(12) **PATENTTIJULKAISU**
PATENTSKRIFT

FI0001181678

(10) **FI 118167 B**

(45) Patentti myönnetty - Patent beviljats **31.07.2007**

(51) Kv.lk. - Int.kl.

G06F 9/46 (2006.01)
H04L 12/24 (2006.01)

(21) Patenttihakemus - Patentansökning **20041380**

(22) Hakemispäivä - Ansökningsdag **26.10.2004**

(24) Alkupäivä - Löpdag **26.10.2004**

(41) Tullut julkiseksi - Blivit offentlig **27.04.2006**

SUOMI – FINLAND
(FI)

PATENTTI- JA REKISTERIHALLITUS
PATENT- OCH REGISTERSTYRELSEN

(73) Haltija - Innehavare

1 •Konsultointi Martikainen Oy, Kaskenpolttajantie 21 B 2, 00670 Helsinki, SUOMI - FINLAND, (FI)

(72) Keksijä - Uppfinnare

1 •Martikainen,Olli, Kaskenpolttajantie 21 B, 00670 Helsinki, SUOMI - FINLAND, (FI)
2 •Naumov,Valeriy A., Hakoportaankatu 4 A 4, 53850 Lappeenranta, SUOMI - FINLAND, (FI)

(54) Keksinnön nimitys - Uppfinningens benämning

Menetelmä tehtävien jakamiseksi
Förfarande för tilldelning av uppgifter

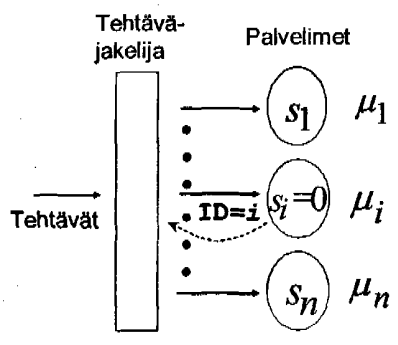
(56) Viitejulkaisut - Anförda publikationer

US 5991808 A,
Banawan, Zeidat, **"A comparative study of load sharing in heterogeneous multicomputer systems", Proc 25th** Annual Simulation Symposium, pp 22-31, 1992 (luku 3), Mitzenmacher, "How useful is old information?" IEEE Transactions on Parallel and Distributed Systems, 11, 1, pp 6-20, 2000

(57) Tiivistelmä - Sammandrag

Keksinnön kohteena on menetelmä palvelimen valitsemiseksi järjestelmässä, johon kuuluu ainakin yksi tehtäväjakelija ja useita palvelimia, jossa järjestelmässä uuden tehtävän saapuessa tehtäväjakelija osoittaa sen yhdelle näistä palvelimista. Keksinnölle ominaista on, että palvelinten valinta, minkä suorittaa tehtäväjakelija, perustuu palvelinten tehtäväjakelijalle lähettämään IPN (Idle Period Notification) informaatioon.

Uppfinningen ger en metod att välja servicestället i ett system som innehåller minst en uppgiftdistributör och många serviceställen, och i vilket en ny inkommande uppgift hänvisas till en av dessa serviceställen som uppgiftdistributören anger. Ett karakteristika för uppfinningen är att valet av serviceställen, som utförs av uppgiftdistributören, beror på IPN (Idle Period Notification) informationen som serviceställen sänder till uppgiftdistributören.



Joukko palvelimia, jotka käyttävät IPN-menetelmää

MENETELMÄ TEHTÄVIEN JAKAMISEKSI

YHTEENVETO

Suorituskyvyn optimointia tarvitaan useiden nykyaikaisten informaatio- ja kommunikaatioteknologiaan perustuvien järjestelmien tehokkaaseen hallintaan. Tällaisia järjestelmiä ovat esimerkiksi tietoliikenneverkot, tietojenkäsittelykeskukset tai työnkulkuun liittyvät järjestelmät. Tärkeä osa optimointia on dynaaminen tehtävien jako palvelimille.

