

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 3266

**WEB-MODUL ZA ANALIZU DOGAĐAJA NA
TRŽIŠTIMA ELEKTRIČNOM ENERGIJOM**

Kristijan Franković

Zagreb, lipanj 2013.

Sadržaj

Uvod	1
1. Natjecanje Power TAC	2
1.1. Motivacija	2
1.2. Opis simulacijske okoline	6
2. CrocodileAgent	10
3. Postojeća rješenja za obradu podataka	12
3.1. Power TAC Visualizer	12
3.2. Power TAC Logfile Analysis (PLA)	14
4. Web-modul za analizu događaja na tržištima električnom energijom	15
4.1. Korisnički zahtjevi	15
4.2. Funkcionalni zahtjevi	16
4.3. Nefunkcionalni zahtjevi	17
4.4. Slučajevi korištenja	17
4.4.1. Slučaj korištenja (1): Grafički prikaz podataka	18
4.4.2. Slučaj korištenja (2): Postavke stvarno-vremenske analize	19
4.4.3. Slučaj korištenja (3): Dohvat podataka	19
4.5. Arhitektura sustava YAPLA	20
4.6. Tehnologije, programske knjižnice i alati	21
4.6.1. Tehnologije	21
4.6.2. Alati i razvojno okružje	23
4.6.3. Dodatne programske knjižnice	23
4.7. Rezultati	24
4.7.1. Programsко rješenje	24
4.7.2. Povjesna analiza	29

4.7.3. Stvarno-vremenska analiza	34
5. Smjernice za daljnji rad	36
5.1. Vlastita baza podataka i modul za učitavanje dnevničkih datoteka.....	36
5.2. Proširenje modula za povijesnu analizu.....	37
5.3. Poboljšanje grafičkog korisničkog sučelja.....	37
5.4. Uvođenje dodatnih strategija za stvarno-vremensku analizu	38
5.4.1. Predviđanje potrošnje el. energije na temelju vremenskih uvjeta i doba dana	38
5.4.2. Predviđanje promjena cijene el. energije ovisno o broju sudionika u pojedinoj igri	39
5.5. Uvođenje modula za praćenje rezultata stvarno-vremenske analize	39
Zaključak	40
Literatura	41
Sažetak	43
Summary	44
Skraćenice.....	45

Uvod

Smanjenje zaliha i povećanje cijena fosilnih goriva, ali i rastući ekološki problemi povezani s konvencionalnim načinima proizvodnje električne energije, u prvi plan posljednjih godina sve više i više guraju obnovljive izvore el. energije kao rješenje za budućnost. Međutim, ovakav razvoj situacije pred tržište el. energije stavlja nove izazove, jer osim učinkovite i jeftine proizvodnje el. energije iz obnovljivih izvora treba osigurati i nove mehanizme za trgovinu istom, imajući pritom na umu sva ograničenja koja ovakva proizvodnja energije donosi. Jedan od ciljeva novih tržišnih modela također je i motivirati potrošnju energije iz održivih izvora i na drugi način potaknuti kupce ka ponašanju u skladu s novonastalom situacijom.

U tom kontekstu, razvijeno je natjecanje Power TAC (engl. *Power Trading Agent Competition*), međunarodni projekt šest sveučilišta iz Europe i Sjeverne Amerike, a jedan od članova jest i Sveučilište u Zagrebu. Odgovornost tima koji predstavlja Sveučilište u Zagrebu obuhvaća razvoj vizualizacijskog modula za platformu Power TAC, vrlo važne komponente presudne za uspjeh cijelog projekta, te vlastitog agenta koji sudjeluje u natjecanjima – Crocodile Agent.

Samo natjecanje Power TAC jest natjecanje koje modelira tržište el. energijom, a osmišljeno je kao otvorena računalna simulacijska platforma čiji je cilj pomoći definirati smjernice za dizajn novih vrsta tržišta el. energijom.

Rezultat pojedinih simulacija u okviru Power TAC simulacijske platforme jest velik broj podataka čija analiza može dati važne informacije pojedinim sudionicima natjecanja, a čija učinkovita obrada zahtjeva specijalizirana programska rješenja.

Predmet ovog završnog rada jest upravo modeliranje, razvoj i dokumentiranje jednog takvog alata, koji korisnicima omogućava jednostavnu analizu događaja u okruženju simulacijske platforme Power TAC. U konačnici to za rezultat ima kvalitetniju analizu informacija, te shodno tome rezultira donošenjem preciznijih zaključaka o novim modelima trgovine el. energijom, a potencijalno predstavlja jedan mali djelić doprinosa tržištima el. energija kakva će jednoga dana moguće opskrbljivati i naše domove.

1. Natjecanje Power TAC

Unutar ovog poglavlja izlažu se temeljni motivi za razvoj Power TAC natjecanja, a potom i opis simulacijske okoline te detaljan uvid u samo okružje natjecanja.

1.1. Motivacija

Posljednjih dvije stotine godina čovječanstvu su donijele tehnološki razvoj nevjerljivih razmjera. Od društva koje zemlju obrađuje gotovo isključivo upotrebom ručnog alata ili životinjske snage, od Svilja u kojem ljudi i vijesti s jednog na drugi kraj kontinenta putuju tjednima, ako ne i mjesecima, došli smo do trenutka kada su računala pronašla put u mnoga kućanstva, a poljoprivredni strojevi do mnogih polja. U tom uznapredovalom društvu, put u Sveti prestao je biti senzacionalna vijest za novinske naslovnice. Takva vrsta promjene stvorila je od našeg okruženja Svet ovisan o energiji, u prvom redu onoj električnoj. Svet kakav je u vrijeme rođenja Wernera von Siemensa (1818.), izumitelja prvog električnog generatora bilo nemoguće zamisliti i onima najmaštovitijima.

Električna energija danas održava život bolesnika u bolnicama, pogoni automobile, osvjetjava gradove, olakšava svakodnevni život milijardama ljudi diljem Svilja. Život današnjice jest život u potpunosti ovisan o el. energiji, ali to osim brojnih prednosti krije i neke nedostatke i opasnosti. Agenda 21 [1], akcijski plan Ujedinjenih Naroda usmjeren ka održivom razvoju, navodi kako se većinski udio svjetske proizvodnje i potrošnje el. energije odvija na način koji nije održiv u budućnosti uz primjenu postojećih tehnologija. Neke od negativnih posljedica proizvodnje el. energije su zagađenje zraka produktima izgaranja, odlaganje otpada u okoliš i dr.

U takvim okolnostima, biti će potrebno u kontekstu proizvodnje el. energije iz fosilnih goriva poboljšati postojeće tehnologije, ali i povećati udio obnovljivih izvora el. energije. Udio tako proizvedene energije na razini 27 zemalja članica Europske unije u 2010. godini iznosi 12.4% [2]. Te iste godine, na svjetskoj razini u ukupnoj proizvodnji el. energije, 16.1% proizvedeno je iz hidroenergije, a samo 3,3% iz ostalih obnovljivih izvora poput energije vjetra, sunca, biomase i dr [3].

Jedna od temeljnih karakteristika obnovljivih izvora el. energije jest u većoj ili manjoj mjeri *nepredvidivost* i *nestalnost* količine proizvedene el. energije, prvenstveno kao posljedica promjenjivih vremenskih uvjeta. Tako primjerice hidroelektrane ovise o protoku vode, vjetroelektrane o smjeru i brzini vjetra, solarne elektrane o insolaciji¹.

Osim toga, tehnologije obnovljivih izvora energije omogućavaju malim potrošačima ugradnju vlastitih kapaciteta za proizvodnju el. energije poput instalacije solarnih panela na krovove kućanstava i priključak na elektroenergetski sustav. Velik broj ovakvim malih kapaciteta može za tradicionalnu el. mrežu predstavljati velik problem zbog njezine nefleksibilnosti. Naime, tradicionalna mreža je centraliziran sustav, upravljan od strane malog broja upravljačkih entiteta sa nekoliko velikih proizvodnih kapaciteta. Buduća proizvodnja se temelji na praćenju potrošnje iz prošlosti. Povećanje broja malih proizvođača priključenih na takvu mrežu otežava upravljačkim entitetima mogućnost predviđanja potrebnih količina energije, imajući na umu nestalnost obnovljivih izvora energije. Primarni razlog za to je nedostatak dovoljno sofisticiranog sustava upravljanja koji bi nadzirao i u obzir uzimao ovakav sustav velikog broja malih proizvođača s krajnjem nepredvidivom proizvodnjom el. energije.

Rješenje navedenih problema postalo je moguće uvođenjem informacijske i komunikacijske tehnologije u postojeću elektroenergetsku mrežu, što je dovelo do razvoja napredne elektroenergetske mreže (engl. *smart grids*). Napredna elektroenergetska mreža jest električna mreža koja pametno usklađuje i integrira akcije svih dionika priključenih na nju – proizvođača, ali i potrošača, kako bi osigurala učinkovitu, održivu, ekonomski isplativu i sigurnu isporuku el. energije [4]. Ovakva mreža pruža krajnjim korisnicima mogućnost ugradnje naprednih brojila el. energije i bilježenje vlastite potrošnje, što proizvođačima pruža više informacija za kvalitetnije predviđanje potreba za proizvodnjom energije. Dodatno, potrošače se motivira na ugradnju uređaja koji s njihove strane podržavaju smanjenje potrošnje el. energije na zahtjev (DSM, engl. *Demand Side Management*). Ovakvo uravnoteženje proizvodnje i potrošnje za posljedicu ima smanjenje proizvodnje energije umjesto povećanja na strani proizvođača u

¹ Insolacija – količina energije koju Zemlja prima putem sunčevih zraka.

trenutcima vršnog opterećenja. Pritom primarnu motivaciju potrošačima predstavlja prihod koji ostvaruju ovakvim kapacitetima, jer je proizvođačima u takvim slučajevima jeftinije plaćati smanjenje potrošnje nego povećavati vlastitu proizvodnju.

Napredna elektroenergetska mreža više se ne sastoji samo od infrastrukture za prijenos el. energije, već i od infrastrukture koja omogućava prijenos informacija između proizvođača i potrošača bazirane na informacijskim tehnologijama.

Idući korak u razvoju napredne elektroenergetske mreže bazirane na ovakvoj infrastrukturi predstavlja uvođenje veleprodajnog (engl. *wholesale*) i maloprodajnog (engl. *retail*) tržišta električnom energijom. Veleprodajno tržište jest tržište el. energijom lišeno rigidnog središnjeg tijela upravljanja koje određuje fiksnu cijenu, a na kojem se velike energetske kompanije natječu za osiguravanje potrebnih kapaciteta za svoje korisnike. Pritom se cijena energije određuje temeljem zakona ponude i potražnje. Kompanije potom energiju prodaju na maloprodajnom tržištu, gdje kupci biraju između ponuđenih tarifa konkurenčkih elektroenergetskih kompanija na temelju informacija koje im pruža ugrađena informacijska infrastruktura. Osim toga, energetske kompanije imaju mogućnost kupcima ponuditi i nove usluge, poput tarifa s promjenjivom cijenom. Sukladno studijama [5], kupci koji kupuju energiju putem tarifa s promjenjivom cijenom štede u prosjeku 2.32% na svojim računima.

Međutim, neka nedavna negativna iskustva u pokušajima decentralizacije i deregulacije postojećih tržišta (poput krize opskrbe el. energijom u Kaliforniji 2000. godine [6]) su pokazala kako uspjeh takvog oblika tržišta ovisi o dizajnu tržišta, smanjenju potrošnje energije na zahtjev (engl. *demand response*), rezervnim kapacitetima, upravljanju finansijskim rizicima i pouzdanom upravljanju lancem opskrbe el. energije [7].

Upravo kako bi se identificirali i izbjegli slični problemi na tržištima el. energije budućnosti, pristupilo se modeliranju i simuliranju tržišnih uvjeta prisutnih u naprednim elektroenergetskim mrežama u računalnom okruženju [8]. Takva istraživanja će rezultirati vrijednim podacima, identificirati probleme ali i pružiti

nove poglede na tržišta el. energijom što će omogućiti regulatornim agencijama uvođenje novih pravila i modela za otvorena tržišta el. energijom.

Međutim, takva istraživanja su se pokazala suviše složenima za modeliranje klasičnim teorijama igara [9]. Istraživanja su potom krenula u smjeru simulacija baziranim na agentima², ali ove studije [10] nisu bile u mogućnosti istražiti čitav spektar neočekivanog ponašanja pojedinih participirajućih strana.

U tom kontekstu razvijen je Power TAC [9] (*Power Trading Agent Competition*), otvorena natjecateljska simulacijska platforma koja simulira otvoreno i slobodno³ tržište el. energijom, a sve s ciljem istraživanja strukture i ponašanja na tržištima el. energijom. Power TAC jest primjer TAC (*Trading Agent Competition*) natjecanja u primjeni na tržište el. energijom. Ova metoda se ranije uspješno primjenjivala na druga slična istraživanja [11][12].

Primarni zadatak Power TAC natjecanja jest ponuditi potpun pregled mogućnosti i ograničenja otvorenih tržišta el. energijom, te identificirati dobre prakse i ukazati na potrebne izmjene zakonodavne legislative kako bi upravljanje elektroenergetskim mrežama budućnosti bilo učinkovito i lišeno problema. Temeljna figura takvog natjecanja jest programski agent, koji služi kao posrednik između veleprodajnog tržišta na kojem su prisutni veliki proizvođači el. energije i maloprodajnog tržišta na kojem su potrošači el. energije.

