

Minimizacija stroškov električne energije v pametni hiši z uporabo globokega spodbujevanega učenja

Matic Pokorn, Jernej Hribar

Department of Communication Systems, Jožef Stefan Institute, Slovenia
mp3558@student.uni-lj.si

Electrical energy cost minimization in a smart home using Deep Reinforcement Learning

In this paper, we design a Home Energy Management System (HEMS) that reduces the energy cost in a smart home using Deep Reinforcement Learning (DRL). Our proposed solution considers the recently proposed Slovenian energy pricing model and can minimize the overall energy cost of the system. We demonstrate in a simulated environment that by learning how to trade the energy back to the electrical grid, it is possible to save up to 44.4% compared to conventional rule-based approaches without altering electricity consumption inside the home.

1 Uvod

V zadnjih letih Evropa doživlja prehod v energetske politiki s CEMS (Centralised Energy Management Systems) na DEMS (Decentralised Energy Management Systems) [1]. V sklopu teh sprememb mnoge evropske države, med njimi tudi Slovenija, uvajajo dinamične oblike zaračunavanja cene energije, ki se spreminja glede na čas v dnevu. Dinamična cena energije ponuja domačim sistemom upravljanja energije (angl. Home Energy Management System (HEMS)) s fotovoltaičnimi (angl. photovoltaics (PV)) celicami in baterijo za shranjevanje energije možnost znižanja stroškov energije s prilagajanjem na cene, hkrati pa pomagajo distribucijskim operaterjem pri uravnavanju obremenitve, saj je cena elektrike dober pokazatelj razpoložljivosti energije. V tem članku predstavljamo rešitev, osnovano na globokem spodbujevanem učenju (angl. Deep Reinforcement Learning (DRL)), ki se prilagaja dinamičnim spremembam cen električne energije in znižuje stroške.

Za doseg zastavljenega cilja minimizacije stroškov energije se lahko HEMS na primer odloči, da bo shranil energijo, ki jo je proizvedla PV, ali pa kupil več energije iz omrežja, ki jo bo kasneje prodal, ko se cene zvišajo, in podobno. HEMS mora najti način, kako izbrati najboljšo možnost glede porabe električne energije v vsakem trenutku odločanja. V takšnem primeru imajo rešitve, ki se učijo, prednost pred heurističnimi rešitvami pri prilagajanju na spremembe v sistemu ob upoštevanju trenutnih razmer, količine že shranjene energije v bateriji,

itd. Pri podobnih problemih izven področja upravljanja z energijo je bilo v preteklosti spodbujevano učenje že večkrat uporabljeno. Eden od primerov so finance [2], kjer je spodbujevano učenje postalo priljubljena tehnika v algoritemskem trgovanju. Naš problem, ki se sicer nahaja v domeni upravljanja energije, obravnavamo kot problem trgovanja, zato s pomočjo DRL-ja predlagamo rešitev, ki je sposobna najti najboljšo možno odločitev v dinamičnem okolju, obenem pa je tudi prilagodljiva na morebitne nenadne spremembe.

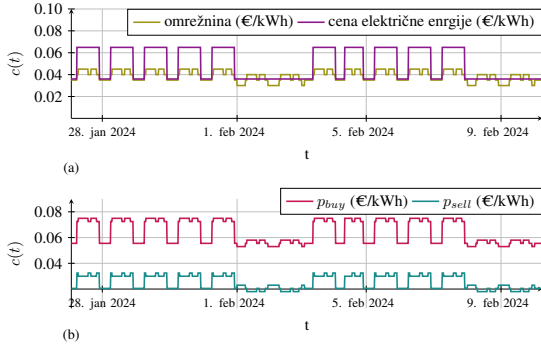
V tem članku se za razliko od že obstoječih raziskav osredotočamo na samostojen HEMS in ga učimo na več spremenljivih parametrih, tako da konstruiramo model sistema, kjer so lahko velikost baterije, učinkovitost baterije in moč baterije poljubne. Spremembo cene energije modeliramo, kot se bo spreminjala po predlaganem slovenskem tarifnem sistemu zaračunavanja električne energije. Naša rešitev, zasnovana na osnovi algoritma globokega Q-omrežja (angl. Deep Q-Network (DQN) [3]), izbira med štirimi možnostmi, kako se sistem prilagodi na vsakem odločitvenem koraku glede na trenutne razmere v sistemu. S pomočjo simulacij in podatkov, pridobljenih iz realnega okolja, dokažemo, da je predlagana rešitev zmožna zmanjšati stroške energije za do 44,4% v primerjavi s heurističnimi rešitvami.