Tässä selostuksessa tarkastellaan tehtävien jakamista palvelinklusterille, jossa kullakin palvelimella on erilainen prosessointikapasiteetti. Esitämme uuden *dynaamisen IPN (Idle Period Notification)* –menetelmän, jossa kunkin palvelimen tarvitsee ilmoittaa vain vapautumisensa tehtäväjakelijalle. Osoitamme simulaation avulla, että tämä menetelmä toimii yhtä hyvin kuin minimivasteaikamenetelmä (Minimum Response Time policy), joka vaatii välittömän tiedon jokaisen palvelimen tilasta jokaisen tehtävän saapumishetkellä kyseiseen palvelimeen. IPN-menetelmän etu on, että se ei vaikuta palvelimien tehtävänsuoritusviiveeseen.

HAKUTERMIT: hajautettu järjestelmä, tehtävien jakaminen

JOHDANTO

Tehtävien reititysmenetelmä on tärkeä osatekijä, joka vaikuttaa järjestelmän suorituskykyyn, koska sen avulla koordinoidaan palvelimien prosessointikapasiteettia. Nopeat Web-klusterit [1] ja Internet-reitittimet [2] soveltavat erilaisia tehtävien reititysmenetelmiä. Tehtävien reititysmenetelmiä sovelletaan myös liiketoiminta-transaktioiden ja työnkulun ohjauksen optimointiin. Tärkeä osatekijä tehtävien reititysmenetelmän määrittelyä on tieto, jonka se tarvitsee toimiakseen. Yleisesti dynaamiset menetelmät käyttävät aikariippuvaa tietoa ja staattiset menetelmät aikariippumatonta tietoa järjestelmästä [3].

Tarkastelemme tehtävien jakeluongelmaa palvelinklusterissa, jossa palvelimilla on erilaiset palveluprosessointiajat. Tehtävät saapuvat tehtäväjakelijaan, jonka vastuulla on niiden jakelu FIFO-palvelimille noudattaen tehtävien jakelumenetelmää, joka ottaa huomioon resurssien saatavuuden palvelimissa. Kiinnostuksemme kohteena ovat menetelmät, jotka minimoivat keskimääräisen kaikkien tehtävien palveluajan.

Tehtävien jakelumenetelmän tärkeä osa on sen toimintaansa tarvitsema tieto. Staattiset menetelmät käyttävät aikariippumatonta tietoa järjestelmästä, kuten tehtävien saapumisintensiteetti ja tehtävien palveluaikojen todennäköisyysjakauma. Tyypillisesti oletetaan, että tehtävien saapumisprosessi on Poisson-jakautunut. Buzen ja Chen [4] johtivat optimaaliset tulointensiteetit λ_i palvelinklusterin palvelimille, joissa ovat yleiset palveluaikajakaumat. Ni ja Hvang [5] kehittivät suljettua muotoa olevan ratkaisun tapauksessa, jossa palveluajat ovat eksponentiaalisesti jakautuneet. Tantawi ja Towsley [6] tutkivat mielivaltaisesti kytkettyä hajautettua järjestelmää. Siinä tehtävät voivat saapua mihin tahansa palvelimeen ja ne voidaan palvella joko paikallisesti tai lähettää edelleen toiselle palvelimelle. He johtivat iteratiivisen algoritmin, joka määrittelee optimaalisen staattisen tehtävien jakelumenetelmän, jota myöhemmin ovat kehittäneet edelleen Kim ja Kameda [7].

Dynaamiset tehtävien jakelumenetelmät käyttävät hyväkseen kunkin hetkistä globaalia systeemin tilatietoa jakaakseen kuorman palvelimien välillä. Tyypillisessä tapauksessa uuden tehtävän saapuessa hetkellä t arvioidaan jokaiseen palvelimeen i liittyvä kustannusfunktio $c_i(t)$. Palvelin, jonka kustannusfunktio saa pienimmän arvon, valitaan suorittamaan tehtävää. Esimerkiksi minimivasteaikamenetelmässä (Minimum Response Time policy, MRT) kustannusfunktio on uuden tehtävän palveluajan odotusarvo, joka arvioidaan kaavalla $c_i(t) = (s_i(t) + 1)/\mu_i$, missä $s_i(t)$ on tehtävien määrä palvelimessa i mukaan luettuna hetkellä t saapunut tehtävä ja μ_i on palvelimen prosessointi-intensiteetti [8]. On kuitenkin epärealistista olettaa, että todellinen tehtäväjakelija pystyisi hankkimaan kunkin hetkisen globaalin tilatiedon.