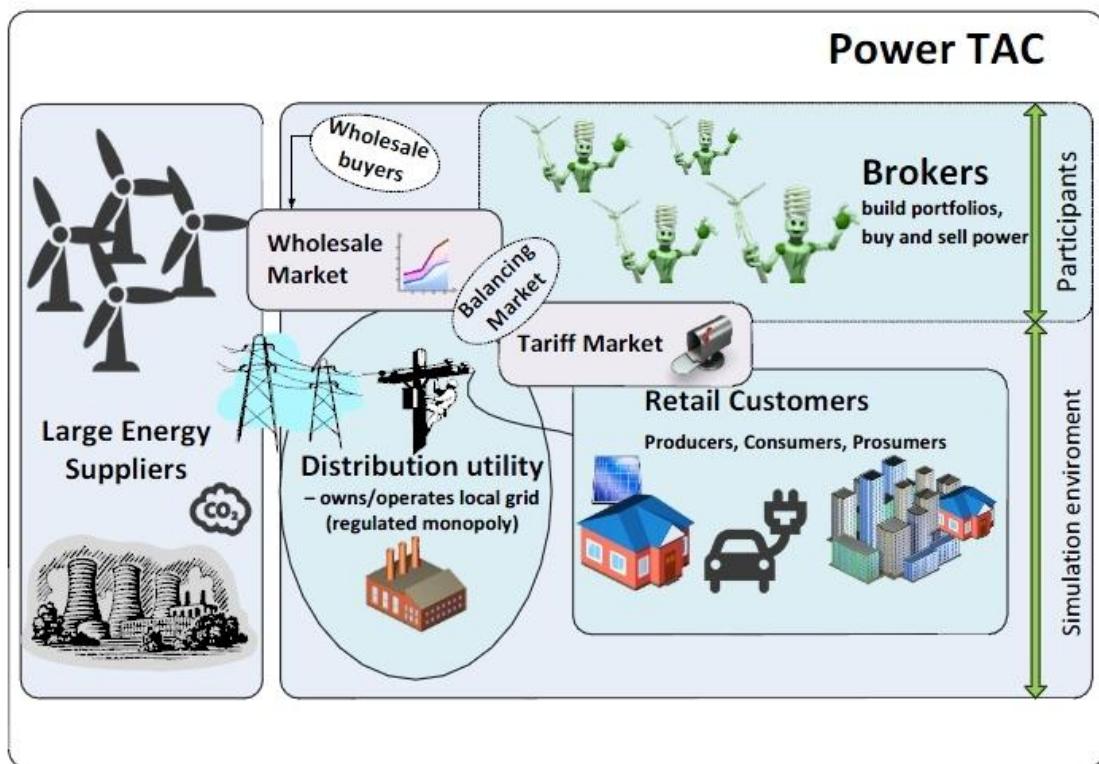
² Programski agent – računalni program koji samostalno obavlja skup poslova zadanih od strane korisnika, odnosno vlasnika

³ Slobodno tržište je tržište gdje je cijena nekog proizvoda ili usluge određena ponudom i potražnjom, što je u suprotnosti s kontroliranim tržištima gdje je cijena strogo kontrolirana ili određena.

1.2. Opis simulacijske okoline

Power TAC natjecateljska okolina simulira tipične sudionike jednog lanca koji obuhvaća proizvodnju, distribuciju i potrošnju el. energije. Sama simulacijska okolina je prikazana na slici 1, a pojedini čimbenici te okoline su opisani u nastavku.

Središnja figura natjecanja jest programski agent, kojeg u slučaju Power TAC natjecanja nazivamo još i brokerom. Brokeri energiju kupuju od velikih proizvođača ili drugih brokera na veleprodajnom tržištu, a prodaju potrošačima putem tarifa na maloprodajnom tržištu. Broker se pritom vodi interesom stvaranja što većeg profita, a količina akumuliranog profita je ujedno i kriterij pobjede na pojedinom natjecanju. Konkurenčija pojedinom brokeru su brokeri drugih timova.



Slika 1: Izgled Power TAC simuliranog okruženja

Potrošači su predstavljeni kućanstvima, industrijskim pogonima, uredima, ali i vlasnicima električnih vozila. Dodatno, potrošači mogu na svoje objekte ugraditi solarne panele ili vjetroelektrane, te proizvoditi električnu energiju čime postaju i proizvođači, a svoju energiju potom prodaju brokeru.

Pojedini brokeri maloprodajnom tržištu nude tarife koje uključuju ugovorene uvjete. Tarife mogu uključivati fiksnu ili dinamičku cijenu el. energije, kao i druge povlastice i kazne (poput kazne za prerani raskid ugovora). Maloprodajni kupci uzimaju u obzir sve ponuđene tarife te se odlučuju za one povoljnije, iako ponašanje samih potrošača nije uvijek potpuno racionalno.

Potrošnja pojedinog potrošača ovisi o tipu (kućanstvo primjerice troši više el. energije u neradne sate, za razliku od uredskih objekata koji više energije troše u radno vrijeme). Navedeno spada u statičke faktore neovisne o razvoju pojedine simulacije. Dodatno, potrošnja pojedinog kućanstva može ovisiti i o agentu, odnosno odabranoj tarifi te uravnoteženju potrošnje. Potrošnja pojedinog potrošača dodatno ovisi o nasumičnom ponašanju te o vremenskim uvjetima.

Osim prije spomenutih malih proizvođača koji el. energiju proizvode u sklopu svoga doma putem primjerice solarnih panela, većinu el. energije tržištu osiguravaju veleprodajni proizvođači el. energije, odnosno - elektrane (engl. *GenCos – Generating Companies*). Svoju proizvodnju veliki proizvođači nude na veleprodajnom tržištu. Međutim, tamo je energiju moguće kupiti i od drugih brokera. Energiju na veleprodajnom tržištu je moguće kupiti za do 24 sata unaprijed.

Ponašanje pojedinih proizvođača je moguće manje ili više predvidjeti. Predvidljivi proizvođači su oni s veleprodajnog tržišta, a među manje predvidljive pripadaju mali proizvođači koji energiju proizvode metodama obnovljivih izvora energije. Manja predvidljivost količine proizvedene el. energije je posljedica promjenjivih vremenskih uvjeta, poput brzine i smjera vjetra u slučaju vjetroelektrana.

Iz navedenoga je vidljivo da je za uspješno poslovanje brokera važno sudjelovati na dva tržišta – maloprodajnom i veleprodajnom. Ugrubo se svaki

vremenski odsječak⁴ može podijeliti u dvije faze – fazu planiranja i fazu rada. Tijekom faze planiranja broker stvara portfelj potrošača, proizvođača i korisnika el. vozila, nudeći tarife i individualne ugovore. U fazi rada, broker mora uravnotežiti razlike u potrošnji svojih kupaca i proizvodnji svojih proizvođača.

Navedeno uravnoteženje je moguće postići kupovinom energije na veleprodajnom tržištu od drugih proizvođača, proizvodnjom i otkupljivanjem iz vlastitih ugovorenih kapaciteta koje pružaju mali proizvođači, kupovinom sustava pohrane (primjerice vlasnici električnih vozila mogu iznajmiti baterije svog vozila u trenucima dok je vozilo spojeno na mrežu), ili pomoću napredne elektroenergetske mreže isključiti na neki kraći vremenski interval pojedine uređaje u nekom kućanstvu (primjerice ugasiti perilicu rublja na 15 minuta što će smanjiti opterećenje mreže, a korisnik neće previše osjetiti), ukoliko je ugovorena tarifa dopuštala ovaku mogućnost.

Broker koji navedenim mogućnostima nije u stanju namiriti potrebe svojih proizvođača (što primjerice može biti rezultat loše procjene buduće potrošnje), a zbog činjenice da je prekid opskrbe el. energijom nedopustiv, opskrbu potrebnim količinama el. energije može osigurati entitet koji nadzire distribucijsku mrežu (engl. *Distribution Utility* – DU). Međutim, takav proces se odvija po mnogo nepovoljnijim cijenama po brokera, što u konačnici ima negativan utjecaj na finansijski rezultat tog brokera, a samim time rezultira i lošijim plasmanom na natjecanju.

Zbog svega navedenog vidljivo je da je od presudne važnosti za uspješno funkcioniranje brokera i konačni plasman mogućnost predviđanja potrošnje i cijena s obzirom na doba dana, portfelj korisnika i vremenske uvjete.

U svakom trenutku brokeru je na raspolaganju informacija o trenutnim vremenskim uvjetima na području koje obuhvaća simulacija, te vremenska prognoza za to područje. Vremenska prognoza koja se komunicira brokeru obuhvaća 24 iduća sata. Brokeru je ovo višestruko važan podatak – neki korisnički modeli, poput kućanstava, će prilagoditi potrošnju energije ovisno o temperaturi (primjerice uključivanjem klima uređaja u slučaju viših temperatura) ili promjena u proizvodnji (solarne elektrane proizvode manje energije u slučaju veće naoblake).

⁴ Vremenski odsječak – trenutak simuliranog vremena u pojedinom natjecanju

Broker ima mogućnost upravljati potrošnjom i ostvarivati profit i putem tarifa koje omogućavaju dinamičku izmjenu cijene prodavane energije. Cijene se dojavljuju klijentima nekoliko vremenskih odsječaka unaprijed, što odmah dovodi do zaključka kako broker mora imati mogućnost predviđati i postavljati cijene koje će po njega biti profitabilne u tom budućem trenutku. Svaki agent u ovom postupku u obzir treba uzeti nekoliko faktora – fiksne cijene nabavke električne energije (cijene s veleprodajnog tržišta na kojem broker također kupuje unaprijed), potrebnu količinu el. energije, promjenu vremenskih uvjeta i dr.

U konačnici pobjeđuje onaj broker koji na kraju natjecanja ima najveći saldo. Postoji mogućnost u nekom od budućih natjecanja uvođenja dodatnih kriterija vrednovanja uspjeha svakog agenta, poput dodatnog pozitivnog vrednovanja uspjeha agenta ako energiju kupuje iz obnovljivih izvora ili kažnjavanja onih koji je kupuju iz neekoloških izvora.

Vremenski, svaka simulacija je podijeljena na vremenske odsječke (engl. *timeslots*). Pojedini vremenski odsječak simulira period od jednog sata, a traje 5 sekundi stvarnog vremena. Trajanje tipične simulacije iznosi približno 2 sata, odnosno 60 simuliranih dana, što podijeljeno na vremenske odsječke iznosi 1440 istih. Simulacija završava nasumični broj dana nakon isteka navedenih 60 dana, što je mjera osiguranja kako fiksan i poznati kraj simulacije ne bi negativno iskoristili pojedini agenti.

Samo natjecanje se tipično odvija u više etapa (kvalifikacije, četvrtfinala, polufinala, finala) s različitim brojem participirajućih brokera (igre s 2, 4, 8 igrača).

2. CrocodileAgent

Jedan od natjecatelja na prethodno održanom Power TAC 2012 natjecanju jest i CrocodileAgent 2012 [13], inteligentni programski agent razvijen od strane tima Sveučilišta u Zagrebu.

U svojoj osnovi, CrocodileAgent se može podijeliti na tri osnovna modula koja međusobno komuniciraju:

- 1) upravitelj tarifama (engl. *tariff manager*)
- 2) upravitelj tržištem (engl. *market manager*)
- 3) upravitelj predviđanjima (engl. *forecast manager*)

Modularni dizajn agenta je česta praksa u okruženju TAC natjecanja [14][15].

Agent u svakom vremenskom odsječku komunicira sa veleprodajnim i maloprodajnim tržištima el. energije te prima podatke o proizvodnji i potrošnji energije, prognozi vremena, obračunskoj cijeni el. energije (engl. *clearing price*). Ti podaci se potom komuniciraju upravitelju predviđanjima.

Temeljni zadatak upravitelja predviđanjima jest predviđanje potrošnje el. energije od strane kupaca za naredna 24 vremenska odsječka kako bi agent u tim odsječcima osigurao dovoljnu količinu el. energije. Navedeni broj vremenskih odsječaka je ujedno maksimalni broj budućih odsječaka za koje agent može kupiti el. energiju.

Analiza ponašajnih modela pojedinih tipova kupaca poput kućanstava i ureda je rezultirala s nekoliko profila temeljenih na tipičnim obrascima potrošnje energije za taj tip. Primjerice, utvrđeno je kako tipičan period potrošnje el. energije u kućanstvima iznosi 24 sata, gdje je potrošnja raste kako dan odmiče - manja ujutro, veća u večernjim satima, a najmanja noću. S druge strane, period potrošnje ureda traje 168 sati (dakle cijeli tjedan). Potrošnja je mnogo veća tijekom radnog dijela tjedna i u radnim satima (08:00-16:00) nego u neradno vrijeme ili tijekom vikenda.

Ovakvu periodičnost s druge strane nije bilo moguće pronaći kod vjetroelektrana ili solarnih elektrana, jer prve ovise o smjeru i brzini vjetra, a potonje o mjeri u kojoj oblaci zakrivaju Sunce, što ovisi o trenutnim vremenskim uvjetima za neko područje koji nemaju tipičnu periodičnost ponašanja. Sukladno tome, nije moguće praćenjem perioda implementirati strategiju predviđanja proizvodnje iz obnovljivih izvora el. energije.

CrocodileAgent potrošnju svojih kupaca izračuna uz pomoć tehnike trostrukog eksponencijalnog zaglađivanja, poznatije i kao Holt-Winters [16][17], uzimajući u obzir prethodno prepoznate profile, te potrošnju baziranu na podacima prikupljenima u upravitelju tarifama.

Kupovina potrebnih količina energije su osnovna zadaća upravitelja tržištem. Na temelju predviđanja i povijesne potrošnje, upravitelj kupuje (ako je predviđeno povećanje potrošnje) ili prodaje (ako je predviđeno smanjenje potrošnje) el. energiju za razdoblje od iduća 24 sata.

Uloga modula za upravljanje tarifama jest nuditi potrošačima na maloprodajnom tržištu energiju putem tarifa. Ključne funkcije modula su kreiranje tarifa, stavljanje u ponudu novih tarifa, zamjena ili povlačenje postojećih tarifa. Pritom se povučene tarife pohranjuju u interni repozitorij agenta, što omogućava modulu za upravljanje tarifama uspoređivanje novokreiranih tarifa s onima koje su tržištu bile ponuđene ranije.

