Reaaliaikaisen osakemarkkinatiedon pursketapahtumien korrelaatiot

Seminaaritutkielma

Kari Palomäki

Helsinki 22. huhtikuuta 2008

Tiedon louhinnan seminaari

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

HELSINGIN YLIOPISTO – HELSINGFORS UNIVERSITET

Tiedekunta/Osasto	Laitos – Institution
Matemaattis-luonnontieteellinen tiedekunta	Tietojenkäsittelytieteen laitos

Tekijä – Författare
Kari Palomäki

Työn nimi – Arbetets titel
Reaaliaikaisen osakemarkkinatiedon pursketapahtumien korrelaatiot

Oppiaine – Läroämne
Tietojenkäsittelytiede

Työn laji – Arbetets art	Aika – Datum	Sivumäärä – Sidoantal
Seminaaritutkielma	22.4.2008	25 sivua

Tiivistelmä – Referat

Tämä tutkielma on referaatti Michail Vlachosin, Kun-Lung Wun, Shyh-Kwei Chen ja Philip S. Yun artikkelista *Correlating burst events on streaming stock market data*.

Kirjoittajat käsittelevät korreloivien purskekuvioiden tarkkailua ja tunnistamista useamman virtauksen aikasarjatietokannasta. Tämä tapahtuu kaksivaiheisesti: ensin tunnistetaan datasta purskeet ja sitten ne talletetaan tehokkaaseen muistihakemistoon. Purskeiden tunnistuksessa käytetään vaihtelevaa kynnystasoa ja hyödynnetään useissa sovelluksissa esiintyvien vinojen jakaumien ominaisuuksia. Havaitut purskeet tiivistetään purskeaikaväleiksi ja talletetaan aikavälihakemistoon, joka mahdollistaa korreloivien purskeiden havaitsemisen hyvin tehokkaalla limityksen tunnistuksella. Alkuperäisessä lähteessä on esitetty ehdotetun hakemistojärjestelmän vaativuuden perusteellinen analyysi, mutta tätä analyysiä ei ole tässä referaatissa. Purskehakemiston vasteaika on reaaliaikakäytön tasoa. Lähestymistavan hyödyllisyyttä havainnollistetaan esimerkkiaineistona käytetystä New Yorkin pörssin osakekaupan datasta löydetyillä mielenkiintoisilla korreloivilla tapahtumilla. Menetelmää ja esitettyjä tietorakenteita voidaan käyttää myös telekommunikaation, verkkoliikenteen ja lääketieteellisen datan anomalioiden tai uusien piirteiden havaitsemiseen.

Avainsanat – Nyckelord
Aikasarjat, indeksointi, korrelaatio, osakekauppa, purskeen tunnistus, tiedon louhinta

Säilytyspaikka –Förvaringställe

Muita tietoja – Övriga uppgifter

Sisältö

Kuvat

Taulukot

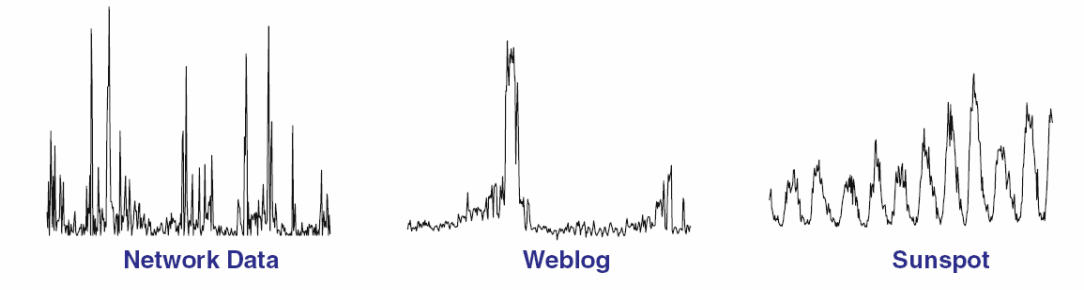
1 Johdanto

Sanotaan, että muutos on nykyään ainoa pysyvä asia. Siksi päätöksiä tehtäessä on tärkeää seurata monien mitattavissa olevien suureiden ajallista vaihtelua, myös muutosten määrää jonkin ajan kuluessa. Tätä voidaan kuvata aikasarjoilla, jolloin yhtenä muutoksen mittarina käytetään purskeisuutta, jolla tarkoitetaan, että merkitseviä tapahtumia sattuu tavallista enemmän jonkin aikakehyksen sisällä. Siksi purskeiden tunnistus voi tarjota hyödyllisen näkymän, kun halutaan päätöksenteon tueksi kiinnittää huomiota muutosten määrään.

Purskekäyttäytymisen tarkkailu ja mallintaminen on tärkeää mm. seuraavilla aloilla:

- Tietoverkoissa on tunnistettu, että tietoliikenne on voi olla purskeista erilaisilla aika-asteikoilla. Siksi purskeiden tunnistus on tärkeää, kun tunnistetaan verkon pullonkaulakohtia tai havaitaan törmäystilanteita. Tulevien datapakettien paljon tavallista suurempi määrä voi kertoa, että tietoverkossa on käynnissä jonkinlainen häirintähyökkäys.
- Tietojärjestelmien lokitiedostaja analysoitaessa purskekäyttäytymisen havaitseminen on hyödyllistä ongelmien tai pullonkaulojen havaitsemiseksi. Purskeisuutta on myös käytetty web-blogien samanlaisuuden mittana.
- Huijausten paljastussovellutuksissa ja vastaavissa on hyvin kriittistä tunnistaa tehokkaasti mikä tahansa epänormaali aktiivisuus (tavallisesti jonkun resurssin ylikäyttönä). Purskeiden havaitsemistekniikoita voidaan tuottavasti hyödyntää tuomaan esille epäilyttävää aktiivisuutta suurivolyymisessä osakekaupankäynnissä tai tunnistamaan puhelinliikenteen huijauksia.
- Luonnontieteissä tutkijoita kiinnostaa purskekäyttäytymisen paljastaminen kosmisesta säteilystä, kuten gammasäteistä tai auringonpilkkuina ilmenevästä auringon aktiivisuudesta, koska näitä mittaustuloksia voidaan käyttää todisteina tulevasta ilmaston muutoksesta. On esimerkiksi huomattu, että auringon säteilyn vaihtelu suuresti vaikuttaa maan ilmastoon ja että auringonpilkkujen määrän kasvu liittyy pohjoisen pallonpuoliskon maalämpötilojen nousuun.
- Epidemiologiassa ja bioterrorismissa tiedemiehiä kiinnostaa tautien määrän kasvun mahdollisimman varhainen havaitseminen. Tämä nähdään sairauksien tai lääkärissä käyntien määrän äkillisenä kasvuna jollain maantieteellisellä alueella.
- Lääketieteessä joittenkin biometristen mittausarvojen purskeisuuden havaitseminen saattaa paljastaa jonkin terveysepänormaaliuden. Esimerkiksi EEG:n purskekuvio voi olla pätevä merkki mahdollisesta aivovauriosta. Myös geenitutkimuksessa voidaan hyödyntää purskeiden tunnistusta.

Kuvassa 1 on kolme esimerkkiä purskedatasta.



Kuva 1: Aikasarjadatan purske-esimerkkejä [Vla07 s.111]

Purskeiden tunnistaminen on tärkeää, mutta monilla aloilla tietämystä löydetään tehokkaammin tunnistamalla useamman datalähteen keskenään korreloivat purskeet. Tiedon louhinnan näkökulmasta tämä tehtävä on jännittävämpi ja haasteellisempi, koska siinä on havaittava purskeklusterit ja koska se voi myös auttaa mahdollisten datavirtojen välisten pursketapahtumien kausaalisten ketjujen löytämisessä. Esimerkkejä tälläisistä ketjuista kohdataan monissa talouden ja osamarkkinoiden sovelluksissa. Purskekorrelaatioiden avulla nähdään myös hyödyllisiä yhteyksiä web-blogien ja jopa web-hakukuvioiden välillä.

Tässä tutkielmassa tarkastellaan purskeiden korrelaatioiden havaitsemista. Purskeiden tunnistamiseen esitetään uusi kriteeri, havaittujen purskeiden tallentamiseen havaitsemisaikävälien avulla, limittyvien purskeiden tunnistamiseen muistiperusteinen hakemistorakenne ja tähän liittyvä hakualgoritmi sekä näihin liittyvä lähestymistapa inkrementaalisesti ylläpitää hakemistorakennetta uusia arvoja lisättäessä. Hakemistorakenteella ja algoritmilla saavutetaan yli kolmen kertaluokan parannus purskeiden limittymisen laskennan hakutehokkuudessa B+puuhun verrattuna. Esitettyjä menetelmiä sovelletaan New Yorkin pörssin (NYSE) osakkeiden vaihdantadataan. Suorituskyvystä esitetään sekä todelliseen että simuloituun aineistoon perustuvia mittaustuloksia.