Mitzenmacher on tutkinut viitteessä [9] erilaisia vanhentunutta tilatietoa käyttäviä malleja sovellettuna identtisien palvelimien klusteriin olettaen eksponentiaaliset tehtävien palveluprosessointiajat. *Periodisessa päivitysmallissa* globaali tilatieto

päivitetään T sekunnin välein. Tässä tapauksessa MRT menetelmä toimii huonosti ellei T ole hyvin pieni, mikä johtuu laumailmiöstä: kaikki kahden mittaushetken välillä saapuvat tehtävät ohjataan samaan palvelimeen. Menetelmät, jotka käyttävät vain pientä osaa tilatiedosta, voivat toimia paremmin. Menetelmä, jossa valitaan lyhin palveluaika kahdesta satunnaisesti valitusta palvelimesta, on havaittu toimivan erittäin hyvin. *Jatkuvassa päivitysmallissa* globaali tilatieto päivitetään jatkuvasti, mutta on X sekuntia jäljessä todellisesta tilasta kaikilla saapumishetkillä, missä X on kaikista tehtävistä riippumaton satunnaismuuttuja. Jos X on kiinteä vakio T, niin jatkuvaa päivitysmallia soveltava järjestelmä toimii samalla tavalla kuin jos se käyttäisi periodista päivitysmallia. Sen sijaan jos X on eksponentiaalisesti tai tasaisesti jakautunut satunnaismuuttuja, tulokset ovat paljon parempia.

Edelleenkin on avoin kysymys paljonko tietoa ja millainen tiedon päivitystaajuus ovat riittävät hyvän suorituskyvyn takaamiseksi. Tässä selityksessä ehdotamme uutta dynaamista *IPN (Idle Period Notification)* –menetelmän, jossa kunkin palvelimen tarvitsee ilmoittaa vain vapautumisensa tehtäväjakelijalle (Kuva 1). Osoitamme, että tämä menetelmä toimii yhtä hyvin kuin MRT-menetelmä, joka tarvitsee välittömän tiedon jokaisen palvelimen tilasta jokaisella hetkellä kun tehtäviä saapuu palvelimille.

IPN-MENETELMÄ

Ehdotettavassa *IPN (Idle Period Notification)* –menetelmässä jokainen palvelin ilmoittaa tehtäväjakelijalle vapautumishetkensä lähettämällä palvelintunnisteensa (ID). Tehtäväjakelija voi käyttää näitä ilmoituksia laskeakseen varattujaksojen kestot, varattujaksoina palvelimille osoitettujen tehtävien määrät ja palveluprosessointinopeudet, jos niitä ei muuten tunneta.

Yleisesti on melko vaikeaa laskea tarkkaa vasteaikojen odotusarvoa käyttämällä lähtökohtana toteutuneita palvelimien varattujaksojen kestoja ja niille osoitettujen tehtävien määriä. Oletetaan, että uusi tehtävä saapuu palvelimelle hetkellä t. Tarkastellaan palvelimen tämänhetkistä varattujaksoa ja oletetaan yksinkertaisuuden vuoksi, että se alkaa hetkellä 0, ja että yhteensä N tehtävää on saapunut ennen

hetkeä t. On tunnettua, että FIFO-palvelimen tehtävän r odotusaika w_r toteuttaa Lindley:n yhtälön $w_{r+1} = (w_r + b_r - a_r)^+$, missä b_r on tehtävän palveluaika, a_r on tehtävien r ja $r+1$ saapumishetkien väliaika, ja $x^+ = x$, jos $x > 0$, ja muulloin $x^+ = 0$ [10]. Koska kaikilla tehtävillä, jotka saapuvat välillä $(0,t]$, on nollasta poikkeava odotusaika, niiden palveluajat ja saapumisten väliajat noudattavat seuraavia epäyhtälöitä

