήματα εσωτερικής καύσης σε ηλεκτρικά οχήματα είναι μια απαιτητική διαδικασία, η οποία πρέπει να γίνει ομαλά και προσεκτικά. Το μεγάλο φορτίο ηλεκτρικού ρεύματος που θα δημιουργηθεί από αυτή τη μετάβαση θα προκαλέσει πολλές αλλαγές στην αγορά της ενέργειας, με αποτέλεσμα να δημιουργεί αρκετά προβλήματα, αν δεν υπάρχουν οι κατάλληλοι μηχανισμοί.

Η πρωτοβουλία του έργου H2020 SOFIE για την εκμετάλλευση της περισσευούμενης ενέργειας από ηλεκτρικούς στόλους δείχνει πως μπορούν τα πιθανά προβλήματα που θα δημιουργούνται στο δίκτυο να αντιμετωπιστούν αποτελεσματικά, δίνοντας τα κατάλληλα κίνητρα σε στόλους ηλεκτρικών οχημάτων να φορτίσουν, και έτσι, να

εξισορροπήσουν το δίκτυο. Αντίστοιχα, όμως, το δίκτυο μπορεί να έχει ανάγκη από ηλεκτρικό ρεύμα. Το πρόβλημα αυτό θα μπορούσε να αντιμετωπιστεί με φορτιστές τύπου V2G (vehicle-to-grid), οι οποίοι επιτρέπουν σε ένα ηλεκτρικό όχημα να αποφορτίσει την μπαταρία του, παρέχοντας έτσι την απαραίτητη ενέργεια στο δίκτυο. Οι διαχειριστές δικτύων, αντίστοιχα, χρησιμοποιώντας πλατφόρμες, όπως αυτή του SOFIE, μπορούν να ετοιμάσουν το έδαφος για δημοπρασίες αποφόρτισης του δικτύου, δίνοντας κίνητρα σε στόλους για να αποφορτίσουν τα οχήματά τους ή ακόμα και για να μη φορτίσουν.

Σημασία έχουν επίσης και τα κίνητρα που θα δίνονται στους οδηγούς ηλεκτρικών οχημάτων. Για να υπάρχει μεγάλη συμμετοχή σε τέτοιους είδους ενέργειες πρέπει οι οδηγοί να έχουν κάποιο σοβαρό κίνητρο για να παρακινηθούν, όπως έκπτωση σε επόμενη φόρτιση και cryptocurrencies που γίνονται αποδεκτά σε διάφορες υπηρεσίες και καταστήματα για την αγορά προϊόντων και παροχών.

Τέλος, σημαντικό ενδιαφέρον έχει και ο τρόπος λειτουργίας τέτοιων νέων αγορών. Πιο συγκεκριμένα, στη διπλωματική εξετάστηκε πως θα μπορούσε να λυθεί μια δημοπρασία στην οποία το δημοπρατούμενο αγαθό ήταν η πρόβλεψη του διαχειριστή δικτύου για την περισσευούμενη ενέργεια που θα έχει αύριο σε κάποιο συγκεκριμένο χρονικό διάστημα. Οι προβλέψεις αυτές, όμως, πολλές φορές μπορούν να έχουν μεγάλο σφάλμα, με αποτέλεσμα, είτε να χρειάζεται περισσότερη ενέργεια το δίκτυο για να εξυπηρετήσει τους «νικητές» της δημοπρασίας (άθροισμα αποδεκτής ενέργειας > πραγματική ζήτηση ενέργειας), είτε να χρειάζονται περισσότερα ηλεκτρικά οχήματα να φορτίσουν (άθροισμα αποδεκτής ενέργειας < πραγματική ζήτηση ενέργειας). Ένας πιθανός τρόπος να αντιμετωπιστούν τέτοια σφάλματα είναι η προσομοίωση του τρόπου λειτουργίας του χρηματιστηρίου ενέργειας, δηλαδή η περάτωση νέων δημοπρασιών πιο κοντά στο χρόνο παράδοσης της ηλεκτρικής ενέργειας, μειώνοντας έτσι το μέγεθος του σφάλματος.