2 Ongelman kuvaus

Tarkastellaan tietokantaa D, joka sisältää m aikasarjasekvenssiä $S=s_1 \ldots s_n$, $s_i \in R$. Purskeaikaväli $b=[t^{start}, t^{end})$, joka kuvaa havaitun purskeen kestoa, johon sisältyy purskeen alkuhetki mutta ei loppuhetki. t^{start} ja t^{end} ovat kokonaislukuja ja $t^{start} < t^{end}$. Kahden purskeaikavälille q ja b määritellään limitys operaattori $\cap$ seuraavasti:

$q \cap b = 0$ jos $t_q^{end} \leq t_b^{start}$ tai $t_q^{start} \geq t_b^{end}$, muutoin = $\min(t_q^{end}, t_b^{end})$-$\max(t_q^{start}, t_b^{start})$

Purskekorrelaatio-ongelma jaetaan seuraaviin vaiheisiin:

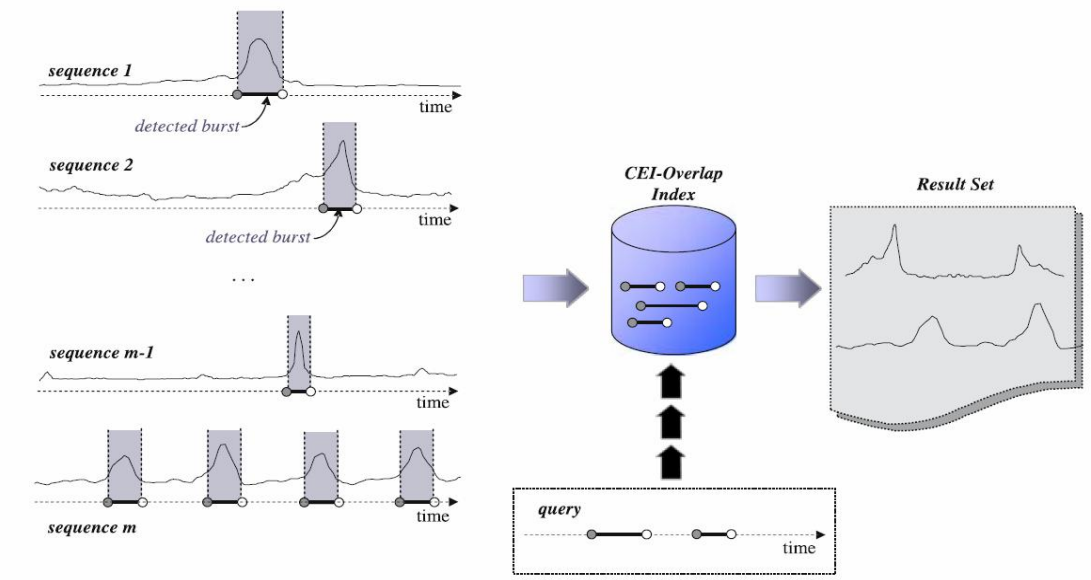
i. Tietokannassa *D* olevien purskeiden tunnistus prosessina, joka palauttaa kullekin sekvenssille *A* purskeaikavälijoukon $B_S = \{b_1, ..., b_k\}$, jossa *k* on eri kullekin sekvenssille. B^D on kaikki tietokannan *D* purskeaikavälit sisältävä joukko

ii. B^D :n järjestäminen CEI-limityshakemistolla *I*.

iii. Hakemistoon *I* limittyvien purskeiden etsintä kyselyllä *Q*, jossa *Q* on myös purskeaikavälijoukko $Q=\{q_1, ...q_l\}$. Indeksoinnin tulos on aikavälijoukko $V=\{v_1,...v_r\}$, $v_j \in B^D$ jolle $\sum\sum q_i \cap q_j \neq 0$

iv. Palautetaan k parasta sovitusta. Tämä askel sisältää palautettujen sekvenssien arvostuksen, joka perustuu limityksen asteeseen purskeaikavälien ja kyselyaikavälien välillä. Koska tässä vaiheessa on kysymys vain tulosjoukon lajittelusta, ei tätä tarkastella tämän enempää.

Kuvassa 2 on havainnollistettu näitä vaiheita ja taulukossa 1 on yhteenveto käytetyistä merkinnöistä.



Kuva 2: Yleiskatsaus käytetystä lähestymistavasta [Vla07 s.113]

Symboli	Kuvaus
D	Aikasarjatietokanta
B^D	Tietokannan D havaittujen purskeiden joukko
I	Purskehakemisto
m	Tietokannan koko (aikasarjojen määrä)
b_i	Purskeaikaväli
q_i	Kyselyaikaväli
τ	Purskeiden tunnistuskynnys
r	Indeksoinnin maksimiaika tulevaisuudessa
L	Ajan pituus hakemistoaluetta kohti

Taulukko 1: Tärkeimmät merkinnät [Vla07 s.113]

3 Purskeiden tunnistus

Purskeiden tunnistusprosessi sisältää niiden aika-alueiden tunnistamisen, joissa jokin piirre on huomattavan korostunut. Asetelmassamme tarkastelemme sekvenssin *S* hetkellistä arvoa purskeen ilmaisijana. Jos $s_i > \tau$, niin aika *i* merkitään purskeeksi. Kynnyksen τ määritys riippuu tietokannan jakauman ominaispiirteistä. Normaalijakauman tapauksessa voitaisiin asettaa τ siten että se on keskiarvo μ lisättynä kolmella keskihajonnalla.

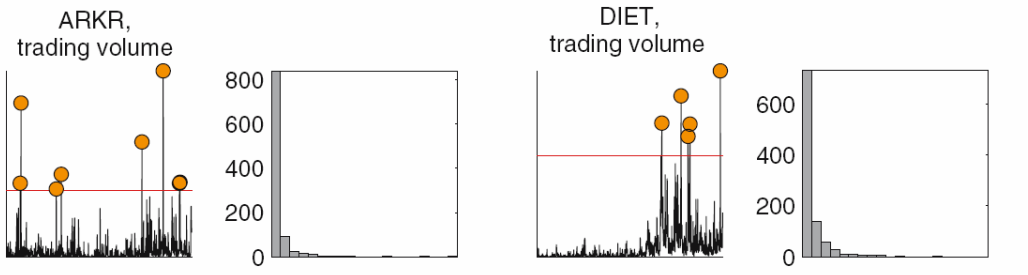
Tässä tutkielmassa esimerkkiaineistona on finanssidataa, joten tutkitaan aluksi niiden arvojen jakautumista. Kuvassa 3 on kahden osakkeen vaihdon volyymin vaihtelu ajan suhteen. Useimmille osakkeille saadaan hyvin samankaltainen jakauma. Jakauma on voimakkaasti vino. Tässä tutkielmassa jakaumaa mallinnetaan eksponentiaalisella jakaumalla, koska jakauma voidaan tällöin kuvata yhdellä parametrilla, joka voidaan helposti määrätä aikasarjan arvoista. Satunnaismuuttujan X eksponentiaalinen jakauma on

$$P(X>x)=e^{-\lambda x}$$

jossa X:n keskiarvo μ on $1/\lambda$. *x* voidaan ratkaista helposti:

$$x = -\mu \ln(P) = -(\sum s_i \ln(P))/n$$

Lasketaan kriittinen kynnysarvo, jonka ylittävät arvot siis merkitsevät pursketta, arvioimalla x jakauman loppupäästä asettamalla P hyvin pieneksi, esimerkiksi arvoon 10^{-4}. Kynnysarvo ja vastaavat purskeet esimerkkiosakkeille on merkitty kuvaan 3.