$$\sum_{i=1}^{r} b_i > \sum_{i=1}^{r} a_i \text{, for } r = 1,2 \ldots N,$$

ja tehtävän $N+1$ poistumisaika palvelusta saadaan kaavasta

$$\delta_{N+1} = \sum_{i=1}^{N+1} b_i .$$

Tästä seuraa, että hetkellä t saapuvan tehtävän poistumisajan tarkka ehdollinen odotusarvo, edellyttäen että sen järjestysnumero on $N+1$ kyseisellä varattuajalla, voidaan laskea yhtälöstä

$$d_{N+1}(t) = \frac{1}{\mu} + E\left(\sum_{i=1}^{N} b_i \mid \sum_{i=1}^{r} b_i > \sum_{i=1}^{r} a_i, r = 1,2 \ldots N, \sum_{i=1}^{N} a_i = t\right),$$

missä μ on tehtävän saama prosessointi-intensiteetti.

Vasteajan odotusarvon sijasta ehdotamme huomattavasti yksinkertaisempaa kustannusfunktiota. Ehdotetussa IPN-menetelmässä tehtäväjakelija yrittää saada kaikkien palvelimien työmäärät yhtä suuriksi jokaisen palvelimen kuluvan varattujakson puitteissa. Määritellään $T_i(t)$ kaavalla $T_i(t) = \sup\{\tau < t \mid s_i(\tau) = 0\}$, missä $s_i(\tau)$ on palvelimessa i hetkellä τ olevien tehtävien määrä. $T_i(t)$ on palvelimen i tämänhetkisen varattujakson alkuhetki, jos se on varattu hetkellä t, muuten $T_i(t) = t$. Olkoon $N_i(t)$ palvelimelle i ohjattujen tehtävien määrä aikavälillä $[T_i(t), t)$. Jos uusi tehtävä saapuu hetkellä t, silloin tehtäväjakelija ohjaa sen palvelimelle i, jonka kustannusfunktion arvo

$$c_i(t) = \frac{N_i(t)+1}{\mu_i}$$

on pienin.

SIMULOINTIESIMERKKEJÄ

Simuloimme kolmea erilaista järjestelmää käyttäen Poisson-saapumisprosessia viitteen [11] tapaan, ja vertaamme MRT- ja IPN-menetelmiä. Ensimmäisessä järjestelmässä nimeltään System 1 on 10 solmua. Solmujen prosessointi-intensiteetit ovat $\mu_1 = 6$, $\mu_2 = \mu_3 = \ldots = \mu_{10} = 1$. Toisessa järjestelmässä, System 2:ssa, on myös 10 solmua. Solmujen prosessointi-intensiteetit noudattavat aritmeettista sarjaa kaavan $\mu_i = 3\left(1 - \frac{i}{11}\right)$, $i = 1,2 \ldots 10$, mukaisesti. Kummankin näiden järjestelmien kokonaisprosessointi-intensiteetti on 15. Kolmannessa järjestelmässä, System 3, on 8 solmua, ja niiden prosessointi-intensiteetit noudattavat geometrista sarjaa $\mu_i = 2^{10-i}$, $i = 1,2 \ldots 10$.

Kuvissa 2 – 3 on esitetty prosessoitujen tehtävien keskimääräiset vasteajat MRT-mallissa (yhtenäiset viivat) ja IPN-mallissa (pisteviivat) tehtävien saapumisintensiteetin λ funktiona. Jos oletamme järjestelmissä eksponentiaalisesti jakautuneet palveluajat, voimme laskea myös optimaalisen staattisen menetelmän keskimääräiset vasteajat (katkoviivat). Kuten odotettiin, molemmat dynaamiset tehtävien jakomenetelmät antavat paremman suorituskyvyn kuin optimaalinen staattinen menetelmä. Yllättävää kyllä, ehdotettu IPN-menetelmä toimii yhtä hyvin kuin MRT-menetelmä jopa raskaalla kuormalla, jolloin varattuajat ovat hyvin pitkiä.