3. Postojeća rješenja za obradu podataka

Power TAC simulacijska platforma kao rezultat pojedinog natjecanja (simulacije) pruža velik broj korisnih informacija čije praćenje može dati nove spoznaje o ponašanju agenata ili Power TAC simulacijske platforme. Takvi podaci su ključ za razumijevanje događaja unutar simuliranog okruženja te donošenje zaključaka koji će poslužiti za modeliranje stvarnih tržišta el. energijom. Osim toga, navedene informacije služe timovima koji razvijaju pojedine agente u svrhu poboljšanja svojstava i ponašanja vlastitih agenata u okviru simulacije.

Podatke možemo podijeliti na:

- podatke dostupne u toku simulacije
- podatke dostupne u obliku dnevničkih datoteka nakon simulacije

Podaci dostupni nakon simulacije se dodatno mogu podijeliti na dnevničke datoteke stanja (engl. *state-log files*) koji sadrže podatke o stanju pojedine simulacije i podatke traga praćenja rada aplikacije (engl. *trace-log files*). Potonji su zanimljivi isključivo u kontekstu praćenja mogućih grešaka u programskoj izvedbi rada simulacijskog okruženja.

Iz svega navedenog, vidljivo je kako je analiza tih podataka u toku simulacije, ali i nakon simulacije od presudne važnosti za uspjeh cijelog projekta, stoga su razvijena dva alata koja omogućavaju analizu podataka.

3.1. Power TAC Visualizer

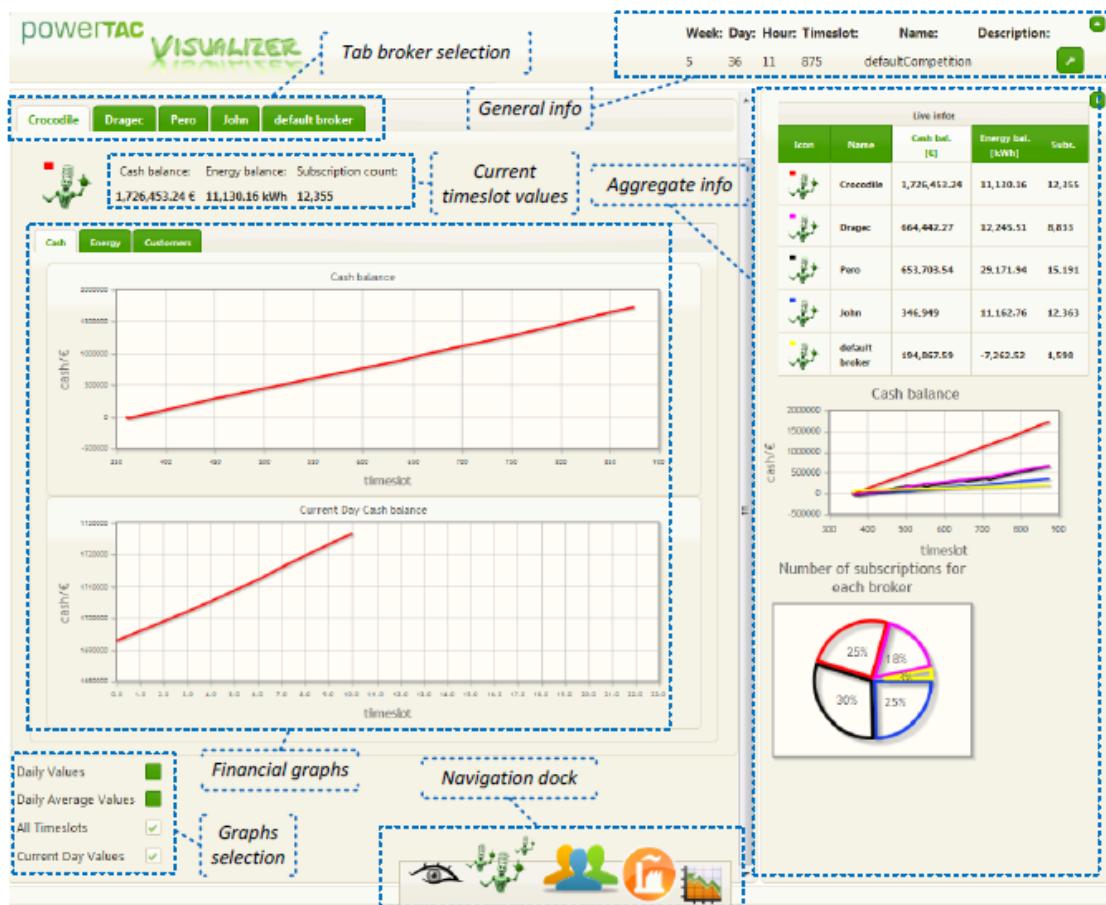
Jedna od temeljnih komponenti Power TAC simulacijske platforme jest i *Power TAC Visualizer* [18], alat koji omogućava vizualno nadgledanje natjecanja, a koji razvija tim u sklopu Sveučilišta u Zagrebu.

Primarna uloga alata jest omogućiti stvarno-vremensko (u toku odvijanja natjecanja) nadziranje pojedinog Power TAC natjecanja te korisnicima pružiti napredne mogućnosti analize podataka putem intuitivnog korisničkog sučelja.

Dodatna uloga *Visualizer-a* kao web aplikacije jest pružanje infrastrukture za upravljanje pojedinim natjecanjem. Ova funkcionalnost omogućava

natjecateljima podešavanje velikog broja parametara i promatranje kako se simulacija odvija u skladu s tim izmijenjenim parametrima.

Razvijeno rješenje je od velike znanstvene vrijednosti jer pruža mogućnost razumijevanja dinamike tržišta el. energije kroz identifikaciju, analizu i prezentaciju dionika, procesa i ključnih interakcija u okružju naprednih energetskih mreža. Slika 2 prikazuje grafičko korisničko sučelje aplikacije *Power TAC Visualizer*.



Slika 2: Grafičko korisničko sučelje aplikacije *Power TAC Visualizer*

3.2. Power TAC Logfile Analysis (PLA)

Za potrebe analize dnevničkih datoteka razvijen je alat koji omogućava pohranu i dohvata podataka koji se nalaze u takvim datotekama u posebno razvijenu bazu podataka. Alatu je pridruženo ime *Power TAC Logfile Analysis* [19].

Temeljni ciljevi projekta:

- omogućiti jednostavan i brz pristup do svih službenih rezultata pojedinih natjecanja i simulacija iz prošlosti, bez potrebe za pregledavanjem dnevničkih datoteka
- omogućiti jednostavan pristup svim zainteresiranim strana uključenim u razvoj Power TAC simulacijske okoline
- pružiti infrastrukturu za jednostavan grafički prikaz podataka unutar relacijske baze podataka

Sustav je izgrađen na temelju MySQL relacijske baze podataka, a sastoji se od virtualnih relacija⁵ koje predstavljaju sučelje prema korisnicima. Sami podaci su sadržani u tablicama koje nisu direktno dostupne korisnicima, već isključivo posredno preko virtualnih relacija.

Međutim, trenutno povrh baze podataka nije razvijen alat za grafički prikaz i analizu tog velikog broja korisnih podataka, već autori preporučuju korištenje SQL upita i ostvarenje grafičkog prikaza u drugim alatima poput Microsoft Excela.

Tim zadužen za razvoj PLA alata je također ponudio ostalim dionicima koji sudjeluju razvoju Power TAC simulacijskog okruženja korištenje trenutno pokrenute baze podataka, ali i odgovarajuću podršku za pokretanje vlastite baze podataka temeljne na PLA izvedbi rješenja.

⁵ Virtualne relacije – relacija kojoj su shema i sadržaj definirani izrazom relacijske algebre čiji su operandi temeljne ili virtualne relacije

4. Web-modul za analizu događaja na tržištima električnom energijom

Prethodna poglavlja pokazala su nedostatak adekvatnog alata koji bi omogućio brzu, jednostavnu i efikasnu analizu podataka zapisanih u dnevničkim datotekama koje su rezultat simulacije Power TAC simulacijske okoline, unatoč činjenici kako se radi o skupu vrlo vrijednih podataka.

Unosi u takvoj datoteci predstavljaju pozive metoda koji modificiraju stanje simulacije, primjerice – specifikacije tarifa, promjene salda pojedinih brokera, potrošnju električne energije i druge vrijedne podatke.

Prikaz i analiza takvih podataka predstavlja povijesnu analizu (engl. *offline analysis*). Glavni zadatak povijesne analize je korisniku grafički prikazati događaje unutar pojedine simulacije.

Dodatno, tako velik skup podataka može poslužiti i kao baza znanja za agenta u toku simulacije, te takvo rješenje predstavlja stvarno-vremensku analizu (engl. *online analysis*). Dodatan izvor znanja agentu može omogućiti bolje ponašanje u natjecanjima.

Konačni cilj razvoja web-modula za analizu događaja na tržištima električnom energijom jest omogućiti povijesnu i stvarno-vremensku analizu dnevničkih datoteka stanja. Modulu je pridijeljeno ime **YAPLA** – Yet Another Powertac Llogfile Analysis, a za potrebe dohvaćanja podataka koristit će se baza podataka razvijena u sklopu projekta PLA.

4.1. Korisnički zahtjevi

Korisnički zahtjevi za aplikaciju YAPLA se mogu ugrubo podijeliti u dvije osnovne kategorije, s obzirom na vremenski kontekst analize:

- Povijesna analiza događaja u okružju simulacijske platforme PowerTAC, odnosno analiza podataka dnevničkih datoteka stanja, grafički i drugi prikaz u pretraživaču web-a.
- Stvarno-vremenska analiza događaja u okružju simulacijske platforme PowerTAC, odnosno omogućavanje pristupa sustavu YAPLA brokeru CrocodileAgent putem REST sučelja i izvršavanje analiza unutar sustava

YAPLA. Rezultat analize čine nove informacije agentu, te shodno tome i očekivano bolje ponašanje agenta unutar simulacije.

Pojednostavljeno, cilj je omogućiti brokeru CrocodileAgent da na temelju stvarno-vremenske analize donosi kvalitetnije odluke na natjecanju, primjerice – analizom kretanja cijena električne energije s obzirom na trenutne vremenske prilike i prognoze.

4.2. Funkcionalni zahtjevi

S obzirom na korisničke zahtjeve aplikaciju možemo razložiti na dva osnova dijela:

- povjesna analiza
- stvarno-vremenska analiza

Na jednak način će biti podijeljeni moduli aplikacije.

Otvaranjem glavnog izbornika aplikacije, moguće je odabrati povjesnu analizu ili postavke stvarno-vremenske analize. Odabirom povjesne analize korisnik ima mogućnost putem čarobnjaka u nekoliko koraka odabrati graf koji želi prikazati. Čarobnjak ima mogućnost povratka unazad u svakom koraku. Dodatno, korisničko sučelje treba korisniku pružati informacije o eventualnim greškama koje su se dogodile tijekom rada.

Odabrani graf se po završetku postupka odabira putem čarobnjaka prikazuje korisniku na ekranu web preglednika. Graf se pritom prikazuje u cijelosti, a moguće je uvećati pojedine segmente radi bolje preglednosti. Postavljanjem pokazivača miša iznad grafa moguće je očitati određeni par vrijednosti. Pritiskom na tipku „reset“, poništava se povećanje prikaza.

Odabirom postavki stvarno-vremenske analize, korisniku se nudi prikaz stanja sustava za predviđanje parametara, odnosno popis igara s kojima je sustav trenutno treniran. Pritiskom na tipku moguće je pokrenuti treniranje, a o napretku se korisnik izvještava putem ekrana. Korisnik se također izvještava o eventualnim problemima u radu sa stvarno-vremenskom analizom.

CrocodileAgent mora imati mogućnost stvarno-vremensku analizu vršiti korištenjem vanjskog sučelja spajanjem na javno dostupan servis. Pritom će

CrocodileAgent sustavu YAPLA isporučiti podatke potrebne za analizu, a sustav YAPLA agentu vratiti rezultate analize.

4.3. Nefunkcionalni zahtjevi

S obzirom na činjenicu kako je trajanje pojedinog vremenskog odsječka samo pet sekundi, stvarno-vremenska analiza se mora odvijati velikom brzinom, odnosno unutar vremena kraćeg od duljine trajanja jednog vremenskog odsječka.

Kao web-modul, aplikacija mora biti dostupna putem Interneta, te sukladno tome mora biti postavljena na odgovarajući poslužitelj.

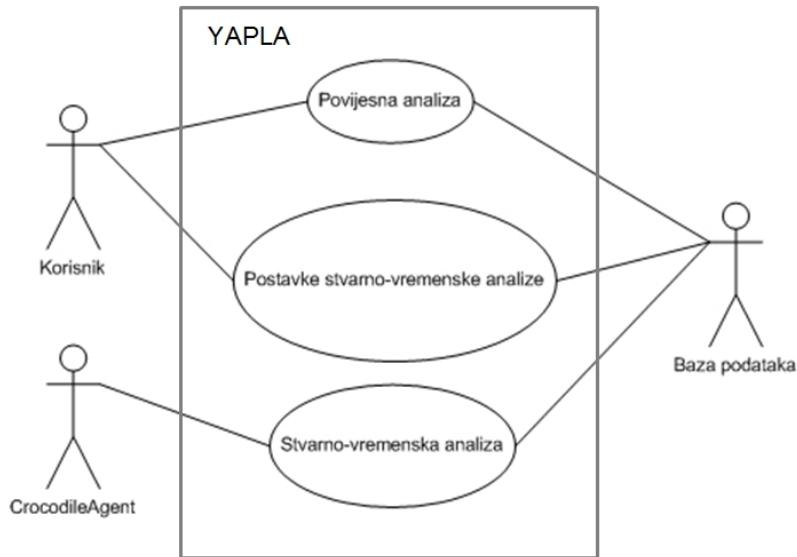
Sigurnost nije od velikog značaja za rad ove aplikacije, s obzirom da je onemogućena bilo kakva interakcija između korisnika i aplikacije koja bi mogla rezultirati narušavanjem integriteta podataka unutar baze podataka.