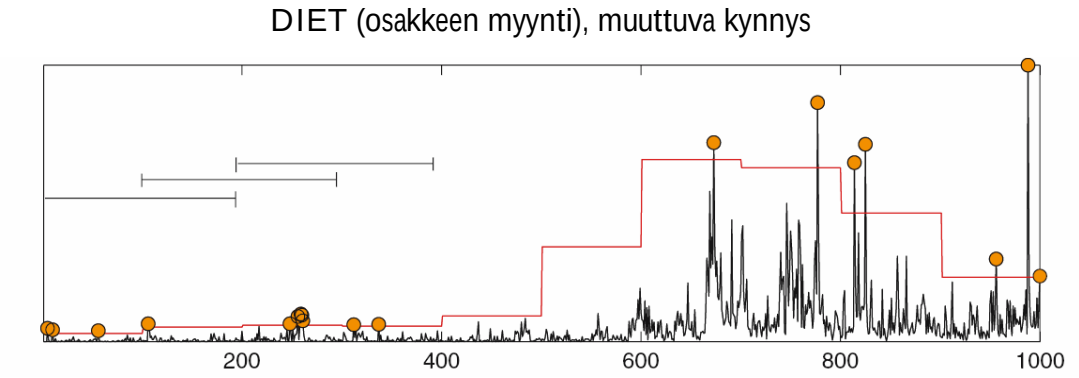


Kuva 3: Kaksi jakaumaesimerkkiä osakekaupan volyymeistä. Vasemmalla on volyymin vaihtelu yhden vuoden aikana ja oikealla vastaava jakauma. [Vla07 s.114]

On huomattava, että laskettu kynnysarvo on helppo laskea jatkuvalle aikasarjalle, koska riittää ylläpitää sekvenssin arvojen kasvavaa summaa.

Globaalin kynnysarvon käyttö saattaa kuitenkin aiheuttaa harhaa, jos arvoalue muuttuu voimakkaasti tarkkailujakson aikana. Silloin voidaan käyttää muuttuvaa kynnystä siten että data jaetaan limittyviin osiin. Jakauma kussakin osassa on edelleen hyvin vino ja niille voidaan käyttää eksponentiaalista jakaumaa. Kuvassa 4 on tästä esimerkki, jossa osien pituus on 200 ja

limitys 100. Limittyvällä osalla kynnysarvo on kahden peräkkäisen osan kynnysarvojen keskiarvo. Näin voidaan kuvan mukaisesti havaita myös vähemmän voimakkaat purskeet.



Kuva 4: Muuttuva kynnys käytettäessä limittyviä ali-ikkunoita [Vla07 s.115]

Kun sekvenssin purskeet on merkitty, kustakin havaitusta purskeesta kirjoitetaan pursketietue. Peräkkäiset purskepisteet tiivistetään purskeaikaväliksi, joka esitetään alku- ja loppuhetken avulla, esim. merkinnällä [m,n), m<n. Yksittäistä purskepistettä hetkellä m kuvaa merkintä [m,m+1). Seuraavaksi esitetään, miten nämä purskealueet voidaan organisoida tehokkaaksi hakemistorakenteeksi. Tässä käytetty purskeen määrittely sopii vinoille jakaumille, jotka kuvaavat hyvin finanssidataa. Muunlaiselle datalle voi olla mielekästä käyttää jotain muuta purskeen määrittelyä. Kuitenkin jos purskeet esitetään aikavälimuodossa, voidaan tässä kuvattua indeksointia soveltaa sellaisenaan.

4 Hakemistorakenne

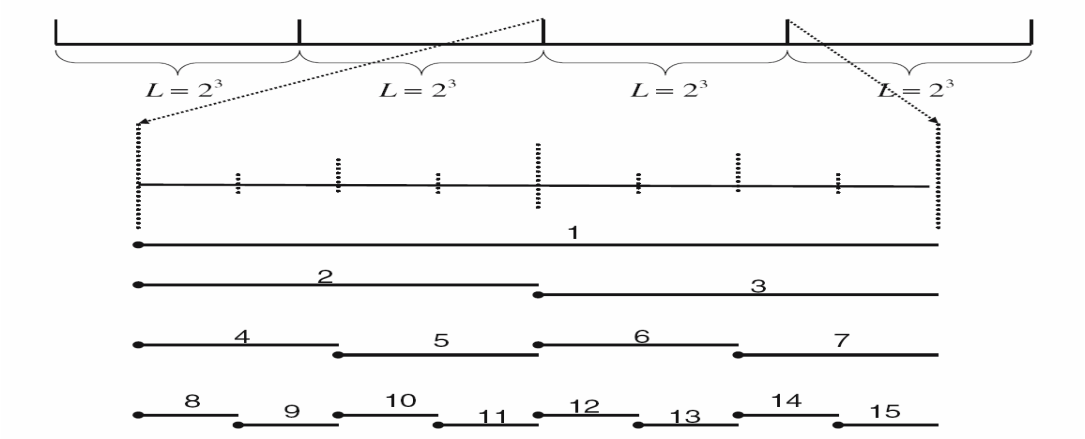
Limittyvien purskeaikavälien nopeaa tunnistamista varten otetaan käyttöön käsite CEI (containment-encoded-intervals) ja esitellään CEI-limityshakemisto. Limittyvien alueiden löytämiseen esitellään tehokas hakutekniikka. Esitellään myös tehokas tapa käsitellä päättymätöntä koko ajan etenevää aikasarjaa.

4.1 CEI-limityshakemiston muodostaminen

CEI-limitysindeksoinnissa käytetään kahta erilaista aikaväliä: (a) purskeaikavälit ja (b) virtuaaliset rakenneaikavälit. Purskeaikavälit on selitetty jo aiemmin luvussa 3. Virtuaalisia rakenneaikavälejä tarvitaan purskeaikavälien jakamiseen ja tehokkaan hakuoperaation toteuttamiseen. Sekä purske- että hakuaikavälit ilmaistaan alku- ja loppuhetken avulla, kuten aiemmin on esitetty.

Kuvassa 5 on esimerkki sisältymiskoodatuista aikaväleistä ja niiden paikallisesta tunnusotsikoinnista. Oletetaan, että koodattavat aikavälit ovat alueella [0,r). Ensin alue jaetaan

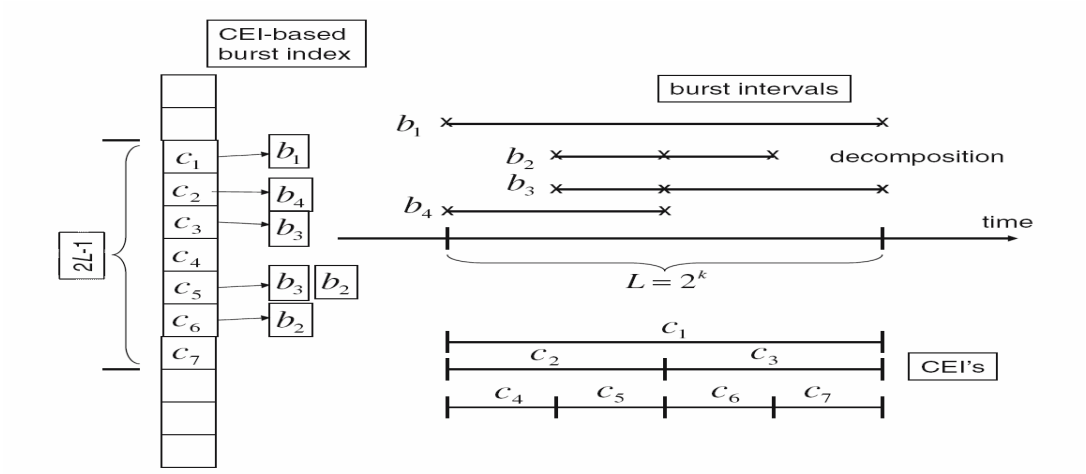
r/L :ään L:n pituiseen segmenttiin, joita merkitään SS_i , jossa i=0,1,...(r/L-1), $L=2^k$ ja k on kokonaisluku. On oletettu, että r on L:n monikerta. Yleissääntö on, että mitä suurempi purskealueiden keskiarvo on, sitä suurempi pitäisi L:n olla. Segmentti SS_i sisältää aikavälin [iL,(i+1)L). Nyt 2L-1 CEI:tä määritellään seuraavasti: (a) Määritä yksi CEI, jonka pituus on L ja joka sisältää koko segmentin; (b) Määritä rekursiivisesti 2 CEI:tä jakamalla CEI kahteen puoliskoon kunnes pituus on 1. Kuvassa 5 on näin muodostettu 15 CEI:tä.



Kuva 5: Esimerkki sisältymiskoodatuista aikaväleistä (CEI) ja niiden tunnusotsikointi [Vla07 s.116]

Näillä 2L-1:llä CEI:llä on sisältymisrelaatio keskenään. Jokainen yhden yksikön mittainen CEI sisältyy johonkin kahden mittaiseen CEI:iin, joka taas puolestaan sisältyy johonkin neljän yksikön mittaiseen CEI:hin jne. CEI nimi koodataan sisältymisrelaatiolla. CEI:n tunnisteessa on kaksi osaa: segmentin tunniste ja paikallinen tunniste. Paikallinen tunniste seuraa täydellisen binääripuun nimeämistä. Näin segmentin SS_i jonkun CEI:n ainutkertainen globaali tunniste on l+2iL, jossa l on paikallinen tunniste.