JOHTOPÄÄTÖKSET

Tässä selityksessä esittelemme dynaamisen tehtävienjakelumenetelmän, jolla näyttää kuluttavan vähiten resursseja. Ehdotetussa IPN (Idle Period Notification) – menetelmässä jokainen palvelin lähettää vain yhden tilatiedon päivityksen tehtäväjakelijalle varattujaksoaan kohden – IPN-ilmoituksen vapautuessaan – ja siihen sisältyy ainoana informaationa palvelimen tunniste (ID). Tästä huolimatta IPN-menetelmä toimii hyvin laajalla järjestelmän parametrialueella.

Toinen tämän menetelmän etu on, että IPN-ilmoitukset lähetetään palvelinten vapautuessa. Näin ollen IPN-menetelmä ei kuormita palvelimia niiden prosessoidessa tehtäviä.

VIITTEET

[1] V. Cardellini and E. Casalicchio. The State of the Art in Locally Distributed Web-Server Systems, *ACM Computing Surveys*, Vol. 34, No. 2, June 2002, pp. 263–311.

[2] O. Younis and S. Fahmy. Constraint-Based Routing in the Internet: Basic Principles and Recent Research, *IEEE Communications Surveys*, Vol. &, No. 1, 2003, pp. 2-13.

[3] T.L. Casavant and J.G. Kuhl. A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems, *IEEE Transactions on Software Engineering*, Vol. 14, No. 2, February 1988, pp. 141-154.

[4] J.P. Buzen and P.P.-S, Chen,"Optimal load balancing in memory hierarchies", in *Information Processing 74* , J.L. Rosenfeld, Ed., New York: North-Holland, pp. 271–275, 1974.

[5] L.M. Ni and K. Hwang, "Optimal load balancing in a multiple processor system with many job classes", *IEEE Transactions on Software Engineering*, vol. 11, no. 5, 1985, pp. 491–496.

[6] A.N. Tantawi and D. Towsley, "Optimal static load balancing in distributed computer systems", *Journal of the ACM*, vol. 32, no. 2, pp. 445-465, Apr. 1985.

[7] C. Kim and H. Kameda, "An algorithm for optimal static load balancing in distributed computer systems", *IEEE Transactions on Computers*, vol. 41, no. 3, pp. 381-384, March 1992.

[8] Y.C. Chow and W.H. Kohler, "Models of dynamic load balancing in a heterogeneous multiple processor system". *IEEE Transactions on Computers*, vol. C-28, no. 5, pp. 354-361, May 1979.

[9] M. Mitzenmacher, "How useful is old information?". *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, no. 1, pp. 6-20, Jan. 2000.

[10] D.V. Lindley, "On the theory of queues with a single server", *Proc. of the Cambridge Philosophical Society,* vol. 48, pp. 277-289, 1952.

[11] S.A. Banawan and N.M. Zeidat, "Comparative study of load sharing in heterogeneous multicomputer systems", *Proc. 25th Annual Simulation Symposium*, Orlando, Florida, USA, pp. 22-31, Apr. 6-9, 1992.

PATENTTIVAATIMUKSET

1. Menetelmä palvelimen valitsemiseksi järjestelmässä, johon kuuluu ainakin yksi tehtäväjakelija ja useita palvelimia, jossa järjestelmässä uuden tehtävän saapuessa tehtäväjakelija osoittaa sen yhdelle näistä palvelimista, ja jossa palvelinten valinta, minkä suorittaa tehtäväjakelija, perustuu palvelinten tehtäväjakelijalle lähettämään IPN- (Idle Period Notification) informaatioon, jonka palvelin lähettää joka kerta vapautuessaan tehtäväjakelijalle, ja joka IPN-informaatio sisältää ainakin palvelimen tunnisteen, *tunnettu* siitä, että tehtäväjakelija käyttää palvelimelle sen tämänhetkisen varattujakson aikana lähetettynä työmääränä lasketun varattujakson aikana palvelimelle lähetettyjen tehtävien työmäärien summaa.