4.4. Slučajevi korištenja

Kao što je već navedeno, glavna uloga alata YAPLA jest grafički prikaz podataka pohranjenih u dnevničkim datotekama te pružanje dodatnih informacija agentu CrocodileAgent. Slika 3 prikazuje osnovne korisnike u radu aplikacije, te osnovne zadaće koje aplikacija mora ispuniti.

Uloge u tom smislu su:

1. Korisnik jest osoba koja želi pregledavati grafički prikaz iz dnevničkih datoteka u sklopu povijesne analize podataka, te osoba koja upravlja ponašanjem stvarno-vremenske analize. Sa sustavom komunicira putem grafičkog korisničkog sučelja.
2. CrocodileAgent jest agent koji koristi stvarno-vremensku analizu za dobivanje novih informacija, a komunicira sustavom putem vanjskog sučelja.
3. Baza podataka jest vanjska baza podataka u koju su pohranjene dnevničke datoteke stanja koje su rezultat izvođenja simulacija u prošlosti na Power TAC simulacijskoj platformi.



Slika 3: Grafički prikaz slučajeva korištenja – YAPLA

4.4.1. Slučaj korištenja (1): Grafički prikaz podataka

Tablica 1. Slučaj korištenja – povijesna analiza

Naslov:	Povijesna analiza:
Opis:	Korisnik putem pretraživača web-a pristupa YAPLA sustavu, odabire podatke koje želi pregledavati i pregledava podatke.
Korisnik:	Korisnik
Preduvjeti:	Aplikacija YAPLA je pokrenuta, poslužitelj aplikacije YAPLA ima pristup Internetu, korisnik ima pristup Internetu, u bazi podataka postoje unosi iz dnevničkih datoteka.
Rezultat:	Preglednik web-a korisniku prikazuje željeni grafički prikaz.
Glavni scenarij:	<ol style="list-style-type: none"> 1.Korisnik otvara YAPLA aplikaciju putem web preglednika. 2.Korisnik u glavnom izborniku odabire opciju za povjesnu analizu. 3.Korisnik odabire podatke koje želi grafički prikazati 4.Korisnikov web preglednik prikazuje grafički prikaz željenih podataka.
Alternativni scenarij:	3a.Nema podataka za prikaz – korisniku se dojavljuje porukom unutar web preglednika kako nema podataka za prikaz

4.4.2. Slučaj korištenja (2): Postavke stvarno-vremenske analize

Tablica 2. Slučaj korištenja – postavke stvarno-vremenske analize

Naslov:	Postavke stvarno-vremenske analize:
Opis:	Korisnik putem pretraživača web-a pristupa YAPLA sustavu, te uređuje postavke stvarno-vremenske analize.
Korisnik:	Korisnik
Preduvjeti:	Aplikacija YAPLA je pokrenuta, poslužitelj aplikacije YAPLA ima pristup Internetu, korisnik ima pristup Internetu, u bazi podataka postoje unosi iz dnevničkih datoteka.
Glavni scenarij:	<ol style="list-style-type: none">1.Korisnik otvara YAPLA aplikaciju putem web preglednika.2.Korisnik u glavnom izborniku odabire opciju za povjesnu analizu.3.Korisnik pokreće treniranje stvarno-vremenske analize4.Preglednik web-a korisniku prikazuje postotak napretka.
Alternativni scenarij:	

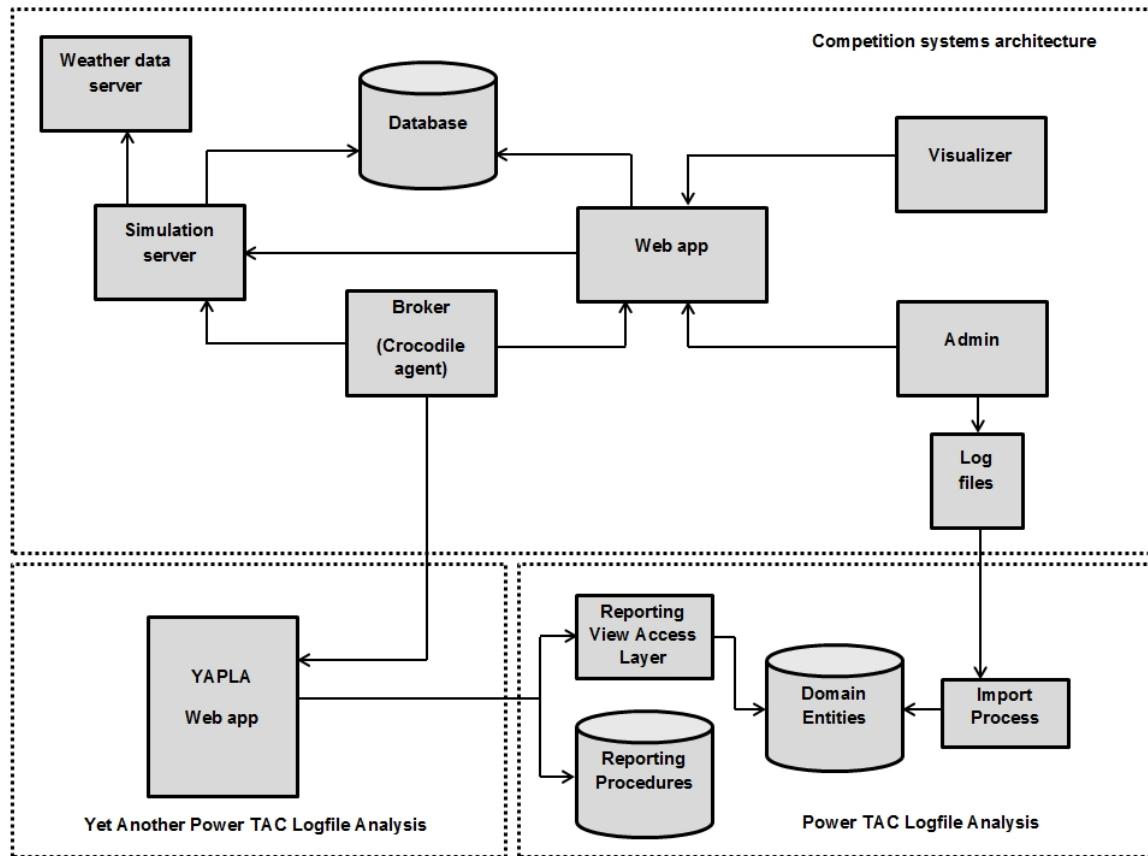
4.4.3. Slučaj korištenja (3): Dohvat podataka

Tablica 3. Slučaj korištenja – stvarno-vremenska analiza

Naslov:	Stvarno-vremenska analiza:
Opis:	CrocodileAgent zahtjeva nove podatke u trenutku dok sudjeluje u simulaciji / natjecanju u sklopu Power TAC simulacijske platforme.
Korisnik:	CrocodileAgent
Preduvjeti:	Aplikacija YAPLA je pokrenuta, poslužitelj aplikacije YAPLA ima pristup Internetu, CrocodileAgent ima pristup Internetu, u bazi podataka postoje unosi iz dnevničkih datoteka, CrocodileAgent može isporučiti podatke potrebne za stvarno-vremensku analizu.
Glavni scenarij:	<ol style="list-style-type: none">1.CrocodileAgent šalje zahtjev za analizom podataka zajedno s potrebnim podacima za obavljanje analize2.Modul za stvarno-vremensku analizu radi analizu podataka3.Modul za stvarno-vremensku analizu po završetku izračuna vraća CrocodileAgentu izračunate vrijednosti
Alternativni scenarij:	<ol style="list-style-type: none">2a.Treniranje mreže nikad nije pokrenuto pa nije moguće obaviti analizu, dojavljuje se CrocodileAgentu

4.5. Arhitektura sustava YAPLA

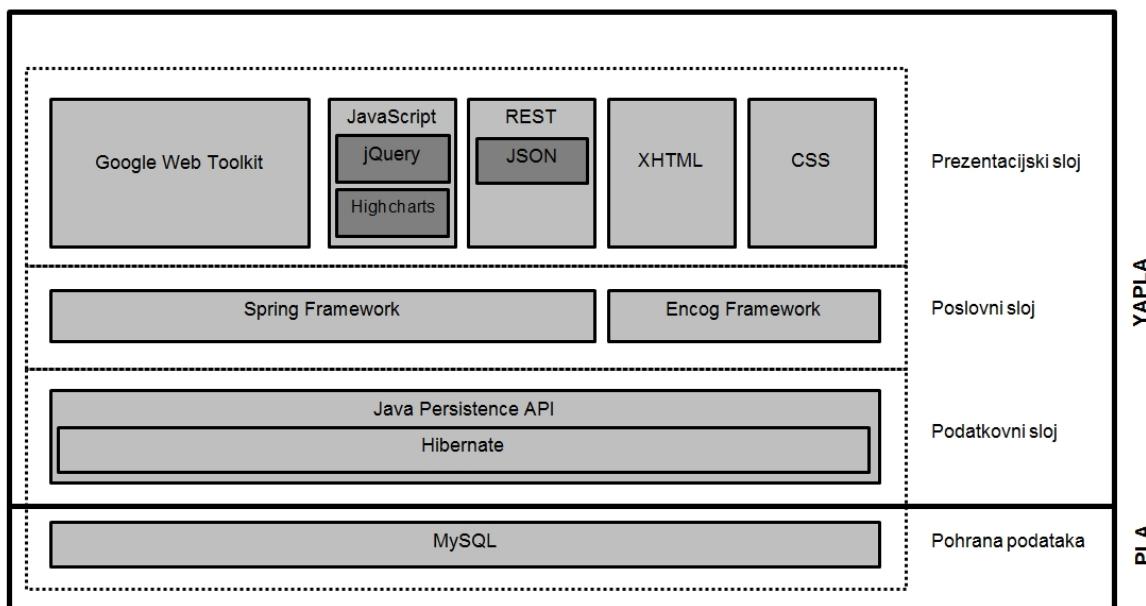
Na slici 4 vidljiv je odnos sustava s obzirom na postojeće komponente Power TAC okoline. Aplikacija YAPLA koristi vanjsku bazu podataka sustava PLA. Baza podataka sustava PLA se puni dnevničkim datotekama Power TAC simulacijskog okruženja. Broker CrocodileAgent ima mogućnost pristupa analizama YAPLA sustava putem REST sučelja.



Slika 4: Izgled rješenja s obzirom na postojeće sustave

4.6. Tehnologije, programske knjižnice i alati

Za razvoj traženog programskog rješenja odabran je jezik *Java*. Dodatno, tu je i niz tehnologija otvorenog koda koje će biti upotrijebljene s ciljem razvoja skalabilnog i robusnog programskog rješenja, a koje će omogućiti daljnju nadogradnju prema potrebi. Sve važnije tehnologije su prikazane na slici 5.



Slika 5: Tehnologije upotrijebljene za razvoj aplikacije

Sve upotrijebljene knjižnice i tehnologije mogu se sukladno licenci koristiti u obrazovne svrhe bez naknade.

4.6.1. Tehnologije

Za razvoj klijentskog okruženja odabran je *Google Web Toolkit*⁶, otvoreni skup alata koji omogućava jednostavan razvoj i održavanje kompleksnih AJAX web aplikacija. Sama aplikacija se primarno razvija u programskom jeziku *Java*, a može se podijeliti u tri dijela – klijentski dio, poslužiteljski dio i dijeljeni dio. Prilikom kompilacije rješenja, klijentski dio se prevodi u *JavaScript* i izvodi u internetskom pregledniku računala korisnika. Poslužiteljski dio se prevodi u *Java* izvršne

⁶ Google Web Toolkit (<https://developers.google.com/web-toolkit/>)

datoteke i izvodi na nekom odabranom poslužitelju, a dijeljeni dio koriste i klijent i poslužitelj te se prevodi u *JavaScript* i *Java* izvršne datoteke.

Iako je uz pomoć *Google Web Toolkit*-a moguće razviti cijelokupno programsko rješenje (i klijentsku i poslužiteljsku stranu) web aplikacije, kao dodatna tehnologija za poslužiteljsku stranu aplikacije odabran je *Spring framework*⁷. *Spring framework* jest skup knjižnica otvorenog koda koje omogućavaju jednostavniji i brži razvoj, te testiranje i nadogradnju web aplikacija.

S obzirom na to da aplikacija treba pristupati bazi podataka, bilo je nužno odlučiti se za *ORM framework* (*Object-relational mapping*), odnosno programsku knjižnicu koja će omogućiti povezivanje relacijske baze podataka i objektnog modela kakav je korišten prilikom razvoja aplikacije. Za tu svrhu odabran je *Hibernate*⁸. U programskom rješenju također će biti korištena u *Hibernate* ugrađena implementacija *JPA*⁹ specifikacije, što omogućava jednostavnu zamjenu tehnologije *Hibernate* tehnologijama kompatibilnima s *JPA* specifikacijom ako se *Hibernate* iz bilo kojeg razloga pokaže neadekvatnim rješenjem u budućnosti.

Za pohranu podataka se brine *MySQL*¹⁰ relacijska baza podataka otvorenog koda koja je odabrana zbog kompatibilnosti aplikacije s vanjskim zavisnostima (sustavom PLA). *MySQL* je u svjetskim razmjerima najpopularniji izbor za bazu podataka otvorenog koda, a u stalnom razvoju je već više od desetljeća.

Modul za stvarno-vremensku analizu koristi *Encog*¹¹ framework, skup knjižnica koje pružaju podršku za rad s neuronским mrežama i umjetnom inteligencijom. *Encog* pruža korisniku velik broj različitih tipova neuronskih mreža kao i metoda treniranja. Velika vrlina knjižnice je i brzina rada, što je ponajprije zasluga višedretvenog rada knjižnice na računalima sa više procesorskih jezgri, činjenica koja u svijetu umjetne inteligencije i aplikacijama od kojih se zahtjeva velika brzina rada, nije zanemariva.