2 Pregled področja

Veliko raziskav je bilo usmerjenih v reševanje izzivov upravljanja z energijo v pametnih hišah [4–9]. Kljub temu ostaja problem samostojnega HEMS brez nadzora nad hišnimi aparati področje, ki je ostalo bolj ali manj neraziskano. Na primer, v [4–7] so se osredotočali na optimizacijo stroškov v skupnostih (zaselek, stanovanjski blok, itd.). Druge rešitve so dosegle znižanje stroškov energije z upravljanjem (izklop/vklop) hišnih aparatov [8, 9]. Spodbujevano učenje je bilo uporabljeno samo v peščici omenjenih rešitev [6, 9]. V teh dveh primerih so uporabili DRL algoritme, vendar se naše delo razlikuje od njih, saj dosega znižanje stroškov energije v gospodinjstvih brez upravljanja porabnikov energije.

3 Model sistema

V tem razdelku definiramo model sistema in opišemo problem. V prvem podrazdelku predstavimo dinamični



Slika 1: Omrežnina in cena električne energije.

model cenitve, strukturo lokalnega energetskega sistema ter njegove spremenljivke in omejitve. V naslednjem podrazdelku obravnavamo problem minimizacije stroškov in ga povežemo s predstavljenim modelom.

3.1 Dinamični model zaračunavanja elektrike

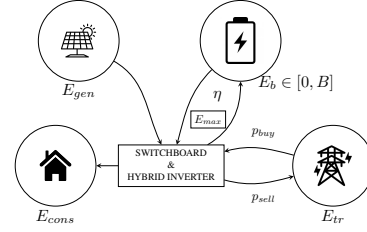
Glavnino dinamičnega dela slovenskega tarifnega sistema predstavljata dejanska cena električne energije in omrežnina, ki sta trenutno obe obračunani po dveh tarifah - višji tarifi (VT) in manjši tarifi (MT). Po novem sistemu, ki stopi v veljavo leta 2024, bo omrežnina obračunana po novem, pet-tarifnem sistemu [10], zaradi česar bo cena postala bolj dinamična in posledično bolj zanimiva za minimizacijo stroškov. V našem sistemu, kjer želimo elektriko prodajati, moramo vseeno plačati omrežnino. Tako je skupna cena elektrike, ki jo prodamo, precej nižja kot cena, po kateri elektriko kupimo.

Slika 1 prikazuje nihanje kupne (p_{buy}) in prodajne cene (p_{sell}). Slika 1a prikazuje cene električne energije in omrežnine v odvisnosti od časa. Slika 1b prikazuje kupno in prodajno ceno energije v našem omrežju. Kupna cena je vsota cene električne energije in omrežnine, prodajna cena pa njuna razlika. Če je cena omrežnine višja od cene energije, ima prodajna cena lahko tudi negativno vrednost.

3.2 Domači sistem za upravljanje energije (HEMS)

Obravnavamo HEMS samostojne pametne hiše s PV strešnimi celicami in baterijo za shranjevanje energije z omejeno kapaciteto. PV proizvaja energijo glede na vremenske razmere in čas v dnevu. Na voljo imamo baterijo, ki lahko shrani B kWh energije. Prav tako predpostavimo, da ima baterija omejeno moč polnjenja in praznjenja (P_{max}) ter izgubo med polnjenjem in praznjenjem (η). Predpostavljamo, da smo v okolju z dinamično ceno $p(t)$.