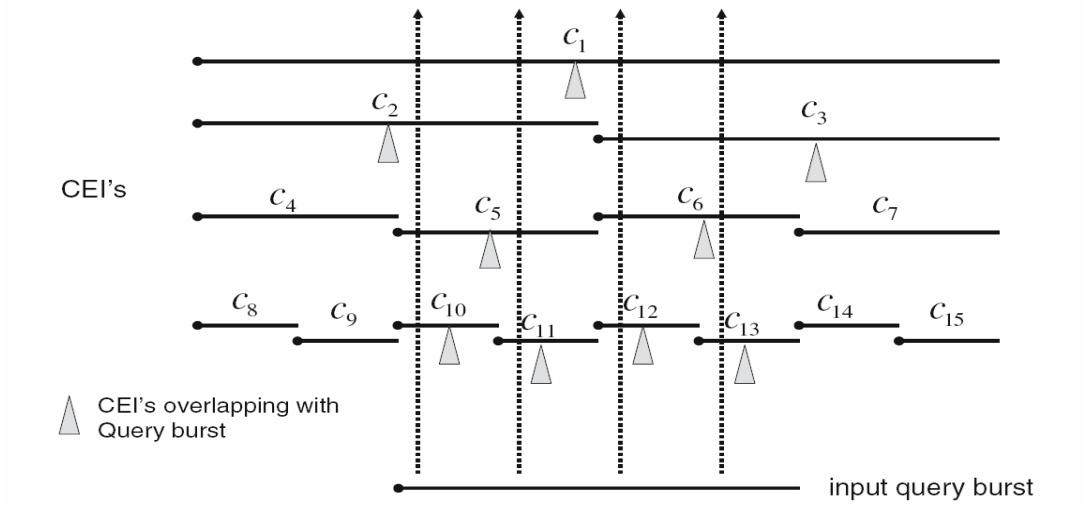
Purskeaikaväli jaetaan tallentamista varten ensin yhteen tai useampaan CEI:hin, sitten sen tunniste lisätään tunnistelistalle joka on liitetty jaettuihin CEI:hin. CEI hakemisto ylläpitää pursketunnistelistojen joukkoa, yhtä kullekin CEI:lle. Kuvassa 6 on tästä yksi esimerkki. Siinä on neljä tietyn segmentin (CEI:t $c_1,...,c_7$) purskeaikaväliä b_1, b_2, b_3 ja b_4 jaettu osiin. b_1 peittää täysin koko segmentin ja sen tunnus liitetään c_1:een. b_2 on segmentin sisällä ja jaetaan osiin c_5 ja c_6. b_3 on myös segmentin sisällä ja se jaetaan osiin c_5 ja c_3. Vastaavasti b_4 liitetään c_4:een.



Kuva 6: Esimerkki CEI-limitysindeksoinnista [Vla07 s.117]

4.2 Limittyvien purskealueiden tunnistaminen

Limittyvien purskealueiden tunnistamista varten on ensin etsittävä limittyvät CEI:t. Yhdessä yksinkertaisessa algoritmissa jaetaan tarkasteltava aikaväli yhden yksikön kokoisiin CEI:hin ja suoritetaan pistehaku kullekin CEI:lle. Replikaattien eliminointi kuitenkin tarvitaan redundanttisten limittyvien CEI:tten poistamiseen. Kuvassa 7 on tästä esimerkki. Siinä on 9 ainutkertaista limittyvää CEI:tä, ja 16 limittyvää CEI:tä (4 kullakin ylöspäin suuntautuvalla nuolella), joista 7 on poistettavia replikaatteja (3 c_1:lle ja 1 c_2:lle, c_3:lle, c_4:lle, c_5:lle ja c_6:lle kullekin). Redundanttisten CEI:tten poistaminen hidastaa hakualgoritmia. Tässä tutkielman lähteen tekijät ovat kehittäneet hakualgoritmin, jossa ei tarvita replikaattien eliminointia. Kuvan 8 mukaisella pseudokoodilla voidaan systemaattisesti tunnistaa kaikki limittyvät purskeet syöttöalueella [x,y), jossa x ja y ovat kokonaislukuja, x>y ja [x,y) sijaitsee kahden peräkkäisen ohjauspaikan (segmenttirajan) välissä.



Kuva 7: Esimerkki siitä, miten CEI-limitys löydetään annetulta aikaväliltä [Vla07 s.118]

```
Search ([x, y)) { // [x, y) resides between two consecutive guiding posts
  i = ⌊x/L⌋; // segment ID
  l_1 = x − iL + L; // leftmost unit-sized CEI overlapping with [x, y)
  l_2 = (y − 1) − iL + L; // rightmost unit-sized CEI overlapping with [x, y)
  for (j = 0; j ≤ k; j = j + 1) {
    for (l = l_1; l ≤ l_2; l = l + 1) {
      c = 2iL + l; // global ID of an overlap CEI
      if (IDList[c] ≠ NULL) { output(IDList[c]); }
    }
    l_1 = l_1/2; // local ID of parent of l_1
    l_2 = l_2/2; // local ID of parent of l_2
  }
}
```

Kuva 8: Pseudokoodi, jolla etsitään limittyvät purskeet [Vla07 s.118]

Ensin lasketaan segmenttitunniste i=[x/l] ja sitten paikalliset tunnisteet eniten vasemmalla olevalle yhden yksikön mittaiselle CEI:lle, l_1=x-iL+L ja eniten oikealla olevalle yhden yksikön mittaiselle CEI:lle, l_2=(y-1)-iL+L, jotka limittyvät [x,y):n kanssa. Lasketuista l_1 ja l_2 voidaan systemaattisesti paikantaa kaikki CEI:t, jotka limittyvät annetun aikavälin kanssa. Jokainen CEI, jonka paikallinen tunniste on l_1:n ja l_2:n välissä myös limittyy annetun aikavälin kanssa. Sitten siirrytään yksi taso ylöspäin l_1:n ja l_2:n vanhempiin. Prosessia toistetaan, kunnes l_1=l_2=c_1. Kukin limittyvä CEI tutkitaan vain kerran, joten replikaattien eliminointia ei tarvita.

Tarkastellaan seuraavaksi tapauksia, joissa annettu aikaväli ei sijaitse kahden peräkkäisen segmenttirajan sisällä. Vastaavasti kuin purskeiden jakoprosessissa, annettu aikaväli voidaan jakaa segmenttirajojen mukaisesti osiin, joille kullekin sovelletaan kuvan 8 hakualgoritmia.

Hakutuloksissa voi olla duplikaatti-pursketunnisteita, koska purske voi tulla jaetuksi yhteen tai useampaan CEI:hin ja useampi niistä voi limittyä annetun aikavälin kanssa. Näiden duplikaattien tehokasta eliminointia varten purskeiden tunnistelistoja ylläpidetään siten että tunnukset lajitellaan yksittäisten tunnistelistojen sisällä. Haun aikana sen sijaan että raportoitaisiin jonkin CEI:n kaikki limittyvät pursketunnisteet yksi CEI kerrallaan, paikallistetaan ensin limittyvät CEI:t. Sitten näihin CEI:hin liittyvät useat tunnistelistat yhdistetään hakutulokseen, ja tässä yhdistämisessä voidaan duplikaatit eliminoida tehokkaasti.

4.3 Hakemiston inkrementaalinen ylläpito

Koska aika etenee pysähtymättä, valittiinpa alkuperäinen [0,r) miten tahansa, yläraja r ohitetaan joskus väistämättä. Tarpeeksi suuren arvon valitseminen r:lle ei ole hyvä ratkaisu, koska hakemiston koko kasvaa vastaavasti tarpeettomasti. Parempi lähestymistapa on valita r suuremmaksi kuin käytössä oleva suurin purskealueikkuna ja pitää kaksi hakemistoa muistissa kaksoispuskuroinnin tapaan. Aloitetaan [0,r):llä. Kun aika ohittaa r:n, luomme toisen

hakemiston [r,2r). Kun aika ohittaa 2r:n, luomme hakemiston [2r,3r), mutta hakemistoa [0,r) ei enää tarvita, joten sen vaatima muistitila voidaan vapauttaa. Tällä lähestymistavalla ei tapahdu virheellisiä hylkäämisiä, koska mikä tahansa kahteen alueeseen kuuluva purskeaikaväli voidaan jakaa aluerajojen mukaisesti ja indeksoida tai etsiä vastaavasti.