⁷ Spring framework (<http://www.springsource.org/>)

⁸ Hibernate (<http://www.hibernate.org/>)

⁹ JPA – Java Persistence API (<http://www.oracle.com/technetwork/java/javase/tech/persistence-jsp-140049.html>)

¹⁰ MySQL (<http://www.mysql.com/>)

¹¹ Encog Framework (<http://www.heatonresearch.com/encog>)

Za potrebne povijesne analize, odnosno funkcionalnost grafičkog prikaza podataka o pojedinim simulacijama, odabrana je knjižnica *Highcharts*¹² koja pruža širok spektar različitih vrsta grafova, zadovoljavajući brzinu rada s velikim količinama podataka, te je kompatibilna s knjižnicom *Google Web Toolkit* putem odgovarajuće knjižnice.

4.6.2. Alati i razvojno okružje

Za automatiziranu izgradnju, dohvaćanje odgovarajućih programskih knjižnica i testiranje, upotrijebljen je alata *Apache Maven*¹³. Razvoj same aplikacije se odvija unutar razvojnog okruženja *Eclipse*¹⁴ u svojoj posljednjoj inačici *Juno*, alat koji koristi velik broj programskih inženjera za razvoj koda baziranog na programskom jeziku Java. Osim toga *Eclipse* nudi odgovarajuću nadogradnju za integraciju alata *Maven*. Sama aplikacija se izvodi putem web poslužitelja *Apache Tomcat*¹⁵.

4.6.3. Dodatne programske knjižnice

Dodatno, osim prije navedenih programskih knjižnica, tu je i niz manjih knjižnica koje omogućavaju integraciju ovako velikog broja tehnologija.

Integracija *Google Web Toolkit framework* i *Spring framework* moguća je uz pomoć knjižnice *Spring4GWT*¹⁶. Ugradnja i upotreba knjižnice *Highcharts* u *GWT* moguća je uz pomoć knjižnice *GWT Highcharts*¹⁷. Također, *Highcharts* knjižnica za svoje izvođenje zahtjeva odgovarajuću *JavaScript* knjižnicu, za što je odabran *jQuery*¹⁸.

¹² Highcharts (<http://www.highcharts.com/>)

¹³ Apache Maven (<http://maven.apache.org/>)

¹⁴ Eclipse Juno (<http://www.eclipse.org/>)

¹⁵ Apache Tomcat (<http://tomcat.apache.org/>)

¹⁶ Spring4GWT (<http://code.google.com/p/spring4gwt/>)

¹⁷ GWT Highcharts (<http://www.moxiegroup.com/moxieapps/gwt-highcharts/>)

¹⁸ jQuery (<http://jquery.com/>)

4.7. Rezultati

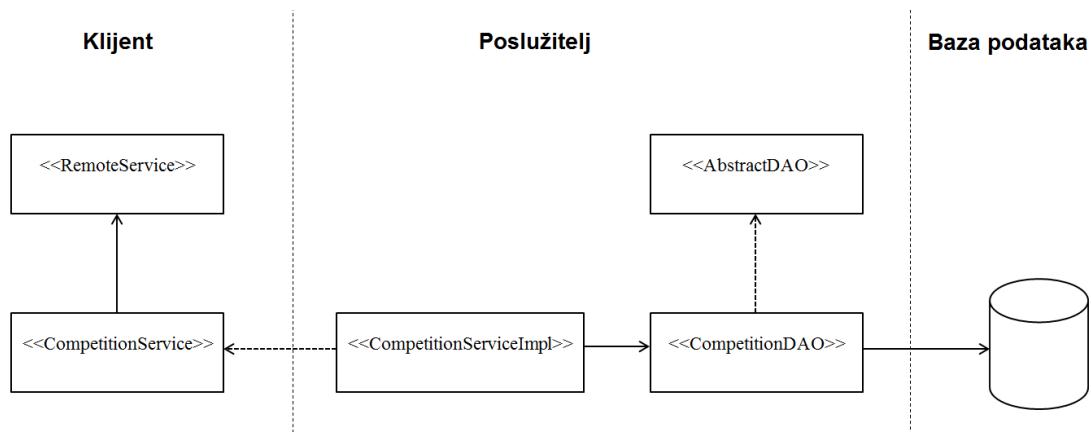
Razvijeno programsko rješenje korisnicima pruža mogućnost grafičkog prikaza nekoliko vrsta grafova u sklopu povjesne analize podataka putem jednostavnog grafičkog korisničkog sučelja.

Prilikom izrade programskog rješenja, jedna od temeljnih niti vodilja je bilo i omogućavanje jednostavne nadogradnje aplikacije prilikom daljnog razvoja. Sukladno tome, ostvareno rješenje jest skalabilno i robusno kako bi se i u budućnosti moglo nastaviti razvijati.

4.7.1. Programsko rješenje

Za izgradnju korisničkog sučelja korišten je Google Web Toolkit. Korisničko sučelje je pisano u programskom jeziku Java, te se u toku procesa prevođenja prevodi u *JavaScript*.

Poslužiteljska strana je razvijena uz pomoć *Spring frameworka* što omogućava jednostavno testiranje i proširenje aplikacije u budućnosti, te uvođenje novih mogućnosti koje Spring Framework pruža. Za perzistenciju podataka zadužen je *Hibernate framework*. Osnovna struktura aplikacije dana je slikom 6.



Slika 6: Osnovna struktura aplikacije

Klijentski dio se sastoji od razreda koji predstavlja ulaznu točku u svaku GWT aplikaciju (engl. *EntryPoint*), a unutar kojega je razvijeno grafičko korisničko sučelje.

Kao metoda komunikacije između klijenta i poslužitelja, odabran je *GWT RPC framework*. *RPC*, odnosno *Remote Procedure Call* omogućava jednostavnu razmjenu Java objekata putem protokola *HTTP* između klijentske i poslužiteljske strane. Implementacija *GWT RPC* servisa se zasniva na *Java servlet* arhitekturi.

S klijentske strane potrebno je izgraditi sučelje putem kojega će klijentska strana aplikacije komunicirati s poslužiteljskom. Sučelje treba sadržavati sve potrebne *RPC* metode. Odsječak koda koji prikazuje sučelje dan je ispod:

```
@RemoteServiceRelativePath ("services/competitionService")
public interface CompetitionService extends RemoteService {
    List<CompetitionDTO> getCompetitionList ();
}
```

Sučelje nasljeđuje *RemoteService*, a također definira i putanju koja će biti korištena prilikom poziva sučelja.

Na strani poslužitelja potrebno je napisati konkretnu implementaciju sučelja *CompetitionService*, što je prikazano sljedećim odsječkom koda:

```
@Service("competitionService")
@Transactional(readOnly = true)
public class CompetitionServiceImpl implements CompetitionService {

    @Autowired
    private CompetitionDAO competitionDAO;
    public List<CompetitionDTO> getCompetitionList() {
        List<CompetitionDTO> competitionDTOList = new
        ArrayList<CompetitionDTO>();
        List<Competition> competitionList =
        competitionDAO.listAll();

        if ((competitionList != null) &&
            (competitionList.size() > 0)) {
            for (Competition comp : competitionList) {
                competitionDTOList.add(
                    new
                    CompetitionDTO(comp.getCompetitionId(),
                    comp.getCompetitionName()));
            }
        }
        return competitionDTOList;
    } else {
        return null;
    }
}
```

Razred implementira sučelje CompetitionService, te odgovarajuće metode. Razred je označen odgovarajućom anotacijom koja označava da se radi o servisu, a unutar zagrada navodi putanju servisa koja mora odgovarati drugom dijelu putanje prilikom definicije sučelja.

Za potrebe dohvata podataka s baze podataka koristi se vrlo jednostavna implementacija *Data Access Object* oblikovnog obrasca. Razlog jednostavnosti leži u činjenici kako je zbog potreba aplikacije potrebno samo dohvatiti podatke iz baze podataka, ali ne i ostale operacije koje pripadaju osnovnih funkcijama svakog persistenntnog sloja, poput brisanja ili uređivanja podatka.

AbstractDAO je apstraktni razred kojeg nasljeđuju sve konkretnе DAO implementacije:

```
public abstract class AbstractDAO {  
  
    @Autowired  
    private SessionFactory sessionFactory;  
  
    public Session getCurrentSession(){  
        return sessionFactory.getCurrentSession();  
    }  
}
```

Za dohvat podataka s baze podataka se koristi konkretna implementacija CompetitionDAO.

```
@Repository  
public class CompetitionDAO extends AbstractDAO {  
  
    public List<Competition> listAll() {  
        Criteria criteria = getCurrentSession().createCriteria(  
            Competition.class);  
        return criteria.list();  
    }  
}
```

Dohvat podataka se vrši iz baze podataka sustava PLA i to putem virtualnih relacija. Ime pojedine virtualne relacije određeno je sa vpla_ime_tablice, primjerice vpla_weather_report.

Za potrebu izrade YAPLA sustava korištene su slijedeće virtualne relacije:

```
vpla_timeslot  
vpla_competition  
vpla_weather_report  
vpla_broker  
vpla_cash_position  
vpla_balancing_transaction
```

U sklopu modela, odnosno sloja perzistencije podataka, navedene virtualne relacije redom su pretvorene u razrede Timeslot, Competition, WeatherReport, Broker, CashPosition, BalancingTransaction.

Primjer dijela jednog takvog razreda, WeatherReport dan je u nastavku.

```
@Entity  
@Table (name = "vpla_weather_report")  
public class WeatherReport {  
    @Id  
    @Column (name = "wre_id")  
    private Integer id;  
    @Column (name = "wre_instance")  
    private Integer instance;  
    @ManyToOne  
    @JoinColumn (name = "wre_competition", referencedColumnName =  
    "cmp_id")  
    private Competition competition;  
    @ManyToOne  
    @JoinColumn (name = "wre_timeslot", referencedColumnName =  
    "tsl_id")  
    private Timeslot timeslot;  
    @Column (name = "wre_temperature")  
    private BigDecimal temperature;  
    @Column (name = "wre_wind_speed")  
    private BigDecimal windSpeed;  
    @Column (name = "wre_wind_direction")  
    private BigDecimal windDirection;  
    @Column (name = "wre_cloud_cover")  
    private BigDecimal cloudCover;  
}
```

Prilikom rada s *Hibernate frameworkom*, svaki razred za koji želimo ostvariti perzistenciju treba označiti s anotacijom `@Entity`. Također, nužno je uz pomoć anotacije `@Table` navesti na koju se tablicu baze podataka razred odnosi (u ovom slučaju `vpla_weather_report`).

Anotacija `@Id` označava da je atribut primarni ključ. Anotacija `@Column` omogućava povezivanje pojedinog atributa i njegove varijable u razredu, uz navođenje imena stupca unutar tablice unutar zagrada.

Referenciranje stranog ključa se obavlja anotacijom `ManyToOne`, što označava da se unutar tablice koju sad promatramo nalazi skup podataka koji imaju isti jedinstveni strani ključ. *Hibernate* dopušta i druge načine referenciranja stranog ključa (`OneToOne`, `OneToMany`, `ManyToMany`), sve zavisno o konkretnim potrebama unutar tablice. Dodatni atribut `name` označava atribut koji referenciramo (dakle stupac tablice koji sadrži strani ključ), a `referencedColumnName` je ime primarnog ključa iz tablice koja je referencirana.

Sukladno preporuci¹⁹, za atribute su korišteni razredi koji omataju primitivne tipove podataka, a koji omogućavaju postavljanje pojedinih tipova na `null` vrijednost (primjerice ako se pojavi `null` vrijednost u bazi podataka), što s primitivnim tipovima nije moguće. Nešto manja brzina rada u ovom slučaju nije od presudnog značaja.

Za svaki pojedini grafički prikaz programski je ostvaren zaseban servis koji omogućava dohvati podataka. Pojedini servis se dohvata tek u trenutku kada korisnik aplikacije odabere odgovarajući graf.

Primjerice, ako korisnik odabere prikaz kretanja temperature zraka za pojedino natjecanje, između poslužitelja i klijenta treba prenijeti listu parova vrijednosti koji sačinjavaju datum i vrijednost temperature zraka za taj datum.

Sukladno tome, za potrebe svakog pojedinog servisa su napravljeni *DTO* (*Data Transfer Object*) razredi, koji sadržavaju isključivo podatke bitne za pojedini grafički prikaz. Razlog za implementaciju ovakvih pojednostavljenih razreda je povećanje brzine rada, s obzirom na to da se u nekim slučajevima odbacuje više

¹⁹ <http://docs.jboss.org/hibernate/core/3.3/reference/en/html/persistent-classes.html#persistent-classes-pojo-identifier>

od polovice podataka i time povećava ukupna brzina rada, što na više tisuća parova vrijednosti može donijeti značajna poboljšanja u brzini rada.

Svi *DTO* objekti zbog zahtjeva *RPC* načina prijenosa podataka moraju implementirati sučelje `Serializable`, odnosno omogućavati svoju serijalizaciju.