2. Vaatimuksen 1 mukainen menetelmä, *tunnettu* siitä, että tehtäväjakelija lähettää uuden tehtävän palvelimelle, jolle lähetetty työmäärä tämänhetkisen palvelimen varattujakson aikana on pienin.

3. Vaatimusten 1-2 mukainen menetelmä, *tunnettu* siitä, että jos tehtäväjakelija ei tiedä tehtävien työmäärä, se voi arvioida palvelimelle lähetetyn työmäärän kertomalla palvelimelle lähetettyjen tehtävien määrän palvelimen keskimääräisellä palveluajalla.

4. Vaatimusten 1-3 mukainen menetelmä, *tunnettu* siitä, että palvelimen keskimääräinen palveluaika voidaan arvioida tehtäväjakelijassa laskemalla keskiarvo osamääristä, jotka saadaan jakamalla kukin palvelimen varattujakson kesto palvelimelle sen aikana lähetettyjen tehtävien määrällä.

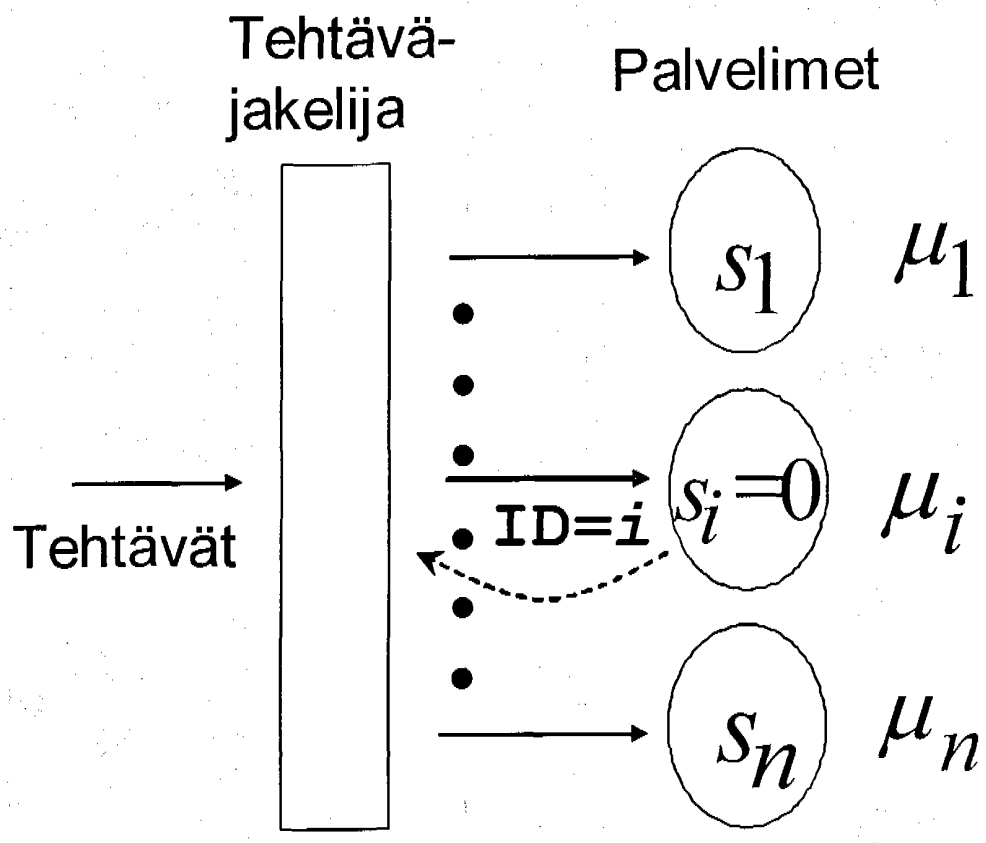
PATENTKRAV

1. En metod för att välja servicestället i ett system som innehåller minst en uppgiftdistributör och många serviceställen, och i vilket en ny inkommande uppgift hänvisas till en av dessa serviceställen som uppgiftdistributören anger, och i vilket valet av serviceställen, som utförs av uppgiftdistributören, beror på IPN (Idle Period Notification) informationen som serviceställen sänder till uppgiftdistributören varje gång serviställen blir ledig, och vilket IPN informationen innehåller åtminstone identifieringen av serviställen, *kännetecknat av*, att arbetsmängden associerat till serviceställen under dess nutida reserveringsperiod är summan av arbetsmängder av alla uppgifter som har hänvisats till serviceställen under den nutida reserveringsperioden.