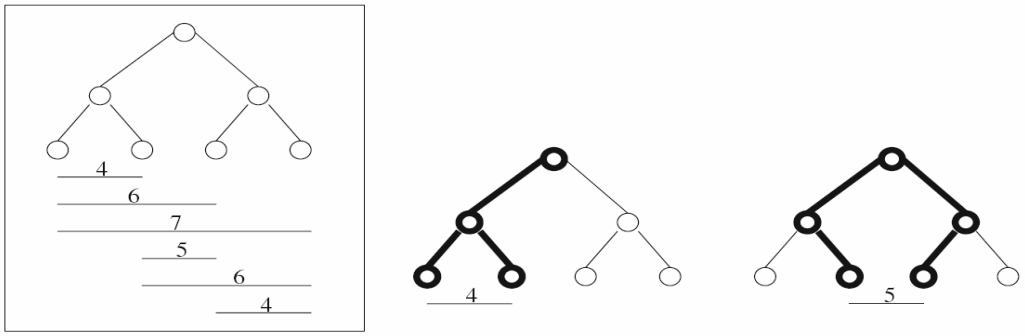
4.4 Pohdintaa ja rajoituksia

Ensiksi, CEI-limityshakemisto on suunniteltu nopeita lisäämis- ja hakutoimintoja varten, ei nopeita poistamisia varten. Purskeaikavälin poistamiseksi se täytyy ensin jakaa joukoksi CEI:tä. Kullekin näistä täytyy sitten suorittaa sekventiaalinen liitettyjen tunnistelistojen läpikäynti pursketunnisteen poistamiseksi, mikä on keskimäärin tehottomampaa kuin lisääminen. Tässä käsiteltävässä purskekorrelaatiosovelluksessa ei tarvita poistoja, joten suorituskyky ei siten kärsi. Toiseksi, CEI-limityshakemiston muistitilavaatimus voi olla suuri, jos r on suuri, erityisesti tapauksessa, jos pitää tallettaa suuri määrä purskeaikavälejä. Yksikertainen mutta tehokas ratkaisu tähän tapaukseen olisi osittaa purskeaikavälit ja rakentaa erillinen CEI-limityshakemisto kullekin osalle.

4.5 Tunnistusalgoritmin laskennallinen vaativuus

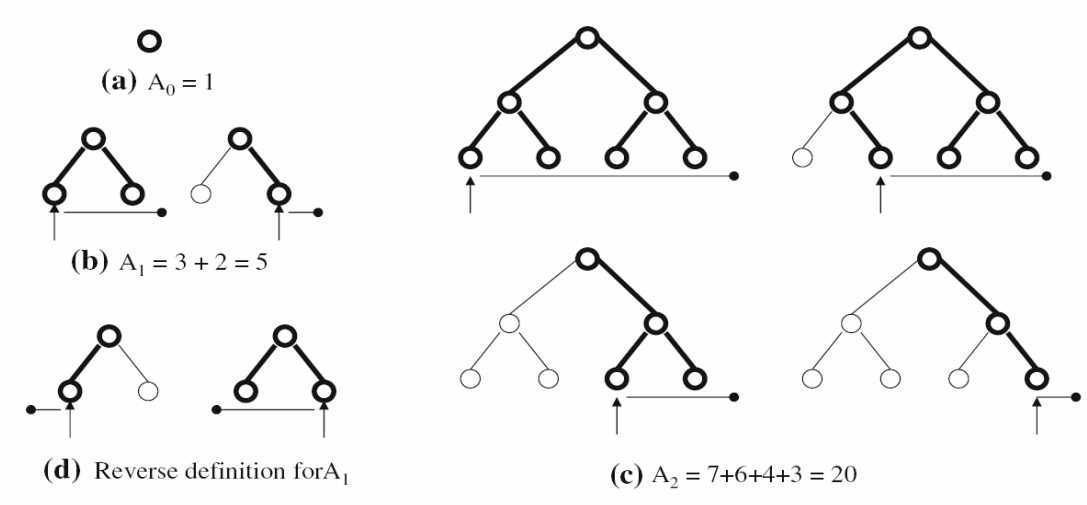
Lähteessä analysoidaan sitten kuvassa 8 kuvatun limityshakualgoritmin vaativuutta. Näytetään, että sen vaativuus on keskimäärin O(L), vakiotekijällä 2/3 ja pahimman tapauksen vaativuus on O(2L-1), kun segmentin kaikki CEI:t pitää tutkia. Aiemmin käytetyn algoritmin tehokkuus on O(L log(L)). Varsin perusteellinen analyysi on lähteessä sivuilla 120...124; tilanpuutteen takia en referoi tätä kohtaa tämän enempää tässä tutkielmassa. Sisällytän tähän kuitenkin lähteessä olleet kuvat 9...13.

(a) $B_1 = 3$

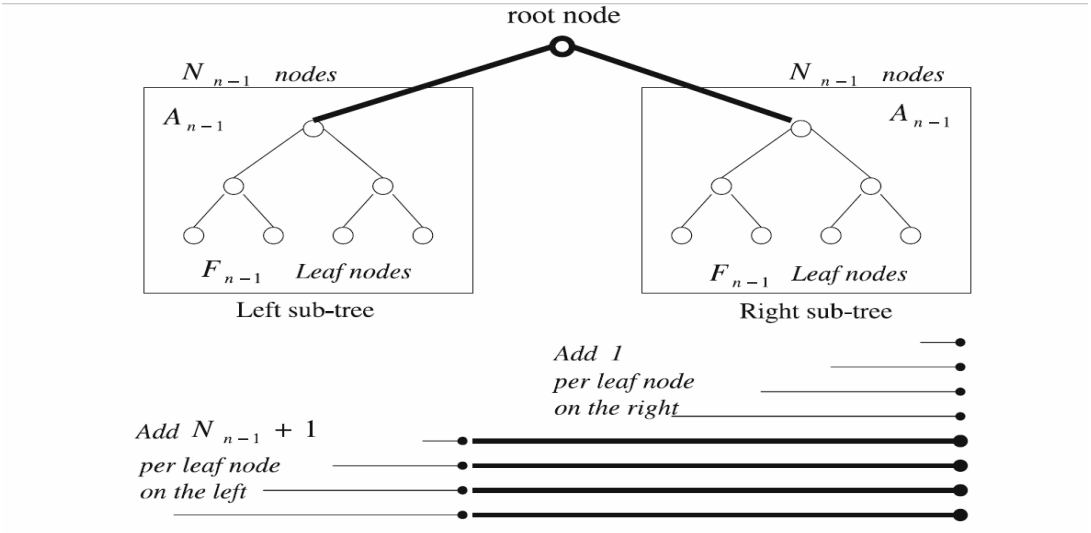


(b) $B_2 = 4+6+7+5+6+4=32$

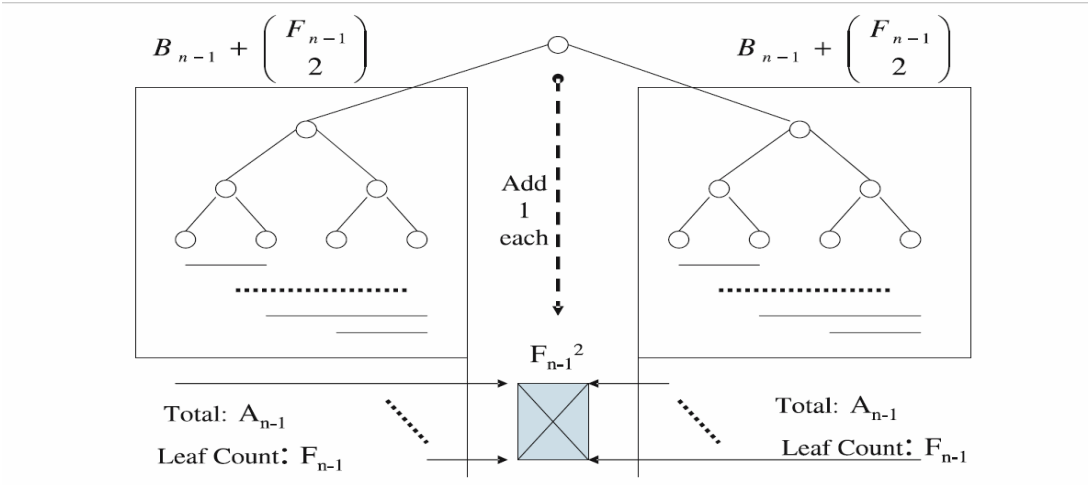
Kuva 9: Esimerkki: B_1 ja B_2 [Vla07 s.121]