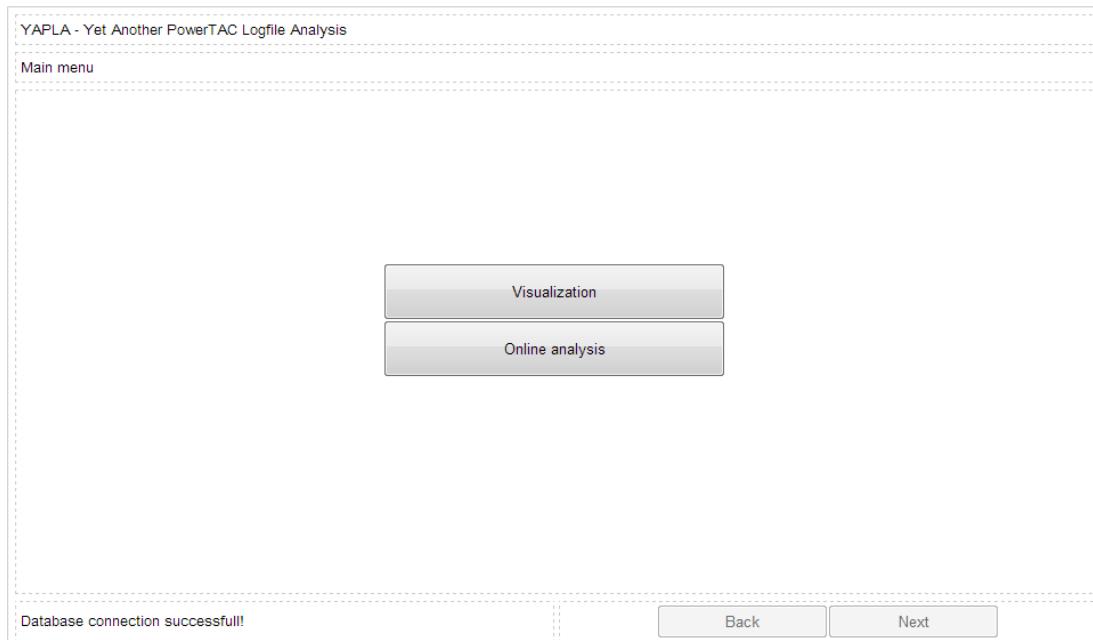
Svi bitni konfiguracijski parametri su zapisani u zasebne konfiguracijske datoteke, što omogućava jednostavnu kasniju promjenu parametara bez ponovnog prevođenja. Za *Spring framework*, konfiguracijska datoteka jest `applicationContext.xml`, a za *Hibernate framework* odgovarajuća datoteka jest `hibernateContext.xml`.

Konfiguracijski parametri potrebni za spajanje na bazu podataka – upravljački program (engl. *driver*), adresa, korisničko ime i lozinka su također zapisani u zasebnu konfiguracijsku datoteku, `database.properties`.

4.7.2. Povijesna analiza

Pristup do svakog pojedinog grafikona je moguć putem čarobnjaka koji konfigurira aplikaciju za prikaz želenog grafa. Opis korisničkog sučelja dan je u nastavku ovog poglavlja.

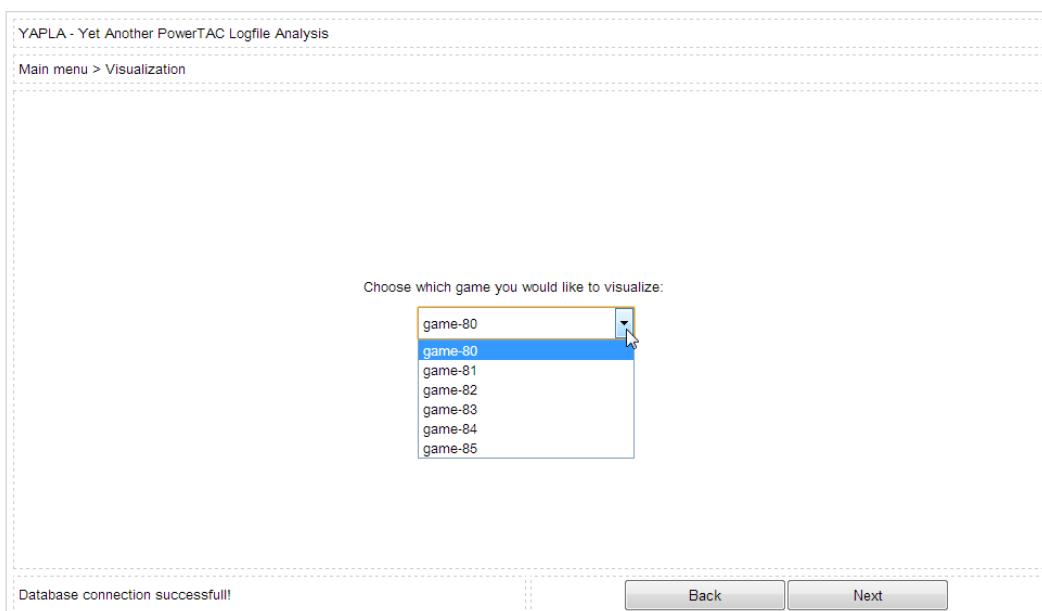
Prilikom otvaranja aplikacije korisnik pred sobom ima situaciju sa slike 7.



Slika 7: Početni ekran aplikacije

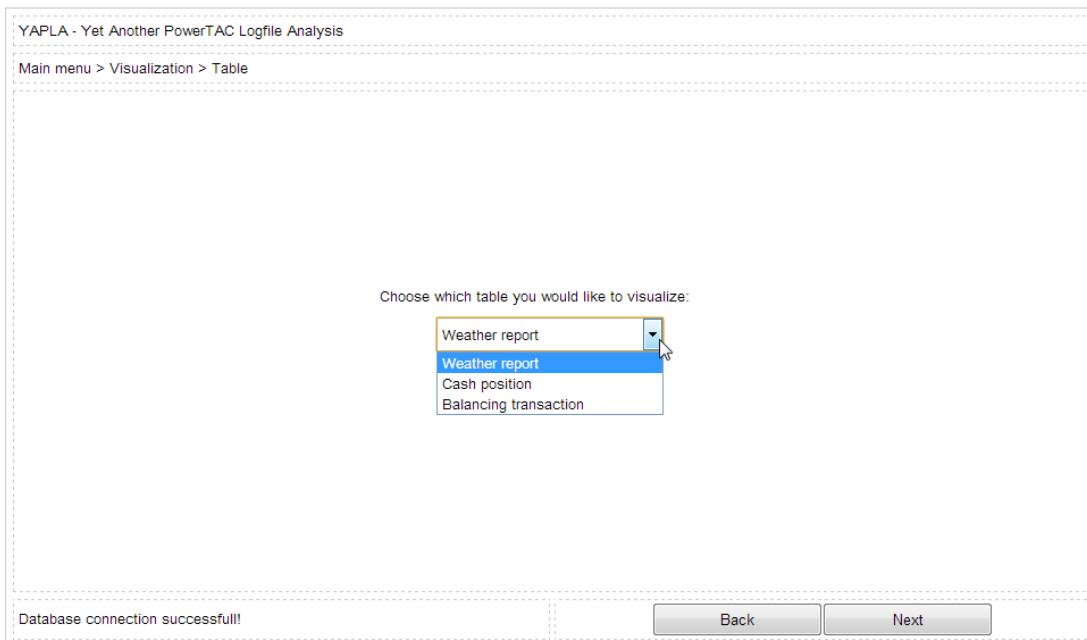
U gornjem lijevom kutu se nalazi naziv aplikacije. Ispod je putanja do trenutnog koraka čarobnjaka u kojem se korisnik nalazi. U krajnje lijevom donjem kutu korisniku se ispisuju obavijesti o uspješnosti priključenja na bazu podataka ili pogreške ako se na bazu podataka nije moguće priključiti. U donjem desnom kutu su tipke za povratak na prethodni korak ili prijelaz na sljedeći korak koje na ovom ekranu nije moguće koristiti. Većinu ekrana zauzima središnji dio na kojem se nalaze dvije tipke, prva „Visualization“ za povijesnu analizu i druga „Online analysis“ za postavke stvarno-vremenske analize.

Pritiskom na tipku „Visualization“ korisniku se prikazuje ekran sa slike 8.



Slika 8: Odabir željenog natjecanja

Korisnik u sklopu ovog ekrana ima mogućnost odabrati natjecanja (igre) koje želi grafički prikazati. Korisnik može odabrati između igri koje su učitane u bazu podataka. Nakon što je u padajućem izborniku igra odabrana, pritisak na tipku „Next“ nas vodi nas sljedeći korak, odabir odgovarajuće skupine grafičkog prikaza, što je vidljivo na slici 9.

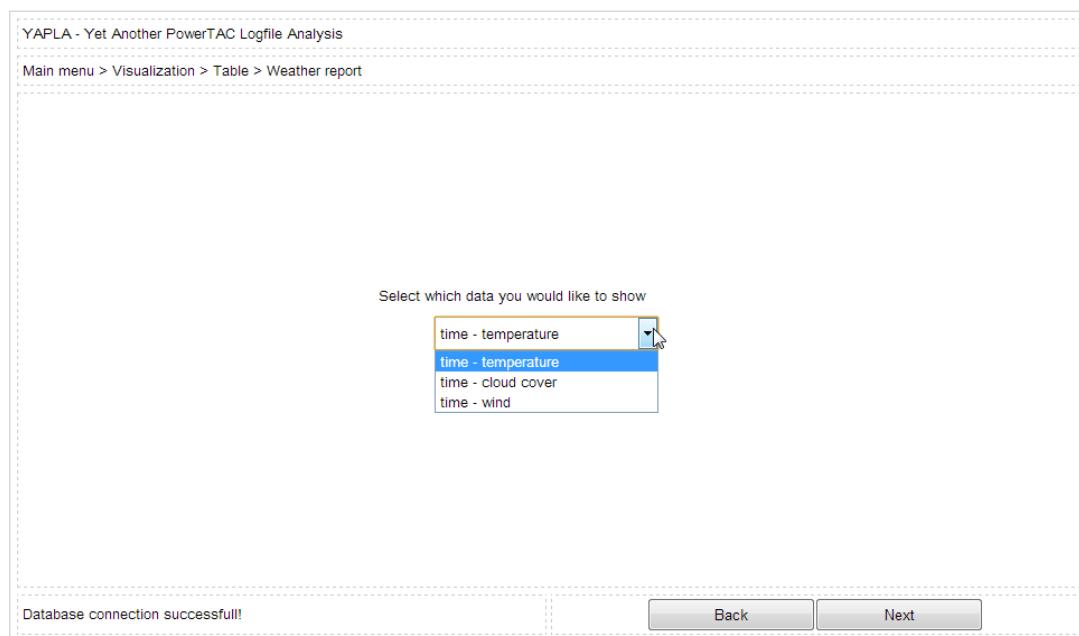


Slika 9: Odabir skupine prikaza

Korisniku su na raspolaganju tri skupine grafičkog prikaza:

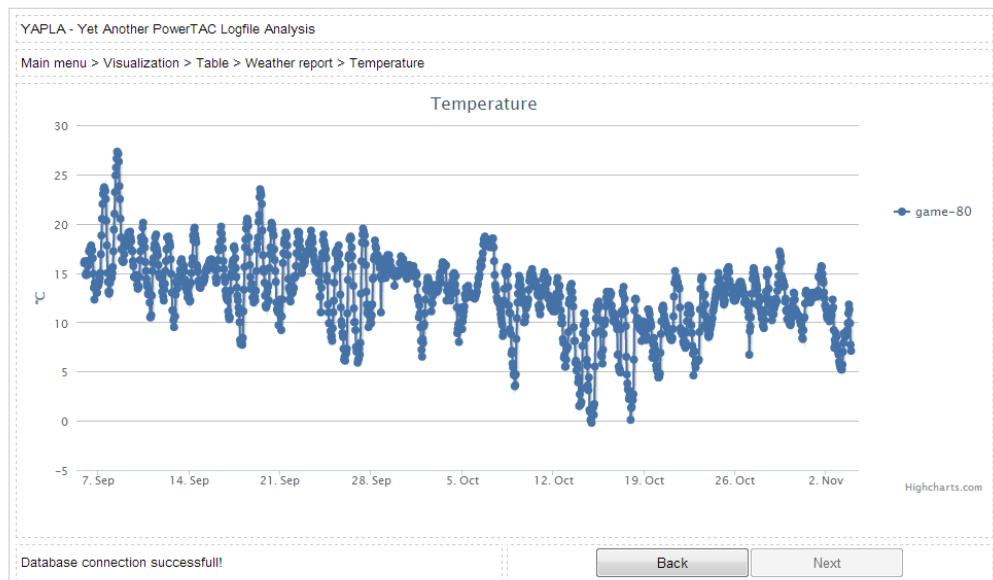
- Prikaz vremenskih uvjeta (engl. *Weather report*)
- Prikaz uravnotežujućih transakcija (engl. *Balancing transaction*)
- Prikaz količine akumuliranog novca (engl. *Cash position*)

Odabirom opcije „Weather report“, korisniku se prikazuje ekran sa slike 10.



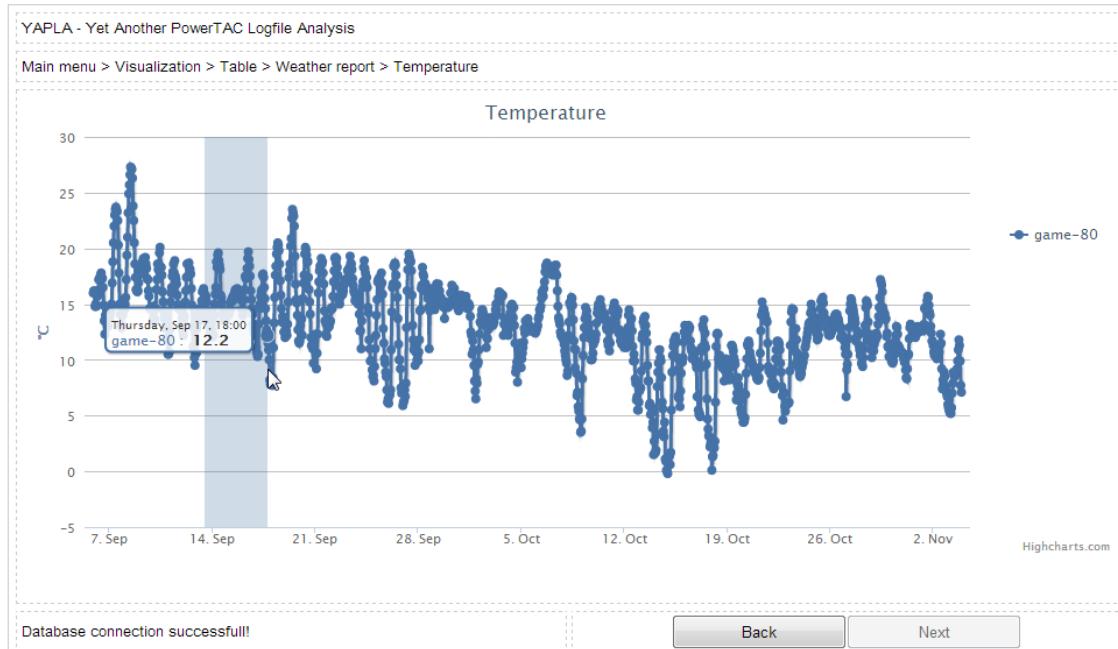
Slika 10: Odabir željenog grafa

Na ekranu sa slike 10, korisnik bira između vremenski prikaza temperature, brzine vjetra i prisutnosti naoblake.