Βιβλιογραφία

1. Υπουργείο Περιβάλλοντος και Ενέργειας, Νόμος Υπ' Αριθμ. 3468 - ΦΕΚ Α 129/27.06.2006 (2006). Available at:
<https://www.kodiko.gr/nomothesia/document/11708/nomos-3468-2006>.
2. Υπουργείων Οικονομικών, Περιβάλλοντος και Ενέργειας, Υποδομών, Μεταφορών και Δικτύων και Εργασίας, Κοινωνικής Ασφάλισης και Κοινωνικής Αλληλεγγύης, Νόμος Υπ' Αριθμ. 4425 (2016) Taxheaven. Available at:
<https://www.taxheaven.gr/law/4425/2016>.
3. Hellenic Energy Exchange S.A., Henex Spot Trading Rulebook – EnExGroup (2021). Available at:

https://www.enexgroup.gr/documents/20126/144557/Spot_Trading_Rulebook_v1.5_en.pdf/.

4. Hellenic Energy Exchange S.A., Henex Spot Trading Rulebook, Financial Energy Market Rulebook (derivatives market) - EnExGroup (2020). Available at: https://www.enexgroup.gr/documents/20126/184422/20200305_Derivatives_Tr_Rulebook_EN.pdf.

5. ΑΔΜΗΕ, Κανονισμός Αγοράς Εξισορρόπησης (2022). Available at: https://www.admie.gr/sites/default/files/users/dda/KAE/%CE%9A%CE%91%CE%9D%CE%9F%CE%9D%CE%99%CE%A3%CE%9C%CE%9F%CE%A3%20%CE%91%CE%93%CE%9F%CE%A1%CE%91%CE%A3%20%CE%95%CE%9E%CE%99%CE%A3%CE%9F%CE%A1%CE%A1%CE%9F%CE%A0%CE%97%CE%A3%CE%97%CE%A3_v12.pdf

6. This is how the Electricity Market Works (no date) SMARD. Available at: <https://www.smard.de/page/en/wiki-article/5884/5840>.

7. Thakur, N. and Titare, L.S. (2016) Determination of Unit Commitment Problem Using Dynamic Programming, Determination of Unit Commitment-540.pdf. Available at: <https://www.noveltyjournals.com/upload/paper/Determination%20of%20Unit%20Commitment-540.pdf>.

8. Gautam, A. et al. (2023) ‘Methods and methodologies for congestion alleviation in the DPS: A comprehensive review’, *Energies*, 16(4), p. 1765. doi:10.3390/en16041765.

9. Montero, L., Bello, A. and Reneses, J. (2022) ‘A review on the unit commitment problem: Approaches, techniques, and Resolution Methods’, *Energies*, 15(4), p. 1296. doi:10.3390/en15041296.

10. Gupta, M. et al. (2017) ‘Mitigating congestion in a power system and role of facts devices’, *Advances in Electrical Engineering*, 2017, pp. 1–7. doi:10.1155/2017/4862428.

11. Stawska, A. et al. (2021) ‘Demand response: For congestion management or for grid balancing?’, *Energy Policy*, 148, p. 111920. doi:10.1016/j.enpol.2020.111920.