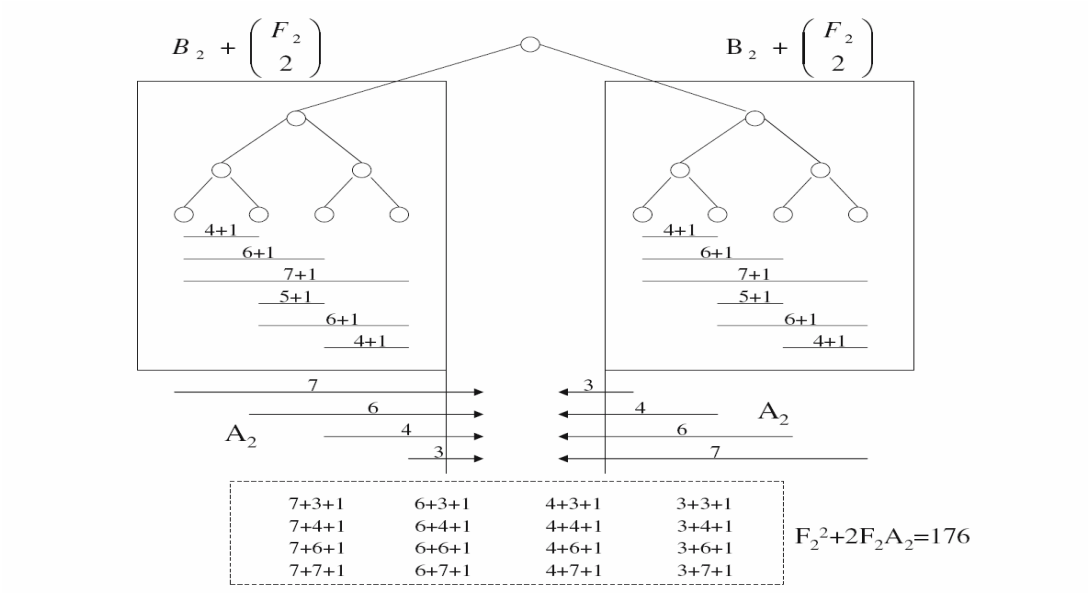
Kuva 10: Esimerkki: A_0, A_1 ja A_2 [Vla07 s.121]



Kuva 11: Toistumis-relaatio A_n:lle [Vla07 s.122]



Kuva 12: Toistumis-relaatio B_n:lle [Vla07 s.123]

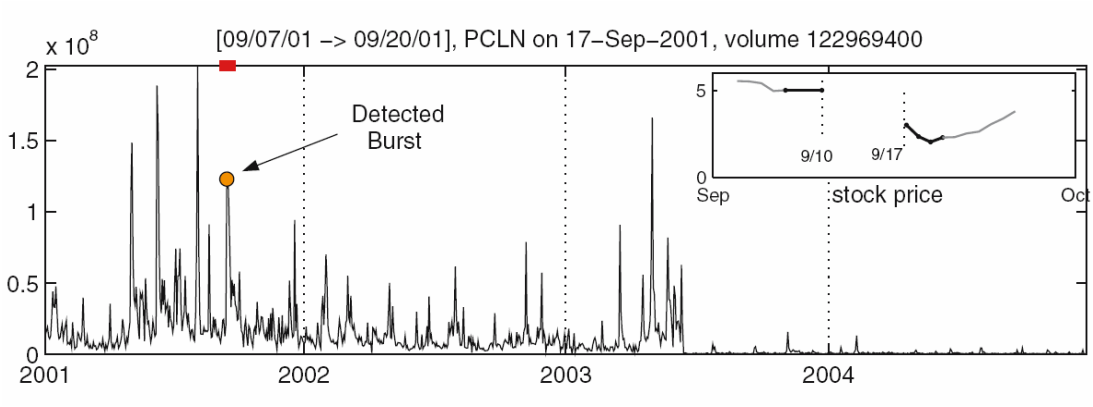


Kuva 13: Toistumis-relaatioesimerkki B_3:lle[Vla07 s.124]

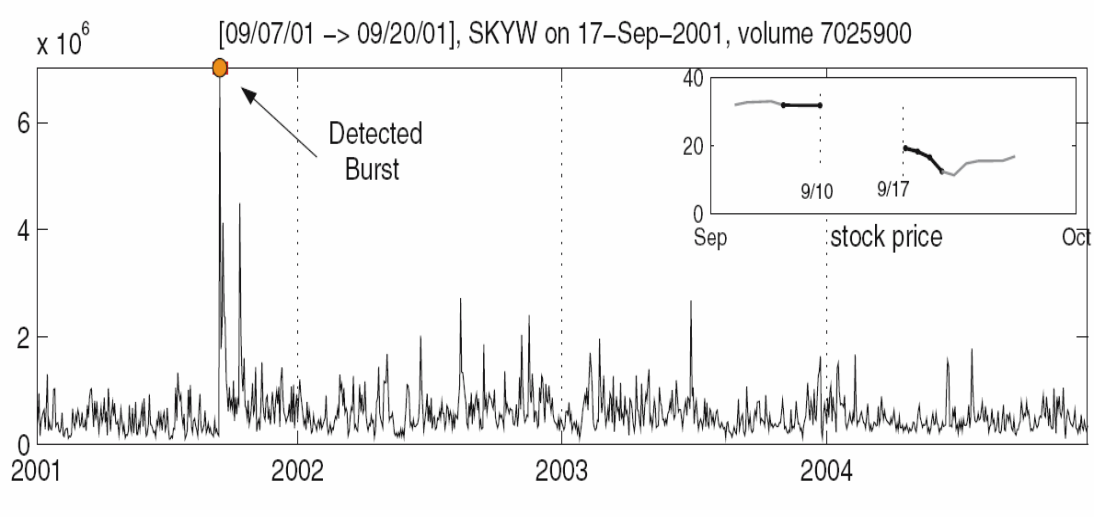
5 Testausta

Arvioidaan kolmea purskekorrelaatioskeeman parametria: (i) tulosten laatu (onko purskekorrelaatio hyödyllinen?), (ii) hakemiston vasteaika (kuinka nopeasti tulokset saadaan?) ja (iii) indeksointimenetelmien vertailua (kuinka paljon parempi se on kuin muut lähestymistavat?).

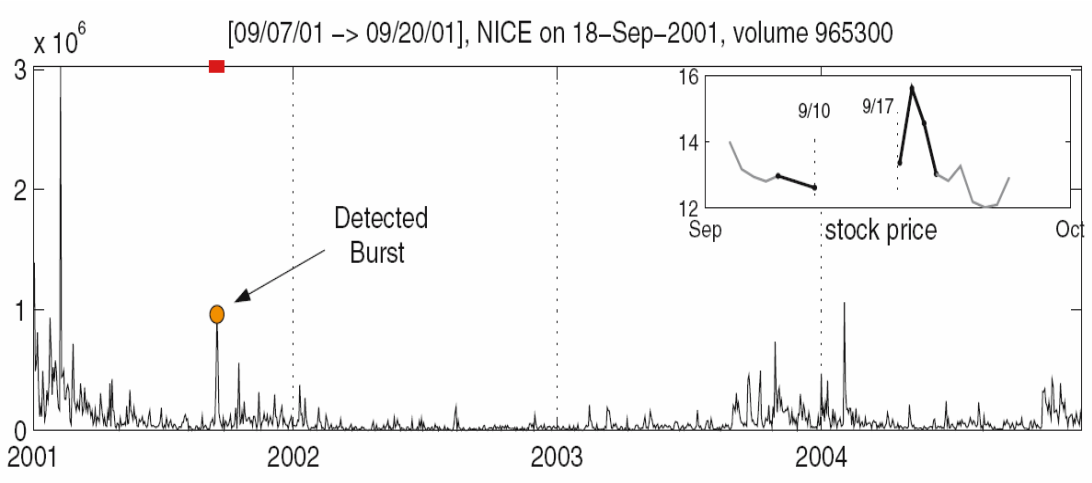
Aluksi arvostellaan purskekorrelaatiomenetelmällä saavutettujen tulosten laatua. Etsitään osakevaihdon purskekuvioita ennen syyskuun 11 päivän hyökkäystä ja sen jälkeen tarkoituksen tutkia hypoteesia, että tapahtumat olisivat vaikuttaneet finanssialan tai matkailualan yhtiöihin. Hyödynnetään finance.yahoo.comilta löytyvää historiatietoa, jossa on mukana 4793 osaketta, datan pituus 1000 ja peittää aikavälin vuodesta 2001 vuoteen 2004. Käytetään kunkin osakkeen vaihdon volyymiä purskeen tunnistusalgoritmin syötteenä. Purskekyselyalueena käytettiin päiviä 7.9.2001... 20.9.2001; on muistettava, että kaupankäynti oli suljettuna päivät 11–16 heinäkuuta 2001. Kuvissa 14...17 on esimerkkejä useista osakkeista, joihin tapahtumat vaikuttivat. Kuvissa on esitetty ao. osakkeen kysyntä; osakkeen markkinahinta on myös mukana oikeassa yläkulmassa koko syyskuun ajalle. Kyselyalueella oleva osuus on merkitty tummalla viivalla. Pricelina ja Skywest ovat matkailualan kohteita, ja ne ilmentävät voimakasta vaihtovolyymin määrän kasvua, mikä johtaa hinnan laskuun markkinoiden jälleen avautuessa. Vastaavasti lennonvalvonnan laiteita toimittavan NICE Systemsin ja puolustuselektroniikkatehdas Mercury Computer Systemsin osakkeet esittävät hinnan nousun (kuvat 16 ja 17). Lisää esimerkkejä on liitteessä taulukossa ja kuvissa. Yleensä matkailualan, ilmailunliikenteen, pankkialan ja lääkealan osakkeet laskevat. Toisaalta puolustusalan ja elokuva-alan osakkeet nousevat, mihin liittyy kysynnän äkillinen kasvu.