Slika 2 predstavlja stanje modela sistema v enem časovnem koraku. Baterija je omejena z maksimalno kapaciteto B in maksimalno količino energije E_{max} , ki je lahko prenešana v in iz nje v enem časovnem koraku. Omrežje je definirano kot neskončni vir in neskončni porabnik, tako da se lahko med njim in lokalnim sistemom izmenja poljubna količina energije. Vsak transfer se pomnoži s ceno, odvisno od smeri pa je, ali bo ta cena p_{buy} ali p_{sell} . E_{gen} je energija, ki jo proizvede PV in E_{cons} je energija, ki jo porabi



Slika 2: Grafična predstavitev HEMS-a.

odjemalec.

Sistem mora zagotoviti dovolj energije za vse porabnike v pametni hiši, obenem pa omogočiti tudi prodajo energije v omrežje. Zato definiramo spremenljivki E_{sell} in E_{buy} , ki predstavljata energijo, ki se izmenjuje med HEMS-om in zunanjim energetskega omrežjem. Nadalje definiramo največjo energijo, ki lahko napolni ali izprazni baterijo v enem koraku t :

$$E_{max \text{ discharge}}(t) = \eta \min\{E_b(t), E_{max}\}, \quad (1)$$

$$E_{max \text{ charge}}(t) = \frac{1}{\eta} \min\{B - E_b(t), E_{max}\}, \quad (2)$$

$$\Delta E_b(t) \in [-E_{max \text{ discharge}}(t), E_{max \text{ charge}}(t)], \quad (3)$$

$$\Delta E_b(t) = \eta E_{pv \rightarrow b}(t) + \eta E_{grid \rightarrow b}(t) - \frac{1}{\eta} E_{b \rightarrow home}(t) - \frac{1}{\eta} E_{b \rightarrow grid}(t), \quad (4)$$

kjer faktor η predstavlja učinkovitost polnjenja in praznjenja baterije, $E_{pv \rightarrow b}(t)$, $E_{grid \rightarrow b}(t)$, $E_{b \rightarrow home}(t)$, $E_{b \rightarrow grid}(t)$ pa energije, ki se izmenjujejo med entitetami, ki so vključene v sistem.

Ceno energije na koraku t izračunamo na sledeči način:

$$E_b(t) \in [0, B] \quad (5a)$$

$$E_b(t) = E_b(t-1) + \Delta E_b(t) \quad (5b)$$

$$E_{buy}(t) = E_{grid \rightarrow home}(t) + E_{grid \rightarrow b}(t) \quad (5c)$$

$$E_{sell}(t) = E_{pv \rightarrow grid}(t) + E_{b \rightarrow grid}(t) \quad (5d)$$

$$c(t) = p_{buy}(t) E_{buy}(t) - p_{sell}(t) E_{sell}(t) \quad (5e)$$

$$C = \sum_{t=1}^T c(t), \quad t \in \{1, \dots, T\}, \quad (5f)$$

kjer je T dolžina časovnega intervala. C predstavlja kumulativni strošek energije v našem sistemu in v našem interesu je minimizirati to vrednost.

3.3 Cilj

Glavni cilj sistema je minimizirati kumulativni strošek na željenem časovnem intervalu T :

$$\min \sum_{t=1}^T c(t) = \max \sum_{t=1}^T -c(t) \quad (6)$$

E_{buy} and E_{sell} lahko nadziramo (upoštevajoč omejitve, opisane v Eq. 3 in Eq. 5a). Posledično lahko manipuliramo tudi stanje napolnjenosti baterije $E_b(t)$ in stroške $c(t)$ na poljubni točki $t \in \{1, \dots, T\}$ in računamo na to, da bomo prihranili nekaj denarja. Ker imamo v sistemu dinamične faktorje (E_{gen} , E_{cons} in p_{buy} ter p_{sell}), za iskanje optimalne rešitve uporabimo DRL.

4 Predlagana rešitev

Predlagana rešitev temelji na algoritmu DQN [3], ki uporablja Q-učenje in je bil razvit za Markovski odločitveni proces (angl. Markov Decision Process (MDP)) z velikimi prostori stanj (angl. state spaces). V našem primeru, kot bomo pokazali v nadaljevanju, se srečujemo s problemom velike dimenzionalnosti zaradi zveznosti spremenljivk, ki sestavljajo prostor stanj. Ena izmed prednosti Q-učenja je učenje brez modela, kar pomeni, da ne potrebujemo informacij o verjetnostih prehodov med stanji MDP modela.