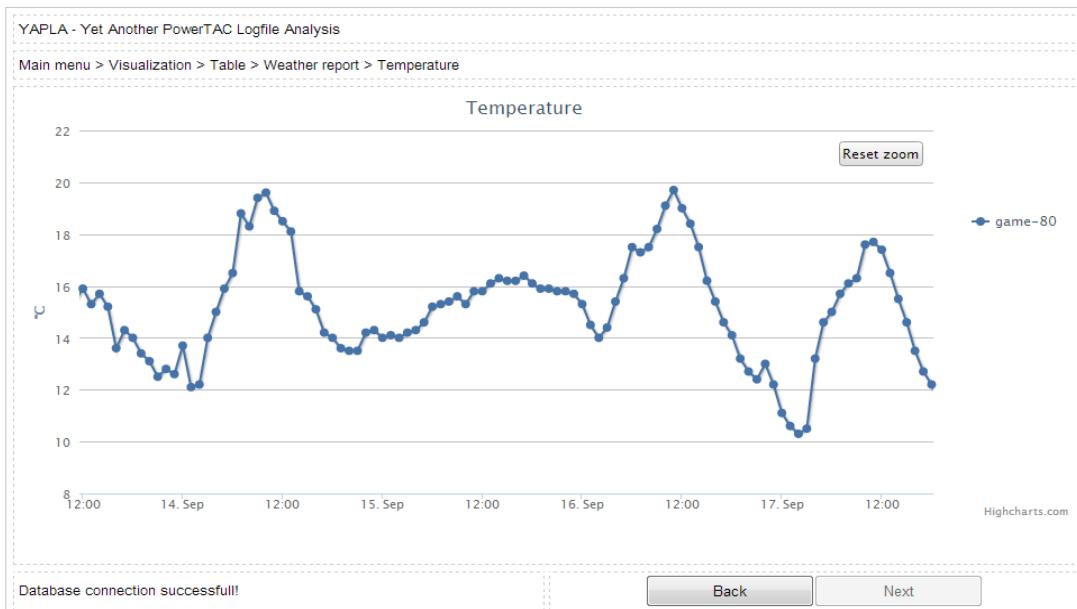
Slika 11: Grafički prikaz kretanja temperature zraka

Ako korisnik želi vidjeti grafički prikaz temperature zraka, odabire opciju „time - temperature“. Nakon kraćeg vremena potrebnog za učitavanje potrebnih podataka ispred korisnika se iscrtava željeni graf vidljiv na slici 11.



Slika 12: Uvećanje pogleda

Pritiskom na lijevu tipku miša na površini grafa te povlačenjem miša lijevo ili desno, označava se segment grafa, kao što je vidljivo na slici 12. Otpuštanjem tipke miša, graf se uvećava na tom segmentu kao što je vidljivo na slici 13.

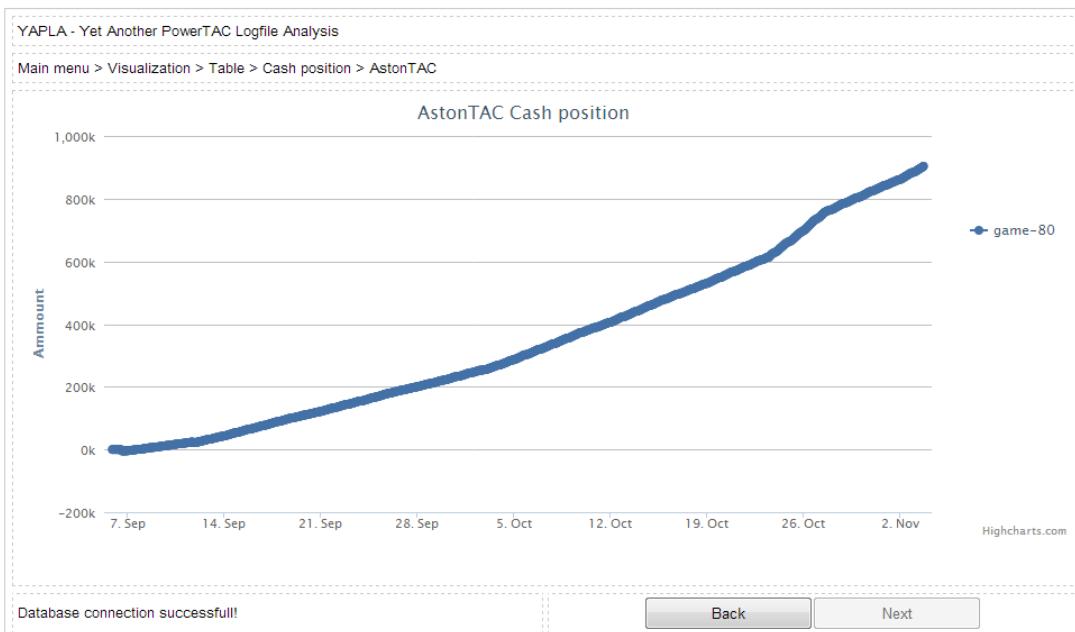


Slika 13: Uvećan pogled

Korisnik može i dalje povećavati pogled, međutim to ovdje nije prikazivano. Korisnik može obrisati povećanje pritiskom na tipku „Reset zoom“ u gornjem desnom kutu prozora. Pritiskom na tipku „Back“, korisnik se vraća na ekran sa slike 10, gdje ima mogućnost odabratи неки od preostala dva grafa iz skupine vremenskih prilika.

Opcije „time – cloud cover“ omogućava korisniku prikaz kretanja pokrivenosti neba oblacima tijekom razdoblja koje obuhvaća pojedina igra. Opcija „time - wind“ korisniku omogućava grafički prikaz kretanja brzine vjetra tijekom razdoblja koje obuhvaća igra.

Povratkom još jedan korak unazad, imamo situaciju sa slike 9. Odabirom opcije „cash position“ korisniku se otvara ekran s padajućim izbornikom u kojem ima mogućnost odabratи brokera za kojeg želi vidjeti kretanje količine novca na računu tijekom vremena. Izgled grafa dan je na slici 14. Također, korisnicima je ponuđen još jedan graf, uravnotežujuće transakcije koji ovdje nije detaljnije razmatran.



Slika 14: Grafički pregled kretanja salda

4.7.3. Stvarno-vremenska analiza

Stvarno-vremenska analiza (engl. *online-analysis*) omogućava agenciju pristup do korisnih informacija zapisanih u dnevničkim datotekama putem REST sučelja, koristeći JSON oblik zapisa podataka.

Kao metoda analize podataka odabrane su umjetne neuronske mreže. Umjetne neuronske mreže su kategorija umjetne inteligencije čija se osnovna zamisao rada temelji na biološkim neuronskim mrežama, kakav je primjerice ljudski mozak.

U odnosu na tradicionalne modele zasnovane na *von Neumann-ovom* modelu sekvenciјalnog računala koje algoritme izvodi korak po korak, neuronska mreža ima dvije temeljne prednosti:

- neuronska mreža uči samostalno
- ulazni podaci ne moraju biti precizni

S druge strane, *von Neumannovovom* modelu se unaprijed detaljno mora opisati algoritam, a ulazni podaci moraju biti precizni. Praksa međutim pokazuje kako su mnogi svakodnevni zadaci preteški da bi ih se formaliziralo preciznim

algoritmom. U slučaju podataka sadržanih u dnevničkim datotekama, moguće je analizom i upotrebom logike donijeti neke zaključke, primjerice poput toga da će kućanstva manje energije trošiti noću nego danju, ali porastom broja ulaznih parametara (poput temperature, brzine vjetra i drugih faktora) postaje potpuno nemoguće donijeti općenite zaključke, a još manje napisati odgovarajući algoritam.

Neuronske mreže ne posjeduju prije navedene probleme, te su se pokazale idealnima za rješavanje problema analize velikog broja podataka gdje nije moguće pronaći eksplisitnu vezu između ulaznih i izlaznih podataka.

4.7.3.1. Strategija analize i predviđanja

S obzirom na to da trenutna inačica CrocodileAgent-a nema mogućnost predviđanja proizvodnje na temelju promjenjivih vremenskih uvjeta, a kako je mogućnost uravnoteženja potrošnje i proizvodnje važan element uspjeha svakog agenta, upravo to je odabранo kao strategija koja će agentu pružiti dodatno znanje.

Obnovljivi izvori unutar Power TAC simulacijske platforme su predstavljeni solarnim elektranama i vjetroelektranama. U uvodnom opisu Power TAC platforme je navedeno kako proizvodnja el. energije iz solarnih izvora ovisi o dobu dana (energija se ne proizvodi noću) i vremenskim uvjetima (pokrivenost neba oblacima). Veća pokrivenost neba oblacima dovodi do manje insolacije, te smanjenje proizvodnje el. energije. S druge strane, vjetroelektrane proizvode energiju u ovisnosti o smjeru i brzini puhanja vjetra.

Treniranje će biti obavljeno s parovima ulaz – izlaz, gdje su ulazni podaci tip elektrane, brzina vjetra, smjer vjetra, pokrivenost neba oblacima te doba dana. Izlazni podaci će predstavljati proizvodnju el. energije za neki ulaz.

Treniranje mreže će biti ostvareno s nekoliko igara, a potom evaluirano uz pomoć igara s kojima nije trenirano, a također i na natjecanju Power TAC 2013.

Nakon treniranja, agent može zatražiti podatke putem REST sučelja na temelju njemu unutar simulacije dostupne vremenske prognoze. Preciznost predviđenih podataka dakle ne ovisi samo o strukturi neuronske mreže i podacima s kojima je mreža trenirana nego i o preciznosti vremenske prognoze.

5. Smjernice za daljnji rad

Nastavak razvoja ovog alata treba se odvijati u četiri osnovna smjera koji će rezultirati većom funkcionalnošću i korisnošću aplikacije. Daljnji razvoj se treba temeljiti na povećanju obujma podataka koji se prikazuju korisniku, ugodnijem korisničkom sučelju, većoj neovisnosti aplikacije, te svakako na uvođenju novih strategija za stvarno-vremensku analizu podataka. Dodatno postoji potencijal ugradnje modula koji bi pratio tijek stvarno-vremenske analize putem grafičkog prikaza.

5.1. Vlastita baza podataka i modul za učitavanje dnevničkih datoteka

Razvijeno programsko rješenje u trenutnom stanju kao bazom podataka služi se vanjskom PLA bazom podataka s ograničenim pristupom. Sukladno tome, predstavlja vanjsku zavisnost i u tehničkom smislu, primjerice s obzirom na dostupnost, ali i u smislu količine podataka - s obzirom na to da je u bazu podataka trenutno učitan samo malen broj dnevničkih datoteka.

Vlastitu bazu podataka je moguće pokrenuti temeljeno na PLA arhitekturi. Takvo rješenje će omogućiti neovisnost o vanjskom izvoru podataka. Modul za učitavanje dnevničkih podataka je moguće pokrenuti jednom kada je pokrenuta vlastita baza podataka. Takav modul će omogućiti učitavanje dnevničkih datoteka prema potrebama, što će povećati broj korisnih informacija korisnicima aplikacije, ali i povećati bazu znanja za agenta. Za izgradnju modula svakako treba iskoristiti i postojeći alat za učitavanje datoteka (*log-tool*) razvijen u sklopu natjecanja Power TAC.

Posljednje proširenje koje je moguće ostvariti u ovom smjeru jest proširenje modula za učitavanje podataka s funkcionalnošću koja će mu omogućiti automatizirano preuzimanje dnevničkih datoteka s poslužitelja po završetku pojedinog natjecanja (bilo periodičkim provjerama i preuzimanjima, bilo preuzimanjem na zahtjev korisnika).

5.2. Proširenje modula za povijesnu analizu

Trenutno aplikacija omogućava grafički prikaz samo malog broja, u dnevničkim datotekama dostupnih podataka. Odnosi se to na prikaz podataka o vremenskim uvjetima, uravnotežujućim transakcijama i finansijskom saldu. Korisnost aplikacije će se svakako povećati grafičkom analizom svih dostupnih podataka. Osim toga, moguće je za isti tip podataka prikazivati više različitih grafikona umjesto postojećih linijskih koji prikazuju kretanje odabranog parametra s obzirom na protok vremena unutar simulacije.

Dodatne informacije mogu pružiti kombinirani grafikoni, grafikoni koji prikazuju više podataka, poput:

- odnos temperature zraka i potrošnje specifične skupine, poput kućanstava.
- odnos proizvodnje vjetroelektrane s obzirom na brzinu i smjer vjetra.
- odnos proizvodnje solarne elektrane s obzirom na pokrivenost neba oblacima.

