

REŠAVANJE "BACKUP" PROCESA PRIMENOM METAHEURISTIKE OPTIMIZACIJA KOLONIJOM PČELA

Ranko Nedeljković, Slobodan Mitrović, Dragana Drenovac
Saobraćajni fakultet u Beogradu

Sadržaj: *U radu je razmatran problem alokacije backup kopija na backup serverima primenom metaheuristike Optimizacija kolonijom pčela. Predstavljeni su osnovni pojmovi backup procesa, problem koji se razmatra, kao i navedena metaheuristika. Kroz numerički primer prikazan je način rešavanja opisanog problema.*

Ključne reči: *backup, servis, metaheuristikčki algoritmi, optimizacija kolonijom pčela*

1. Uvod

U radu je razmatran proces skladištenja rezervnih podataka, u praksi poznat i kao „data backup“ proces. Uređene računarske mreže, koje svojim korisnicima pružaju određen broj on-line servisa, povećavaju kvalitet svoje usluge, između ostalog i formiranjem rezervnih kopija podataka uskladištenih na mrežnim serverima. Na navedeni način se stvara „hladna rezerva“ samih podataka, koji, u slučaju raznih vrsta otkaza hardvera i softvera, mogu biti restaurirani i čime se može ostvariti kontinuitet servisa sa potencijalno minimalnim gubicima.

Na uređenim računarskim mrežama postoji veći broj servisa koji su funkcionalno povezani i koji stvaraju jednoj ili većem broju grupa korisnika integrisano okruženje pogodno za produktivno funkcionisanje. Određen broj servisa je vezan za „back-end“ okruženje i ovi servisi ne stoje direktno na raspolaganju samim korisnicima, ali su vezani za njih. Primer za to su servisi poput „MS Active Directory“, NIS ili LDAP, i sl., koji mogu biti, nadalje, povezani sa RADIUS serverom, i svim ostalim servisima, a koji imaju za zadatak da drže baze podataka o samim korisnicima, njihovoj hijerarhiji u okviru posmatrane mreže, kao i kredencijalima koje imaju u okviru svojih pozicija. „Back-end“ grupaciji pripadaju i drugi vidovi servisa, poput DNS, Proxy, DHCP, kao i bezbednosni (Cisco Secure Access Control Server – ACS) vezani za NAC/NAP okruženje i sl. i koji takođe mogu uticati na rad korisnika. Druga grupacija servisa je vezana za „front-end“ okruženje i ovi servisi mogu stajati direktno na raspolaganju korisnicima sistema. Primer za to su e-mail, kao i razne vrste Web servisa, poput, webmail-a, e-learning, e-commerce portala, portala za rezervaciju resursa, socijalnih mreža, kao i standardni web sajtovi informativnog karaktera. Najveći deo navedenih „front-end“ servisa je podložan dinamičkoj izmeni sadržaja od strane korisnika.

Ozbiljni sistemi koji pružaju najveći deo navedenih servisa u sebi imaju inkorporirane komponente redundantnosti na hardverskom, softverskom, kao i organizacionom nivou. On-line sistemi za skladištenje podataka, u zavisnosti od nivoa investicije mogu, kod najskupljih rešenja, sadržati redundantne sisteme na nivou "storage" servera, dok jeftinija rešenja, na hardverskom nivou, po pravilu sadrže RAID Mirror (uz ili bez Stripe-a) konfiguraciju hard diskova. Najjeftinija rešenja se moraju oslanjati na on-line ili near-line backup bez prisustva hardverske redundanse. Bez obzira na nivo investicije u svakom od sistema, bi trebalo da bude prisutan i organizacioni nivo backup procesa koji podrazumeva formiranje rezervne kopije podataka na drugoj lokaciji (Backup site ili DRC - disaster recovery center), tj. na jednom ili više izdvojenih backup servera ili storage sistema. Razlozi za to su opravdani, čak i u slučaju najskupljih sistema, jer se u datom slučaju pretpostavlja da uskladišteni podaci imaju i najveću vrednost ili važnost, pa motivi za formiranje rezervne kopije nisu direktno vezani za mogućnost otkaza hardvera, već za daleko ozbiljnije scenarije poput požara, elementarnih nepogoda i sl.

Da bi se u slučaju otkaza ili nezgode minimizirala potencijalna šteta, poželjno je vršiti backup datoteka što češće. Ovakav zahtev ima svoja ograničenja, ne samo u pogledu materijalnih troškova, već i vremena koje je utrošeno na sam backup proces, koji se najčešće može obavljati u periodu kada u sistemu postoji minimalan broj aktivnih korisnika ili ih nema uopšte.