4.1 Markovski odločitveni proces (MDP)

Za oblikovanje okolja MDP je potrebno definirati prostor stanj, prostor dejanj in nagradno funkcijo:

Prostor stanj: posamezno stanje $s(t) \in \mathcal{S}$ je definirano z naslednjimi količinami v zadnjih 48 urah. Ker imajo podatki 15-minutne meritve, 48 ur sestavlja 192 časovnih točk. Torej:

$$s(t) = \{E_b(t-192) \cdots E_b(t), E_{gen}(t-192) \cdots E_{gen}(t), \\ E_{cons}(t-192) \cdots E_{cons}(t), el(t-192) \cdots el(t), \\ E_{omr}(t-192) \cdots omr(t), c(t-192) \cdots c(t)\} \quad (7)$$

Tukaj $omr(t)$ in $el(t)$ predstavljata tarifni razred omrežnine in električne energije.

Prostor dejanj: imamo štiri dejanja $a(t) \in \mathcal{A}$, ki jih lahko izvede algoritem. Vsa uporabljajo verzijo rezanja viškov, s katerim se porablja viške PV energije. Najprej prerazporedijo vso energijo, ki je bila proizvedena s PV, nato pa vsaka na drugačen način reši problem odvečne proizvedene energije ali višjega odjema:

- Napajaj hišo iz omrežja, polni baterijo iz omrežja;
- Napajaj hišo iz baterije, prodaj energijo iz baterije;
- Napajaj hišo samo iz omrežja;
- Napajaj hišo iz baterije (če gre, sicer iz omrežja);

Nagradna funkcija: Nagradna funkcija je sestavljena iz treh členov:

$$R(t) = R_1(t) + R_2(t) + R_3(t) \quad (8)$$

Prvi člen doseže, da baterija ni nikoli polna, ker bi potem moral sistem energijo zavreči, niti prazna, ker omeji možnosti trgovanja:

$$R_1(t) = \begin{cases} -4(b(t))^2 & , E_b(t) > \beta \\ 0 & , \alpha < E_b(t) < \beta \\ -2(b(t))^2 & , E_b(t) < \alpha \end{cases} \quad (9)$$

Parametra α in β predstavljata nivoje napolnjenosti baterije, med katerima RL algoritem prejme kazni.

Drugi člen predstavlja razmerje med polnjenjem/praznjenjem baterije in ceno energije. Na primer, če je naklon funkcije polnjenja baterije pozitiven (baterija se polni) in je cena elektrike nad mediano, algoritem prejme kazni (če je cena visoka, ima smisel poskusiti izprazniti baterijo oz. prodati energijo).

$$R_2(t) = -\gamma (E_b(t) - E_b(t-1)) (p_{buy}(t) - \tilde{p}_{buy}(t)), \quad (10)$$

$\tilde{p}(t)$ predstavlja mediano prodajne cene v zadnjih 48 urah v točki t .

Tretji člen predstavlja količino, ki jo plačamo in jo želimo optimizirati:

$$R_3(t) = -\delta c(t). \quad (11)$$

Parametra γ in δ uravnotežita pomembnost posameznih členov. Člen 9 torej služi temu, da baterijo ohranja v "stabilnem" stanju, medtem ko kombinacija med preostalima dvema členoma išče spremembe v ceni na trgu in poskuša doseči profit.

4.2 Algoritem DQN

Algoritem DQN je algoritem s področja DRL, ki združuje globoke nevronske mreže z algoritmom Q-učenja in s tem omogoči rešitvi, da se nauči optimalne strategije v kompleksnih okoljih. Stanje okolja, tj., $s(t)$, je vhod v nevronske mreže, izhodi pa predstavljajo dejanja. Algoritem DQN uporablja ponovni pregled izkušenj (angl. experience replay) za izboljšanje učinkovitosti vzorcev. Izkušnje shrani v svojem medpomnilniku (angl. replay buffer) in naključno vzorči mini-serije (angl. mini-batch) izkušenj med treniranjem. Algoritem začne naključno raziskovati okolje in tekom treniranja postopoma preide k izkoriššanju naučenih Q-vrednosti da dela odločitve. Uporablja postopek iterativnega učenja, ki algoritmu DQN omogoča, da s časom izboljša svojo strategijo.