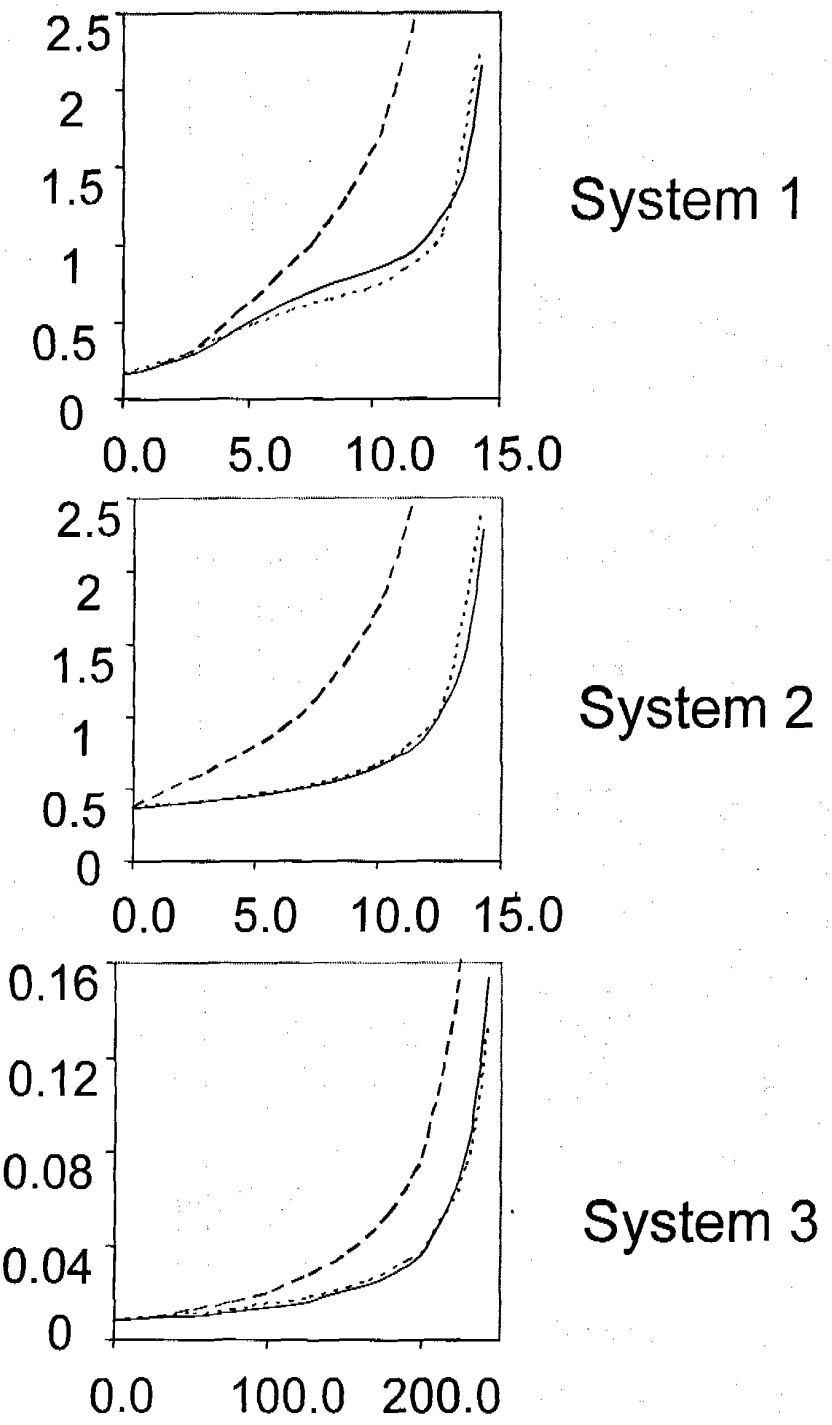
2. En metod enligt patentkravet 1, *som kännetecknas av*, att uppgiftdistributören hänvisar den nya uppgiften till servicestället, som har den minsta hänvisade arbetsmängden under dess nuvarande reserveringsperiod.