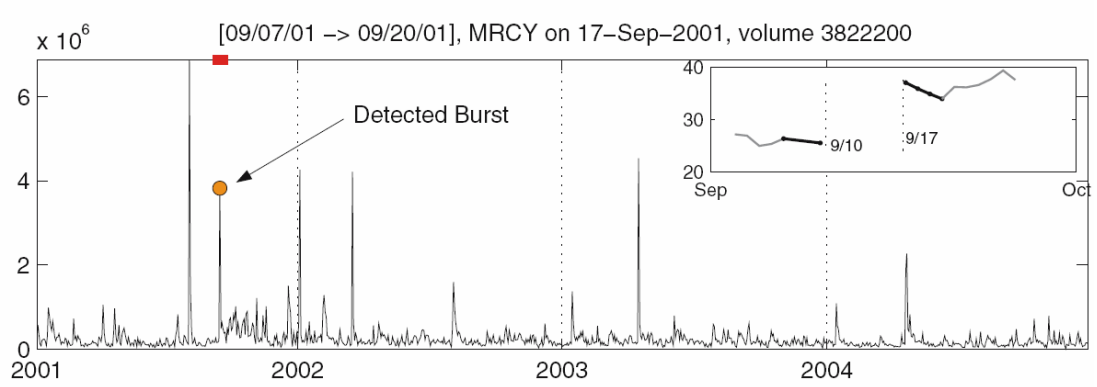
Kuva 14: Pricelinen osakkeen vaihtovolyymi. Havaitaan voimakas myyntipyrkimys, mistä seuraa osakkeen hinnan pudotus [Vla07 s.125]



Kuva 15: Skywestin osakkeen vaihdon volyymi [Vla07 s.125]

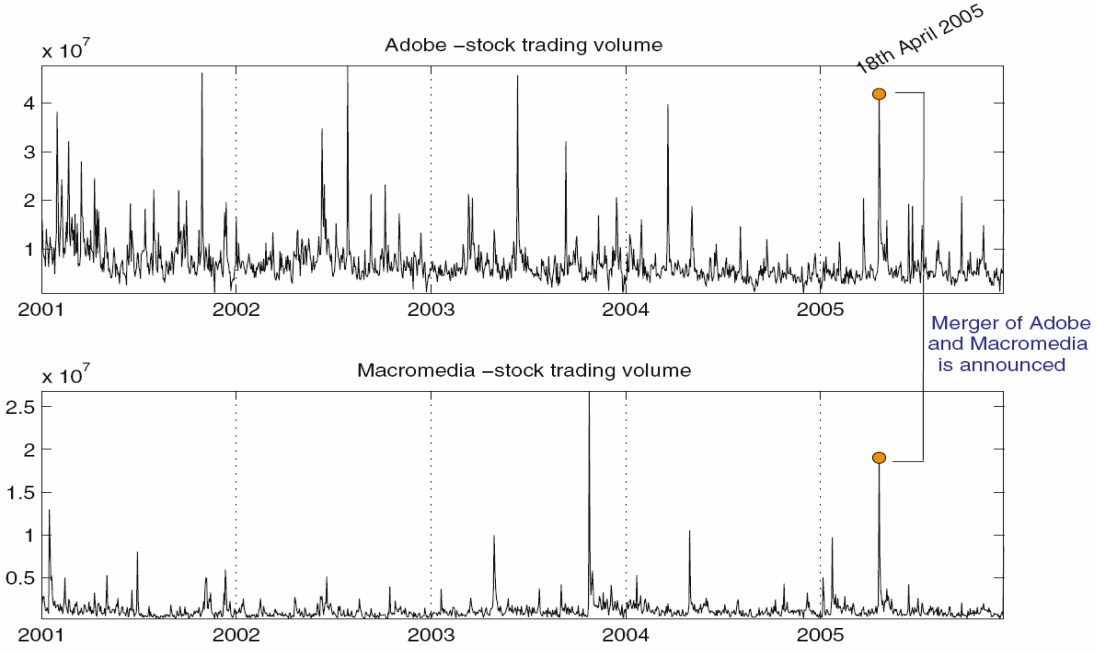


Kuva 16: Nice Systemsin (lennonvalvontajärjestelmiä toimittava yritys) osakkeen vaihdon volyymi. Tässä tapauksessa osakkeen korkea kysyntä aiheuttaa osakkeen hinnan nousua [Vla07 s.126]



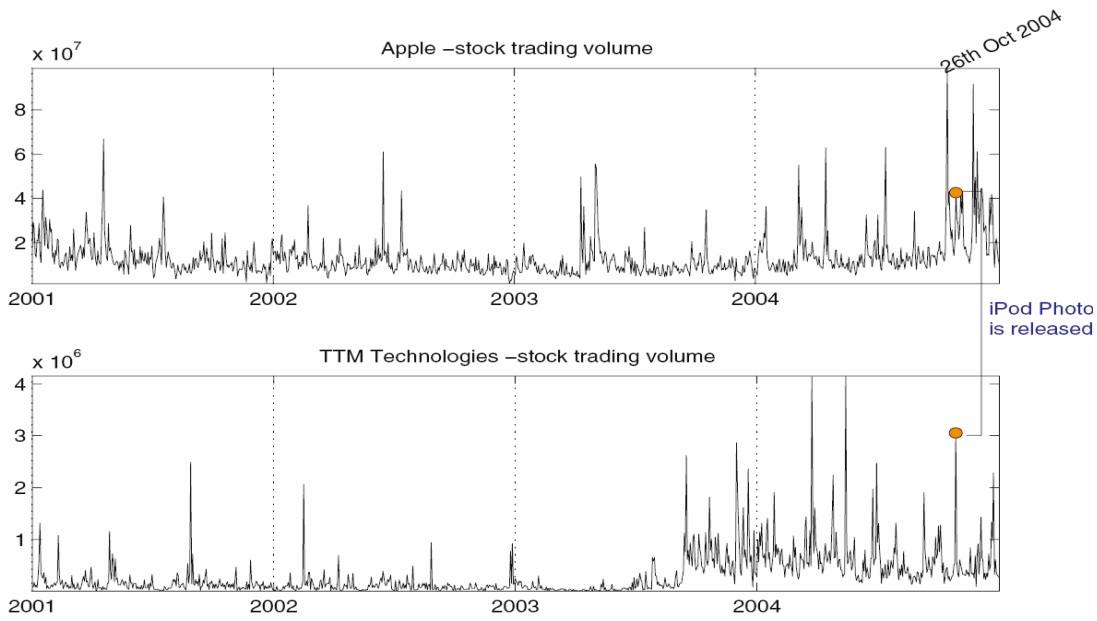
Kuva 17: Mercury Computer Systems in (puolustuselektroniikan suunnittelu- ja valmistusyritys) osakkeen vaihdon volyymi. Osakkeen hinta nousi merkittävästi 17.9.2001 [Vla07 s.126]

Lisäesimerkkeinä tarkastellaan erilaisilla kronologisilla alueilla, kuinka voimakas työkalu purskekorrelaatio voi olla, erityisesti yhteyksien ja vuorovaikutusten päättelyyn yhtiöiden ja tapahtumien välillä. Ensimmäinen esimerkki on vahvasti korreloivat purskeet 18.4.2005 Adoben ja Macromedian vaihtomäärissä. Molempien yhtiöiden osakkeilla on suuri ostokysyntä. Uutishistoriaa seuraamalla havaitaan, että tuolloin tapahtui näiden kahden yhtiön fuusio. Kuva 18 esittää tätä korrelaatiota.