5 Rezultati

V tem razdelku analiziramo učinkovitost predlagane rešitve in jo primerjamo z dvema izhodiščnima rešitvama.

5.1 Izhodiščni strategiji

Našo predlagano rešitev primerjamo z:

Izhodiščna strategija 1 (IS1):- Požrešna: obravnava hišo samo s PV brez baterije. Sledeča enačba opisuje, kako je izračunan strošek električne energije:

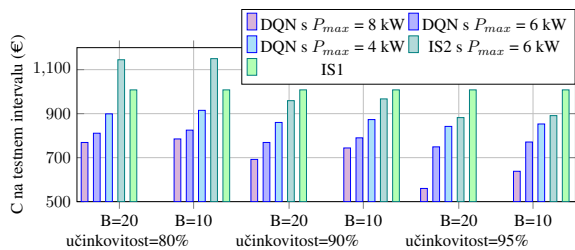
$$c(t) = \begin{cases} -p_{sell}(t) (E_{gen}(t) - E_{cons}(t)) & ; E_{gen}(t) > E_{cons}(t) \\ p_{buy}(t) (E_{cons}(t) - E_{gen}(t)) & ; E_{gen}(t) < E_{cons}(t) \end{cases} \quad (12)$$

Izhodiščna strategija 2 (IS2): - Alternirajoča: uporablja štiri dejanja, kot jih uporablja predlagana rešitev, ki se izmenjujejo ob vsakem časovnem koraku.

Prva izhodiščna strategija nam pomaga, da dokažemo prednosti hranjenja energije v baterijo kljub izgubam pri polnjenju / praznjenju, druga pa, da dokažemo, da so rezultati predlagane rešitve boljši zaradi načrtno izbranih dejanj.

5.2 Podatki

V simulacijah smo uporabili realne meritve s pametne hiše v Solunu, Grčija, na 15-minutnih intervalih za obdobje od oktobra 2018 do oktobra 2020. Podatkovna



Slika 3: Rezultati z baterijama velikosti 20 kWh in 10 kWh.

zbirka vključuje 59.844 podatkovnih točk [11], ki zajemajo proizvodnjo in porabo energije s sončnimi celicami. Cenovno funkcijo smo generalizirali z uporabo slovenskih tarif in cen električne energije ter prilagodili trenutnim cenam. Podatkovno zbirko smo razdelili na polovico, kjer prva polovica predstavlja učno, druga pa testno množico.

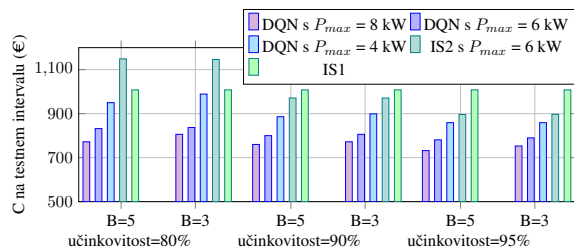
Treniranje algoritma DQN je obsegalo 200 epizod iz učne množice, vsaka s 7-dnevnim intervalom. Postopek smo večkrat ponovili zaradi nestabilnosti algoritma DQN [12], a smo dosegli želeni izid. Rešitev smo ocenili z različnimi lastnostmi baterije: velikosti (3kWh, 5kWh, 10kWh, 20kWh), močjo polnjenja/praznjenja (4kW, 6kW, 8kW) in učinkovitostjo (80%, 90%, 95%).