12. Blatiak, A. et al. (2022) 'Value of optimal trip and charging scheduling of commercial electric vehicle fleets with vehicle-to-grid in future low inertia systems', *Sustainable Energy, Grids and Networks*, 31, p. 100738.
doi:10.1016/j.segan.2022.100738.
13. Lago, J. et al. (2021) 'A market framework for grid balancing support through imbalances trading', *Renewable and Sustainable Energy Reviews*, 137, p. 110467.
doi:10.1016/j.rser.2020.110467.
14. Najafi, S. and pourjamal, Y. (2012) 'A new heuristic algorithm for Unit Commitment Problem', *Energy Procedia*, 14, pp. 2005–2011.
doi:10.1016/j.egypro.2011.12.1201.
15. Nicolosi, F.F. et al. (2021) 'Unit commitment optimization of a micro-grid with a MILP algorithm: Role of the emissions, bio-fuels and Power Generation Technology', *Energy Reports*, 7, pp. 8639–8651. doi:10.1016/j.egy.2021.04.020.
16. Yu, X. and Zhang, X. (2014) 'Unit commitment using lagrangian relaxation and particle swarm optimization', *International Journal of Electrical Power & Energy Systems*, 61, pp. 510–522. doi:10.1016/j.ijepes.2014.03.061.
17. Neissery Abdulla, S. *Dynamic Programming*, (PDF) *Dynamic Programming - Researchgate*. Available at:
https://www.researchgate.net/publication/329390711_Dynamic_programming.
18. Bellman, R.E. (1954) *The theory of dynamic programming*. Available at:
<http://smo.sogang.ac.kr/doc/bellman.pdf>.
19. Singhal, P.K. and Sharma, R.N. (2011) 'Dynamic programming approach for large scale unit commitment problem', 2011 *International Conference on Communication Systems and Network Technologies* [Preprint].
doi:10.1109/csnt.2011.152.
20. Pushpa, S.K., Mrunal, T.V. and Suhas, S. (2016) *A Study of Performance Analysis on Knapsack Problem*. Available at:
<https://research.ijcaonline.org/ncrtit2016/number2/ncrtit201626.pdf>.

21. Ioan, D. et al. (2020) Mixed-integer programming in motion planning, Mixed-integer programming in motion planning - ScienceDirect. Available at: <https://www.sciencedirect.com/science/article/abs/pii/S1367578820300754>.
22. Fischetti, M., Monaci, M. and Salvagnin, D. (2016) 'Mixed-integer linear programming heuristics for the prepack optimization problem', *Discrete Optimization*, 22, pp. 195–205. doi:10.1016/j.disopt.2015.03.004.
23. Lodi, A. (2009) 'Mixed integer programming computation', *50 Years of Integer Programming 1958-2008*, pp. 619–645. doi:10.1007/978-3-540-68279-0_16.
24. Viana, A. and Pedroso, J.P. (2013) 'A new MILP-based approach for unit commitment in power production planning', *International Journal of Electrical Power & Energy Systems*, 44(1), pp. 997–1005. doi:10.1016/j.ijepes.2012.08.046.
25. Kantor, I. et al. (2020) 'A mixed-integer linear programming formulation for optimizing multi-scale material and Energy Integration', *Frontiers in Energy Research*, 8. doi:10.3389/fenrg.2020.00049.
26. Mitchell, S., O'Sullivan, M. and Dunning, I. (2011) Pulp: A linear programming toolkit for python, 2011. Pulp: A linear programming toolkit for python.pdf. Available at: <https://www.dit.uoi.gr/e-class/modules/document/file.php/216/PAPERS/2011.%20PuLP%20-%20A%20Linear%20Programming%20Toolkit%20for%20Python.pdf>.
27. Gad, A.G. (2022) 'Particle swarm optimization algorithm and its applications: A systematic review', *Archives of Computational Methods in Engineering*, 29(5), pp. 2531–2561. doi:10.1007/s11831-021-09694-4.
28. Logenthiran, T. and Srinivasan, D. (2010) 'Particle swarm optimization for unit commitment problem', *2010 IEEE 11th International Conference on Probabilistic Methods Applied to Power Systems [Preprint]*. doi:10.1109/pmaps.2010.5528899.
29. Yuan, X. et al. (2009) 'An improved binary particle swarm optimization for unit commitment problem', *Expert Systems with Applications*, 36(4), pp. 8049–8055. doi:10.1016/j.eswa.2008.10.047.

30. Sriyanyong, P. and Song, Y.H. (2005) 'Unit commitment using particle swarm optimization combined with Lagrange relaxation', IEEE Power Engineering Society General Meeting, 2005. doi:10.1109/pes.2005.1489390.
31. Bakirtzis, A.G. and Zoumas, C.E. (2000) 'Lambda of lagrangian relaxation solution to unit commitment problem', IEE Proceedings - Generation, Transmission and Distribution, 147(2), p. 131. doi:10.1049/ip-gtd:20000173.