3. En metod enligt patentkraven 1-2, *som kännetecknas av*, att om uppgiftdistributören ej känner arbetsmängden av uppgifter, kan den approximera arbetsmängden associerad till ett serviceställe genom att multiplicera mängden av till serviceställen hänvisade uppgifter med den genomsnittiga servicetiden vid serviceställen.

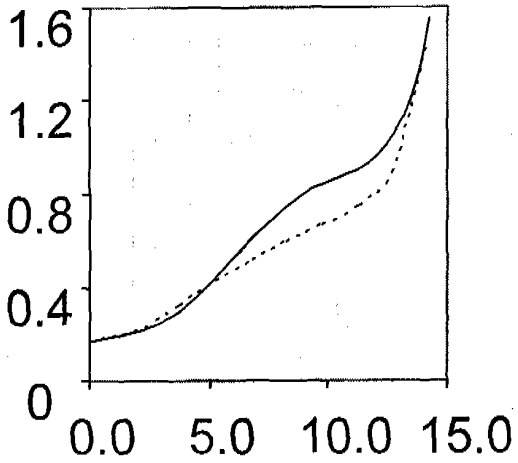
4. En metod enligt patentkraven 1-3, *som kännetecknas av*, att den genomsnittiga servicetiden vid serviceställen kan approximeras av uppgiftdistributören genom att räkna genomsnittet av delmängder, som fås genom att dividera längden av reserveringsperioden med mängden av uppgifter vid serviceställen under reserveringsperioden.



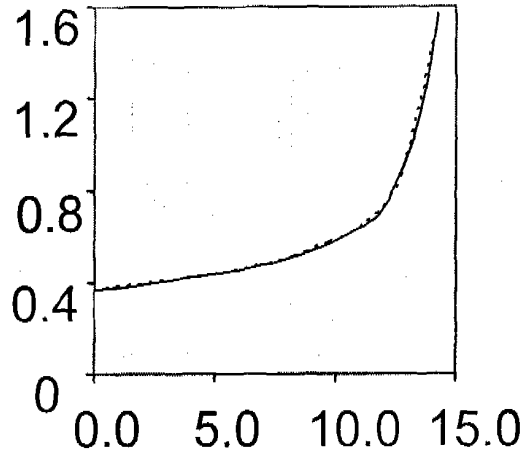
Kuva 1. Joukko palvelimia, jotka käyttävät IPN-menetelmää



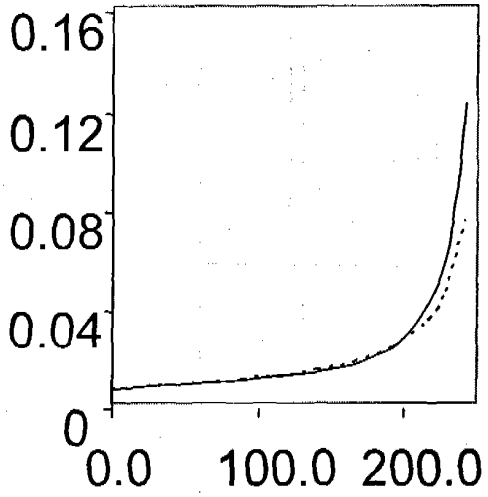
Kuva 2. Keskimääräiset vasteajat, kun palveluajat ovat eksponentiaalisesti jakautuneet



System 1



System 2



System 3

Kuva 3. Keskimääräiset vasteajat, kun palveluajat ovat vakiot