U navedenim sistemima se mogu primenjivati različiti oblici backup repozitorijuma, poput punog, inkrementalnog i diferencijalnog backup-a. Pun backup (full backup) podrazumeva kopiranje svih podataka sa date lokacije. Inkrementalni backup podrazumeva kopiranje samo onih datoteka, koje su, na bilo koji način, izmenjene u odnosu na stanje pri poslednjem backup-u. Diferencijalni backup podrazumeva kopiranje samo onih datoteka koje su na bilo koji način, izmenjene u odnosu na stanje pri poslednjem punom backup-u. Iz navedenog se može zaključiti da je najviše vremena upravo potrebno za pun backup, tako da se u praksi često primenjuju strategije u kojima se pun backup vrši jednom sedmično, u toku vikenda, dok se diferencijalni ili inkrementalni backup vrši jednom dnevno.

2. Opis problema

Posmatra se uređena mreža sa određenim brojem mrežnih servisa koji su postavljeni na više različitih servera. U okviru mreže su postavljena 3 backup servera za skladištenje rezervnih podataka. Na backup serverima se vrši pun backup i diferencijalni backup. Obzirom da backup serveri imaju ograničen kapacitet, zadato je da se, u slučaju kada na prvom od backup servera slobodan prostor padne na manje od 15%, na sva tri servera se briše stari backup zapis i pravi se novi pun backup, koji se nadopunjuje diferencijalnim backup-om. Kako je računarska mreža angažovana 7 dana u nedelji, ne postoji mogućnost primene strategije puštanja punog backup-a tokom vikenda i dopunjavanja inkrementalnim backup-om na dnevnom nivou. Pošto se obim strukture datoteka na određenom broju servisa značajno menja u toku jednog vremenskog perioda koji obuhvata nekoliko navedenih backup ciklusa, postoji potreba za rekompozicijom pozicija backup-a različitih servisa u skladu sa raspoloživim prostorom na backup serverima. Backup proces treba izvršiti u vremenskom periodu u kome je utvrđeno da nema nijednog korisnika na sistemu ili ih ima zanemarljivo malo. Bitno je napomenuti da

je rad orijentisan ka jeftinijim mrežama, kao i mrežama starijeg datuma, kod kojih dominiraju LAN veze kapaciteta 100Mbps. Rad nije orijentisan ka mrežama sa skupljom infrastrukturom, jer organizacije koje su vlasnici takvih mreža nemaju finansijsko ograničenje vezano za rešavanje problema u domenu backup-a podataka, mada se navedeno rešavanje problema, u određenim slučajevima, može primeniti i kod njih.

3. Opis metaheuristike

Metaheuristika pod nazivom Optimizacija kolonijom pčela nedavno se pojavila kao nov pravac na polju inteligencije grupe a predložili su je i koristili autori Lučić i Teodorović (2001, 2002, 2003). Veštačke pčele predstavljaju agente koji kroz međusobnu saradnju mogu da reše složene probleme kombinatorne optimizacije. Na početku procesa pretrage one se nalaze u košnici iz koje izlaze kada treba da načine niz lokalnih poteza stvarajući tako parcijalno rešenje problema. Prilikom stvaranja jednog ili više dopustivih rešenja, veštačke pčele, uz neposrednu komunikaciju, postepeno dodaju komponente i tako proširuju postojeće parcijalno rešenje. Najbolje rešenje pronađeno u prvoj iteraciji se pamti, a proces traganja za boljim rešenjem pčele nastavljaju kroz nove iteracije. Donosilac odluke zadaje broj iteracija. Pretražujući prostor rešenja, veštačke pčele izvode letove unapred i letove unazad. Tokom leta unapred, na osnovu pojedinačnih sposobnosti i prethodno stečenog kolektivnog iskustva, one generišu različita parcijalna rešenja. Potom se vraćaju u košnicu u kojoj razmenjuju informacije i upoređuju generisana parcijalna rešenja. Na osnovu toga svaka veštačka pčela odlučuje da li da odustane od parcijalnog rešenja do kojeg je došla poslednjim letom unapred i postane neopredeljeni sledbenik ili da ostane lojalna tom rešenju i nastavi da ga proširuje. Veštačka pčela pre novog leta unapred može igrom da motiviše neopredeljene pčele da je slede na njenom putu ili da bez regrutovanja drugih pčela nastavi da proširuje svoje parcijalno rešenje. Regrutovana pčela prati pčelu čije je parcijalno rešenje prihvatila do početka proširivanja tog rešenja, a kasnije nastavlja sama pretragu prostora rešenja. Iteracija se sastoji od niza letova unapred i letova unazad, a njom se dobijaju jedno ili više dopustivih rešenja. Metaheuristikom Optimizacija kolonijom pčela problemi kombinatorne optimizacije rešavaju se u etapama. Svakoju etapi odgovara po jedna promenljiva. Neka je m unapred odabrani (konačan) broj etapa, a $ST = \{st_1, st_2, \dots, st_m\}$ skup tih etapa. Sa B označen je broj pčela koje učestvuju u procesu pretrage, a sa I ukupan broj iteracija. S_j ($j=1,2,\dots,m$) predstavlja skup parcijalnih rešenja u etapi st_j .