5.3 Uspešnost

Na sliki 3 je prikazana primerjava kumulativnih stroškov med uporabo rešitve algoritma DQN in izhodiščnih rešitev pri uporabi baterij z velikostjo 20 kWh in 10 kWh. Izmed obravnavanih konfiguracij je najnižje stroške dosegla rešitev algoritma DQN z baterijo velikosti 20 kWh, močjo 8 kW in učinkovitostjo 95%. Podobno primerjavo kumulativnih stroškov prikazuje slika 4, tokrat z baterijami velikosti 5 kWh in 3 kWh. Tudi v tem primeru je rešitev algoritma DQN z baterijo velikosti 5 kWh, močjo 8 kW in učinkovitostjo 95% dosegla najnižje stroške. Vredno je omeniti, da smo na x-osi grafov prikazali posamezne konfiguracije sistema, medtem ko smo na y-osi prikazali skupne stroške, ki jih je moral lastnik pametne hiše plačati v času testiranja. Najboljšo učinkovitost smo dosegli pri rešitvi s parametri $B = 20kWh$, $P_{max} = 8kW$ in $\eta = 95\%$, ki je v primerjavi z algoritmom požrešne strategije prinašala 44,4% prihranka in letno prinašala 448 € prihranka. Rezultati, predstavljeni na slikah 3 in 4, kažejo, da naša predlagana rešitev omogoča prihranke pri stroških energije v različnih okoliščinah v primerjavi z izhodiščnimi rešitvami. Opazili smo, da se uspešnost rešitve zmanjšuje ob zmanjšanju učinkovitosti baterije, kljub temu pa je predlagana rešitev boljša od izhodiščnih rešitev v vseh primerih.

6 Zaključek

V tem članku smo raziskali problem minimizacije stroškov električne energije v pametnih hišah na primeru slovenskih dinamičnih cen elektrike. Rešili smo ga z algoritmom DRL in ugotovili, da lahko na ta način prihranimo tudi do 44,4% dinamičnega dela stroškov. V prihodnosti bi lahko uspešnost algoritma še izboljšali s



Slika 4: Rezultati z baterijama velikosti 5 kWh in 3 kWh. temeljito analizo nagradne funkcije in učnega procesa, okvir pa bi lahko razširili tudi na upravljanje z domačimi aparati.

Zahvala

Delo je bilo podprto s strani Slovenske raziskovalne agencije (P2-0016) in HORIZON-MSCA-IF projekt TimeSmart (Grant agreement ID: 101063721).

Literatura

- [1] D. Brown, S. Hall, and M. E. Davis, "What is prosumerism for? Exploring the normative dimensions of decentralised energy transitions," *Energy Research & Social Science*, vol. 66, p. 101475, 2020.
- [2] T. Théate and D. Ernst, "An application of deep reinforcement learning to algorithmic trading," *Expert Systems with Applications*, vol. 173, p. 114632, 2021.
- [3] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, 2015.
- [4] C. Changsong, D. Shanxu, C. Tao, L. Bangyin, and Y. Jinjun, "Energy trading model for optimal microgrid scheduling based on genetic algorithm," in *In proc. IEEE 6th IPEMC*, 2009, pp. 2136–2139.
- [5] Wang *et al.*, "A game-theoretic approach to energy trading in the smart grid," *IEEE Transactions on Smart Grid*, vol. 5, no. 3, pp. 1439–1450, 2014.
- [6] T. Chen and W. Su, "Local energy trading behavior modeling with deep reinforcement learning," *IEEE Access*, vol. 6, pp. 62 806–62 814, 2018.
- [7] R. Kanamori, T. Yoshimura, S. Kawaguchi, and T. Ito, "Evaluation of community-based electric power market with agent-based simulation," in *Proc. IEEE/WIC/ACM International Joint Conferences on WI and IAT*, vol. 2, 2013, pp. 108–113.
- [8] S. Bahramara, "Robust optimization of the flexibility-constrained energy management problem for a smart home with rooftop photovoltaic and an energy storage," *Journal of Energy Storage*, vol. 36, p. 102358, 2021.
- [9] Yu *et al.*, "Deep reinforcement learning for smart home energy management," *IEEE IoT-J*, vol. 7, no. 4, pp. 2751–2762, 2020.
- [10] "Akt o metodologiji za obračunavanje omrežnice za elektrooperaterje," <http://pisrs.si/Pis.web/pregledPredpisa?id=AKT.1266>, 2023.
- [11] Zyglakis *et al.*, "Greek smart house nanogrid dataset," Nov. 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.4246525>
- [12] M. Roderick, J. MacGlashan, and S. Tellex, "Implementing the deep q-network," 2017.