5.3. Poboljšanje grafičkog korisničkog sučelja

Povećanjem funkcionalnosti aplikacije, prikazivanjem većeg broja grafičkih prikaza, općenito povećanjem broja informacija koje će aplikacija nuditi korisniku pojavit će se potreba za redizajnom i novom organizacijom korisničkog sučelja. Pritom su temeljne smjernice kojima se treba voditi ergonomija i jednostavnost korištenja. Također, aplikacija bi trebala slijediti dizajn i osnovne grafičke ideje koje su ugrađene u druge alate koji se koriste u Power TAC natjecanju (primjerice *Visualizer*). To će omogućiti jednostavnije snalaženje potencijalnim korisnicima aplikacije, naviknutima na postojeće komponente sustava.

5.4. Uvođenje dodatnih strategija za stvarno-vremensku analizu

Analiza podataka sadržanih u dnevničkim datotekama je pokazala kako je moguće implementirati nekoliko strategija, koje bi CrocodileAgentu kao korisniku istih omogućilo više dodatnih informacija i bolje ponašanje u simulacijama. Daljnja analiza dnevničkih datoteka i razvoj (jedne ili više) strategija svakako treba biti u primarnom fokusu nastavka razvoja aplikacije. Osim ovdje navedenih prijedloga strategija, nije isključena mogućnost postojanja još nekih strategija, ali i mogućnost da su neke ovdje predložene strategije korak u krivom smjeru.

5.4.1. Predviđanje potrošnje el. energije na temelju vremenskih uvjeta i doba dana

Iako Crocodile Agent već sada implementira strategiju za praćenje periodičnosti potrošnje pojedinih skupina korisnika, nedostaje mogućnost analize s obzirom na promjenjive vremenske uvjete. Strategija polazi od pretpostavke kako potrošnja električne energije različitih tipova potrošača (kućanstva, uredi) ne ovisi samo o dobu dana ili danu u tjednu već i primjerice o porastu temperature što će dovesti do povećanja potrošnje el. energije zbog uključivanja klima uređaja.

Ovakav model predviđanja treba u obzir uzeti i sat u danu, ili dan u tjednu jer izostavljanje ovih podataka može dovesti do krivih zaključaka (pad potrošnje noću bi se mogao pogrešno dovesti u vezu isključivo s padom temperature u noćnim satima, bez informacije kako se radi o većinskom padu uzrokovanim smanjenom potrošnjom u noćnim satima).

Skup ulaznih podataka sukladno tome čine sat u danu, dan u tjednu, vrsta potrošača, te vremenski uvjeti (temperatura zraka). Skup izlaznih podataka predstavlja jediničnu potrošnju el. energije za taj tip korisnika.

Smisao ovakvog predviđanja jest omogućiti agentu (CrocodileAgent) bolje predviđanje budućih potreba za električnom energijom, na temelju vremenske prognoze dostupne unutar pojedine igre za 24 iduća vremenska odsječka (24 sata). Baratajući s takvom informacijom, agent može lakše upravljati razlikama

između količine osigurane energije i zahtjevima za energijom od strane svojih potrošača. Ova strategija je naročito usmjerena prema izbjegavanju velikih deficit-a energije koje potom treba nadoknaditi DU, po visokim cijenama.

5.4.2. Predviđanje promjena cijene el. energije ovisno o broju sudionika u pojedinoj igri

Različiti agenti na različit način utječu na tijek odvijanja pojedine simulacije. Strategija polazi od toga da se pokuša ustanoviti veza između kretanja cijena na veleprodajnom tržištu i pojedinih agenata koji su prisutni u igri. Strategija zbog svoje složenosti nije detaljnije razmatrana u ovom radu.

5.5. Uvođenje modula za praćenje rezultata stvarno-vremenske analize

Ako se pokaže potreba prilikom implementacije nekih strategija, postoji mogućnost ugradnje interaktivnog grafičkog prikaza odvijanja simulacije. Sama ideja je opisana u idućih nekoliko koraka:

1. CrocodileAgent zatraži predviđanje nekog podataka za neki budući vremenski odsječak t na temelju njemu dostupnih podataka.
2. YAPLA sustav izračuna predviđanje za vremenski odsječak t i odgovori CrocodileAgent-u.
3. CrocodileAgent prilikom novog zahtjeva za predviđanje u podatke pohranjuje stvarnu vrijednost parametra koji je bio predviđan u vremenskom odsječku t
4. YAPLA sustav korisniku na ekran iscrtava interaktivni grafički prikaz s predviđenom i ostvarenom vrijednosti parametra u trenutku t .

Ovakvo interaktivno rješenje bi odmah moglo numerički izračunavati pogrešku, ali je i grafički zorno prikazati korisniku. Osim toga, zanimljivu dodatnu funkcionalnost predstavlja mogućnost pohrane dinamičkog praćenja rezultata stvarno-vremenske analize. Korisnom opcijom se može pokazati i mogućnost bilježenja vremenskih odsječaka u kojima su ostvarena velika odstupanja između rezultata simulacije i rezultata predviđenih unutar strategije predviđanja. Sve navedeno ima potencijal pružiti vrijedan izvor informacija za ocjenjivanje uspješnosti pojedine strategije stvarno-vremenske analize ili podataka koji su korišteni kao baza znanja za strategiju.

Zaključak

Budućnost pred elektroenergetske sustave stavlja nove izazove koje treba svladati – povećanje udjela obnovljivih izvora el. energije, liberalno tržište, napredne elektroenergetske mreže. Elektroenergetski sustavi budućnosti zasigurno neće više biti samo infrastruktura za prijenos električne energije, već složena infrastruktura bazirana na komunikacijskim i informacijskim tehnologijama te nizom novih usluga koje će rezultirati učinkovitijom potrošnjom el. energije, većom primjenom obnovljivih izvora te smanjenjem troškova po krajnje korisnike.

Razvoj takvih sustava zahtjeva podrobnu analizu i pripremu prije uvođenja novih koncepata distribucije el. energije na tržište, ponajprije zbog ogromne važnosti koju el. energija zauzima u modernom društvu.

Natjecanje Power TAC je međunarodni projekt šest sveučilišta iz Europe i Sjeverne Amerike čiji je cilj izgradnja otvorene tržišne računalne simulacijske platforme za tržišta el. energijom. Konačni cilj projekta je pomoći definirati smjernice za dizajn novih vrsta tržišta el. energijom. Sveučilište u Zagrebu jedan je od članova projekta razvoja Power TAC, u sklopu kojeg razvija vlastitog agenta Crocodile Agent i važnu zajedničku komponentu projekta, *Visualizer*.

Velika količina podataka koja je rezultat svake pojedine simulacije treba biti adekvatno obrađena kako bi se na temelju tih podataka mogli donijeti neki zaključci korisni za razvoj elektroenergetskih sustava budućnosti. Stoga je u okviru ovog završnog rada razvijeno rješenje za povijesnu i stvarno-vremensku analizu dnevničkih podataka koji su rezultat izvođenja pojedinih simulacija.

Razvoj aplikacije nije samo doveo do gotovog rješenja već i do razvoja novih ideja koje u budućnosti razvoja ovog alata svakako trebaju biti razmotrene. Ponajprije se to odnosi na dodatne strategije stvarno-vremenske analize, te povećanje korisnih informacija u sklopu povijesne analize, što će u budućnosti dati nove vrijedne informacije za razvoj Power TAC simulacijske okoline, a tako i elektroenergetskih sustava budućnosti.

Literatura

- [1] UN, "Agenda 21: Earth Summit - The United Nations Programme of Action from Rio," *United Nations Conference on Environment and Development UNCED or the Earth Summit Rio de Janeiro Brazil 1992*. United Nations, p. 294, 1993.
- [2] Euroobserver, "The State of Renewable Energies in Europe - 11th EurObserv'ER Report," 2011.
- [3] REN21, "Renewables 2011 Global Status Report," Deutsche Gesellschaft für Technische Zusammenarbeit (GTZ) GmbH, 2011.
- [4] European Commission, "European Technology Platform SmartGrids Strategic Deployment Document for Europe's Electricity Networks of the Future," 2010.
- [5] B. Dupont, C. De Jonghe, K. Kessels, and R. Belmans, "Short-term Consumer Benefits of Dynamic Pricing," in *2011 8th International Conference on the European Energy Market EEM*, 2011, pp. 216-221.
- [6] S. Borenstein, J. B. Bushnell, and F. A. Wolak, "Measuring Market Inefficiencies in California's Restructured Wholesale Electricity Market," *American Economic Review*, vol. 92, no. 5, pp. 1376-1405, 2002. EEGI, "Roadmap 2010-18 and Detailed Implementation Plan 2010-12," *Technical Report*, 2010.
- [7] EEGI, "Roadmap 2010-18 and Detailed Implementation Plan 2010-12," *Technical Report*, 2010.
- [8] C. Block, J. Collins, W. Ketter, and C. Weinhardt, "A Multi-Agent Energy Trading Competition", *Technical Report*, no. ERS-2009-054-LIS. Erasmus Research Institute of Management, Erasmus University, 2009.
- [9] W. Ketter, J. Collins, P. Reddy, M. Weerd, "The 2012 Power Trading Agent Competition", *Technical Report*, no. ERS-2012-010-LIS, Erasmus Research Institute of Management, Erasmus University 2012.
- [10] A. Somani and L. Tesfatsion, "An agent-based test bed study of wholesale power market performance measures", *IEEE Computational Intelligence Magazine*, vol. 3 no. 4 pp. 56–72, November 2008.
- [11] J. Collins, W. Ketter, and N. Sadeh, "Pushing the limits of rational agents: the trading agent competition for supply chain management", *AI Magazine*, vol. 31, no. 2, pp. 63–80, 2010.
- [12] P. R. Jordan, B. Cassell, L. F. Callender, and M. P. Wellman, "The ad auctions game for the 2009 trading agent competition", *Technical report*, University of Michigan, Department of Computer Science and Engineering, 2009.
- [13] J. Babic et al., "The CrocodileAgent 2012: Research for Efficient Agent - based Electricity Trading Mechanisms," in *Special Session on Trading Agent Competition @ KES-AMSTA 2012*, 2012, pp. 1-13.
- [14] A. Petric, V. Podobnik, and G. Jezic, "The CrocodileAgent : Analysis and comparison with other TAC SCM 2005 agents," in *Proceedings of the*

AAMAS Joint International Workshop on the Trading Agent Design and Analysis and Agent Mediated Electronic Commerce (TADA/AMEC), 2006, pp. 202-205.

- [15] I. Siranovic, T. Cavka, A. Petric, and V. Podobnik, "A Bidding Agent for Advertisement Auctions : An Overview of the CrocodileAgent 2010," in *Proceedings of the 2011 Workshop on Trading Agent Design and Analysis, IJCAI*, 2011, pp. 68-75.
- [16] P. R. Winters, "Forecasting Sales by Exponentially Weighted Moving Averages," *Management Science*, vol. 6, no. 3, pp. 324-342, 1960.
- [17] C. C. Holt, "Forecasting seasonals and trends by exponentially weighted moving averages," *International Journal of Forecasting*, vol. 20, no. 1, pp. 5-10, 2004.
- [18] J. Babic, „A competitive simulation platform for electricity trading in smart grids“, *Master's thesis*, Department of Telecommunications, Faculty of Electrical Engineering and Computing, University of Zagreb, 2012.
- [19] M. Peters, *Analysing the Power Trading Agent Competition*, Rotterdam School of Management, 2012.

Sažetak

Web-modul za analizu događaja na tržištima električnom energijom

Sve veći udio obnovljivih izvora električne energije pred elektroenergetske sustave budućnosti stavlja nove izazove. Takav elektroenergetski sustav više neće biti samo infrastruktura za prijenos el. energije, već složen sustav obogaćen informacijskim i komunikacijskim tehnologijama koji će omogućiti nove usluge i kvalitetnije upravljanje elektroenergetskom mrežom.

Razvoj naprednih elektroenergetskih sustava zahtjeva složene analize. Natjecanje Power TAC jest međunarodni projekt čiji je cilj izgradnja otvorene tržišne računalne simulacijske platforme za tržišta el. energijom. Rezultat simulacija predstavlja niz podataka koji sadrže korisne informacije koje mogu dati odgovor na brojna otvorena pitanja vezana uz elektroenergetske sustave budućnosti. Predmet ovog rada jest izrada sustava za povijesnu i stvarno-vremensku analizu takvih podataka. Osim programskog rješenja, u radu je dan i skup prijedloga za daljnji nastavak rada na području analize podataka.

Ključne riječi: napredne energetske mreže, simulacijska platforma, programski agenti, povijesna analiza, stvarno-vremenska analiza, Power Trading Agent Competition

Summary

Web-module for the analysis of events in the electricity market

An increase in availability of renewable energy sources presents new challenges for power systems of the future. Energy transmission infrastructure will no longer be the only infrastructure for transmission of electricity. It will be enriched with a complex system of information and communication technologies that will enable new services and better management of the power grid. Thus, the development of advanced power systems requires complex analysis. Power TAC is an international project which aims to build an open simulation platform for power markets of the future. The output of the simulation is a set of data which contains valuable information which can then be used to answer many open questions related to the electric power system of the future. The subject of this paper is to develop a system for offline and online analysis of such data. In addition to a software solution, the paper also outlines proposals for future work in the field of data analysis.

Keywords: smart grids, simulation platform, software agents, off-line analysis, on-line analysis, Power Trading Agent Competition.

Skraćenice

DSM	<i>Demand Side Management</i>
TAC	<i>Trading Agent Competition</i>
Power TAC	<i>Power Trading Agent Competition</i>
DU	<i>Distribution Utility</i>
PLA	<i>PowerTAC Logfile Analysis</i>
AJAX	<i>Asynchronous JavaScript and XML</i>
XML	<i>Extensible Markup Language</i>
GWT	<i>Google Web Toolkit</i>
CSS	<i>Cascading Style Sheets</i>
XHTML	<i>Extensible HyperText Markup Language</i>
JSON	<i>JavaScript Object Notation</i>
REST	<i>Representational State Transfer</i>
DAO	<i>Data Access Object</i>
DTO	<i>Data Transfer Object</i>
API	<i>Application Programming Interface</i>