Sledećim pseudo-kodom može se opisati Optimizacija kolonijom pčela:

Korak 1: Inicijalizacija.

Zadati broj pčela B i broj iteracija I . Izabрати skup etapa $ST = \{st_1, st_2, \dots, st_m\}$. Naći početno dopustivo rešenje x posmatranog problema i postaviti ga kao najbolje.

Korak 2: Neka je $i=1$. Sve dok nije ispunjen uslov $i=I$, ponavljati sledeće korake:

Korak 3: Neka je $j=1$. Sve dok nije ispunjen uslov $j=m$, ponavljati sledeće korake:

Let unapred: Dozvoliti pčelama da izlete iz košnice i odaberu B parcijalnih rešenja iz skupa parcijalnih rešenja S_j u etapi st_j .

Let unazad: Vratiti sve pčele u košnicu. Omogućiti pčelama da steknu uvid u kvalitet svih postignutih parcijalnih rešenja i da svaka od njih odluči da li da odustane od svog parcijalnog rešenja i postane neopredeljeni sledbenik, nastavi da proširuje postignuto parcijalno rešenje bez podsticanja neopredeljenih sledbenika da prihvate njeno parcijalno

rešenje ili da svojom igrom, koju izvodi u prostoru za igru, ubedi neku od neopredeljenih pčela da zajedno proširuju postignuto parcijalno rešenje. Postaviti da je $j=j+1$.

Korak 4: Ako je najbolje rešenje otkriveno u i -toj iteraciji bolje od do tada pronađenog najboljeg rešenja, postaviti da je najbolje poznato rešenje $x = x_i$

Korak 5: Postaviti da je $i=i+1$.

Zadati broj iteracija ne mora biti jedini uslov za prekid letova pčela. Pčele se mogu pustiti da lete unapred i unazad zadati broj puta ili da između dva uzastopna poboljšanja kriterijumske funkcije izvrše unapred određeni broj letova.

Iz metaheuristike Optimizacija kolonijom pčela mogu se izvesti različiti heuristički algoritmi. Svakim od njih može se opisati specifično ponašanje pčela i njihovo rezonovanje npr. kako pčele donose odluku da napuste ili zadrže svoje parcijalno rešenje.

4. Rešavanje backup procesa primenom Optimizacije kolonijom pčela

U datoj mreži servera može se uočiti jedna zatvorena oblast kojoj pripadaju svi čvorovi koji predstavljaju servise. Košnica je locirana u mreži izvan svih čvorova. Svakoj pčeli se dozvoli da izleti iz košnice i da počne sa pretragom oblasti dopustivih rešenja. Neka je V_i korist koju pčela ima ako odabere i -ti čvor za opslugu. Usvojeno je da je korist koju pčela ima prilikom odabira nekog čvora utoliko veća ukoliko je vreme utrošeno na backup podataka manje i ukoliko je važnost podataka veća.

Neka je p_i verovatnoća da će pčela odabrati i -ti čvor za određeni server. Model izbora je LOGIT model, a verovatnoća izbora je:

$$p_i = \frac{e^{-V_i}}{e^{-V_1} + e^{-V_2} + \dots + e^{-V_n}} \quad (1)$$

gde je n broj čvorova.

Kasnije se, prilikom novog leta unapred, izuzimaju već odabrani čvorovi.

Po povratku pčela u košnicu za svaku od njih vrši se izračunavanje ukupnog vremena utrošenog na backup podataka iz izabranih čvorova, a zatim vrši upoređivanje tih parcijalnih rešenja.

Neka je Π_i ($i=1, b$, gde je b broj pčela) ukupno vreme utrošeno na backup podataka svih servera u čvorovima koje je generisala i -ta pčela. Neka je Π_{norm_i} normalizovana vrednost te veličine. Ona se izračunava:

$$\Pi_{norm_i} = \frac{\Pi_{max} - \Pi_i}{\Pi_{max} - \Pi_{min}}, \quad \Pi_{norm_i} \in [0, 1], \quad i=1, b \quad (2)$$

gde su Π_{max} i Π_{min} najveća i najmanja vrednost ukupnog vremena potrebnog za backup podataka. Usvojeno je da se verovatnoća da će pčela na početku sledećeg leta unapred biti lojalna svome da tada otkrivenom rešenju izračunava na sledeći način:

$$p_i^{u+1} = e^{\frac{\Pi_{norm_{max}} - \Pi_i}{u}}, \quad i=1, b \quad (3)$$

gde je u redni broj leta unapred.

Na osnovu tih verovatnoća utvrđuje se da li će pčela nastaviti let ili postati neopredeljena. Pčele koje ostaju pri svom parcijalnom rešenju, ubeđuju neopredeljene pčele da ih slede na njihovim putevima. Ako je broj opredeljenih pčela u određenom momentu jednak r ,

verovatnoća da će se u narednom letu neopredeljena pčela pridružiti j -toj opredeljenoj pčeli jednaka je:

$$p_j = \frac{e^{-\Pi_{norm_j}}}{e^{-\Pi_{norm_1}} + e^{-\Pi_{norm_2}} + \dots + e^{-\Pi_{norm_r}}} \quad (4)$$

Na osnovu tih verovatnoća određuje se sa kojom opredeljenom pčelom neopredeljena pčela započinje novi let. Pčele tada zajedno lete do poslednjeg čvora puta u parcijalnom rešenju koje je generisala opredeljena pčela. Posle toga svaka samostalno bira naredni skup čvorova. Svaka iteracija daje određeno rešenje. Bira se najbolje postignuto rešenje tokom zadatog broja iteracija.

5. Numerički primer

Data je logička mreža servera na kojima je pušteno u rad 14 različitih servisa nad kojima bi trebalo izvršiti backup proces. Količina, važnost podataka, stepen kompleksnosti i intenzitet backup procesa koji su vezani za navedene servise dati su u tabeli 1. Sistem poseduje tri backup servera čiji je prosečan kapacitet storniranja podataka dat u tabeli 2. Pretpostavlja se da je prostor predviđen za backup raspoloživ u momentu zahteva za tim procesom.

Potrebno je da se uz pomoć raspoloživih servera izvrši backup proces za što kraće vreme, u periodu kada na računarskoj mreži ne postoje bitne aktivnosti, a to je vreme između 3 i 5:30 ujutro.

Tabela 1. Karakteristike servisa (Backup intenzitet predstavlja količinu podataka koja može biti uskladištena u jedinici vremena direktno ili u kompresovanom obliku)*

Servisi	Količina podataka	Vажnost podataka	Stepen kompleksnosti	Backup intenzitet* MB/s
S1	19,5 GB	10	17	3
S2	34,4 GB	10	14	6
S3	2 GB	10	14	6
S4	3,1 GB	10	17	3
S5	3,4 MB	10	15	5
S6	14,6 MB	10	15	5

S7	135 GB	10	5	15
S8	4 GB	10	16	4
S9	393 MB	10	13	7
S10	4,8 GB	10	16	4
S11	600 MB	9	16	4
S12	950 MB	8	11	9
S13	120 MB	6	15	5
S14	140 MB	3	10	10

Tabela 2. Serveri i njihov kapacitet

Backup serveri	Kapaciteti backup servera
I server	500 GB
II server	100 GB
III server	150 GB

Rešavanjem problema backup procesa podataka raspoređuju se raspoloživi serveri na postojeće servise. Rešenje problema koje se dobija primenom Optimizacije kolonijom pčela dato je u tabeli 3. Treća kolona tabele 3 sadrži klastere koji se sastoje od pojedinih servisa. U četvrtoj koloni je vreme koje svaki server utroši prilikom izvršavanja backup procesa. Bitno je napomenuti da izraz *klaster* u ovom slučaju ima matematičko značenje, odnosno nije vezan za istovetni izraz koji se javlja u teoriji računarskih mreža i servera.

Tabela 3. Rešenje numeričkog primera

Backup serveri	Količina uskladištenih podataka	Servisi	Vreme backup procesa
I server	135 GB	S7	2.56 h
II server	27.34 GB	S3, S10, S1, S12, S13	2.32 h
III server	42.63 GB	S5, S6, S9, S8, S4, S2, S11, S14	2.27 h

6. Zaključak

U radu je razmatran problem alokacije rezervnih kopija podataka na više backup servera koji su prisutni na mreži. Definisan je ukupan broj servisa nad kojima se vrši backup proces, kao i njihovi parametri koji se odnose na količinu podataka, njihovu kompleksnost, procenjenu brzinu transfera (backup intenzitet), koja zavisi od tehnologije obrade posmatranih podataka (kopiranje fajlova, dumping SQL baza i sl.), kao i od sertifikovanih brzina prenosa podataka na posmatranoj LAN mreži. Empirijski je utvrđeno da na mreži nema aktivnih korisnika ili ih ima zanemarljivo malo u periodu od 2,5h, i to u periodu od 3 do 5:30 ujutru, posmatrano na dnevnom nivou. Problem je rešavan primenom metaheuristike Optimizacija kolonijom pčela, sa funkcijom cilja koja se odnosila na minimizaciju vremena trajanja celokupnog backup procesa.

Rezultat primene navedene heuristike je pokazao da, u navedenom primeru, nije moguće izvršiti kompletan backup u željenom vremenu, tj. kompletan backup može biti izveden u intervalu trajanja 2.56h. Navedeni interval, u praksi, može biti prihvatljiv jer, sa jedne strane, ne odskaje drastično od zadate vrednosti, a sa druge strane u tom intervalu se nalazi prihvatljivo mali broj aktivnih korisnika u sistemu.

Dati rezultat se može poboljšati uvođenjem određenih promena u infrastrukturi posmatrane računarske mreže. Naime, posmatrana LAN mreža poseduje sertifikovanu 100Mbps pasivnu i aktivnu opremu. Investicionim ulaganjem se aktivna mrežna oprema, kao i pasiva koja povezuje serverska čvorišta može prilagoditi za primenu 1Gbps brzina, što za direktan rezultat ima prosečno 10-tostruko uvećanje transfera podataka, a samim tim i prosečno 10-tostruko skraćanje vremenskog intervala, potrebnog za izvršavanje celokupnog backup procesa.

Predstavljeni problem, može naći svoju primenu u okviru *Network Management* sistema jer parametri definisanog zadatka mogu biti automatski prikupljeni. Primenom predstavljene metaheuristike se može formirati zadovoljavajući raspored full-backup kopija na backup serverima, takav da plan slobodnog prostora, uz prisustvo inkrementalnih backup-ova, zadovolji ograničenja definisana u zadatku u zadovoljavajućem vremenskom periodu. Ako slobodan prostor na bilo kom od backup servera padne na nivo manji od 15%, *Network Management* sistem može automatski startovati program kojim bi se automatski definisali parametri narednog ciklusa alokacija backup kopija. Navedeno rešenje bi, u tom slučaju, predstavljalo značajnu pomoć administrativnom osoblju koje radi na posmatranoj računarskoj mreži.

Literatura

- [1] D. Teodorović, "Swarm intelligence systems for transportation engineering: Principles and applications", Transportation Research Part C (2008), doi:10.1016/j.trc.2008.03.002.
- [2] P. Lučić, D. Teodorović, "Bee system: modeling combinatorial optimization transportation engineering problems by swarm intelligence". In: Preprints of the TRISTAN IV Triennial Symposium on Transportation Analysis, Sao Miguel, Azores Islands, Portugal, (2001), p. 441–445.
- [3] S. Nakrani, C. Tovey, "On honey bees and dynamic server allocation in internet hosting centers", International Society for Adaptive Behavior (2004), Vol 12 (3-4), p. 223-240

Abstract: *In this paper backup allocation problem is considered. The problem is solved by the metaheuristic approach Bee Colony Optimization (BCO). Some backup process basics, BCO metaheuristic algorithm and numerical example are presented.*

Keywords: *backup, services, metaheuristic algorithms, bee colony optimization*

BACKUP ALLOCATION PROCESS: BEE COLONY OPTIMIZATION APPLICATION

Ranko Nedeljković, Slobodan Mitrović, Dragana Drenovac