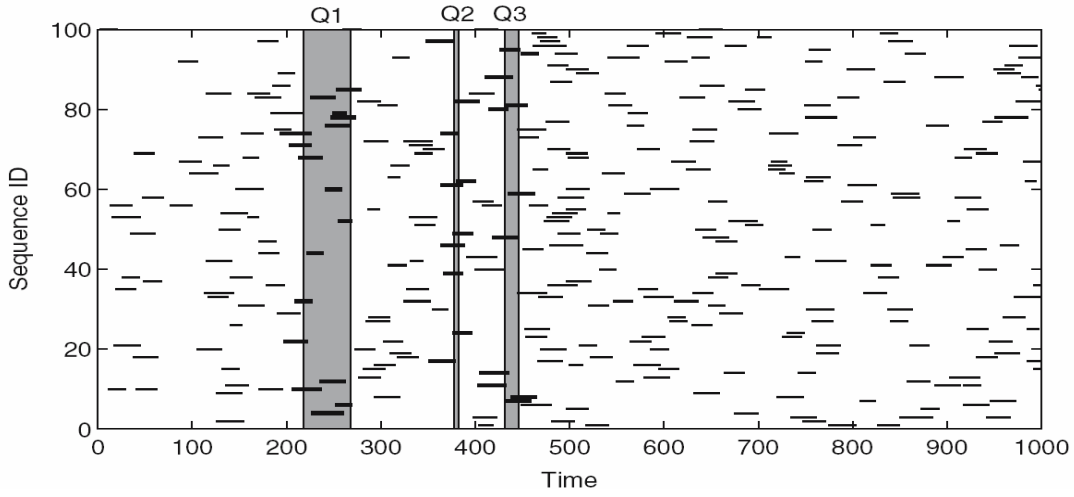
Kuva 18: Adoben ja Macromedian osakkeiden vaihtomäärien välillä havaittu voimakas korrelaatio 18.4.2005 jolloin niiden fuusiosta ilmoitettiin [Vla07 s.127]

Toinen esimerkki on hienostuneempi. Yritämme selvittää, mihin osakkeisiin vaikutti Applen IPod Photon julkistus 26.10.2004. Suurta kysyntää oli Applen osakkeen lisäksi myös piirilevyjä toimittavan TTM Tecnologiesin osakkeilla. Aluksi ei huomata mitään yhteyttä yhtiöiden välillä, mutta sitten selviää, että Apple on TTM Tecnologiesin asiakas.



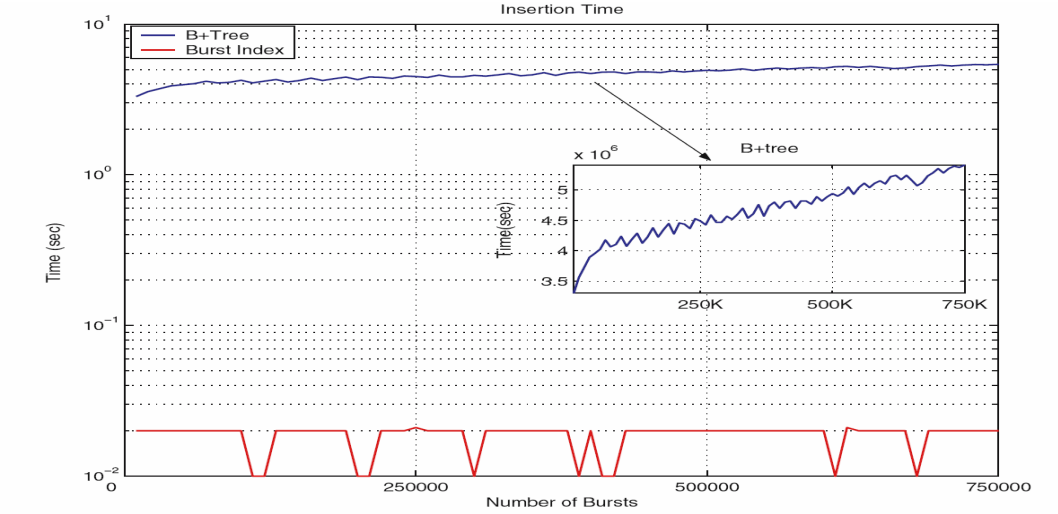
Kuva 19: Applen ja TTM Technogiesin korreloiva purske viittaa vahvaan yhteyteen iPod Photon markkinoille tuonnin aikana [Vla07 s.128]

Verrataan uuden CEI-limitys indeksointijärjestelmää B+puuhun. Tässä esimerkissä purskeindeksiä käytetään MSN-hakukoneen hakusanojen vastaavaan esiintymisen tunnistamiseen. Molemmista menetelmistä käytetään muistissa toimivaa versiota. Vertaillussa voitaisiin käyttää myös Hansonin ja Johnsonin 1996 esittää Interval Skips Lists-menetelmää tai Guttmanin R-puu-menetelmää. On kuitenkin näytetty, että CEI on näitä parempi, joten näihin ei nyt tehdä vertailua. Suorituskykyä mitataan hakemiston lisäysajalla ja hakemiston vasteajalla. Koska osaketietokanta on liian pieni, generoidaan kolme erikokoista keinotekoista pursketietokantaa (250K, 500K ja 750K tietuetta) satunnaislukugeneraattorilla. Tietokannasta on näyte kuvassa 20.



Kuva 20: Keinotekoinen datajoukko ja esimerkki kolmesta purskealuekyselystä [Vla07 s.130]

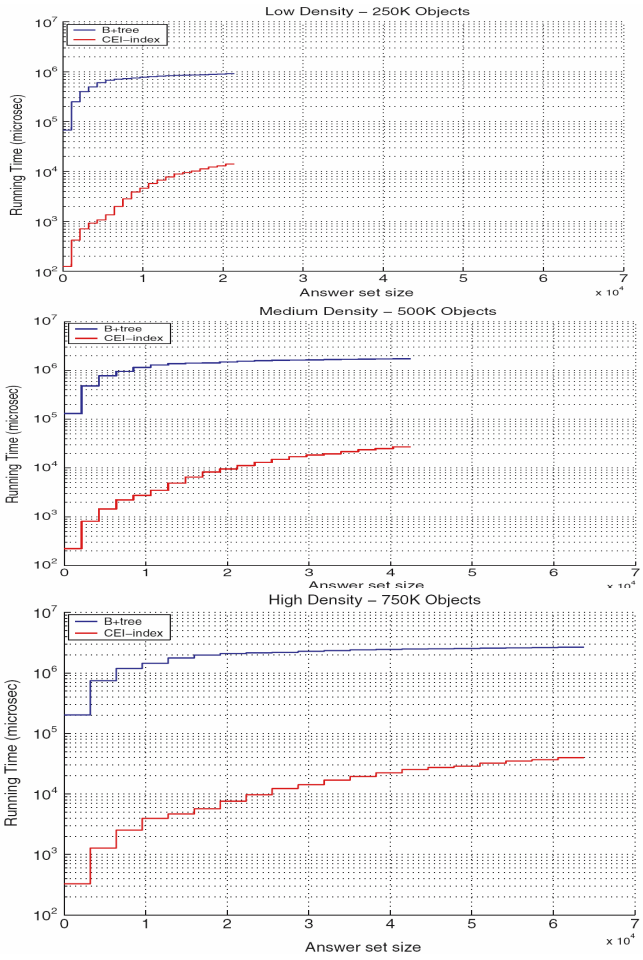
Ensin mitataan purskealuehakemiston muodostamiseen tarvittava aika molemmille menetelmille 750000 purskealueella (kuva 21).



Kuva 21: Indeksien laskenta-aika. B+puun lisäysaika riippuu lineaarisesti objektien lukumäärästä, kun taas CEI-indeksillä on vakio lisäysaika [Vla07 s.128]

B+puulla lisäysaika on riippuu melko lineaarisesti purskeiden määrästä, mikä onkin odotettavissa, koska tarvitaan puun tasapainottava vaihe, jonka kesto kasvaa tietokannan koon kasvaessa. CEI-pohjaisella hakemistolla lisäysaika on vakio tietokannan koosta riippumaton vakio, mikä selittyy suurimmaksi osaksi nopeasta hajautustyyppisestä sijoitusmekanismista. Tasapainotusta ei tarvita. CEI on noin kolme kertaluokkaa nopeampi kuin B+puuhun perustuva lähestymistapa.

Lisäysaikaa kriittisempi suorituskykymittari on erilaisten kyselyjen vaatima aika, eli toisin sanoen, kuinka paljon aikaa tietty määrä aikavälien limittymiskyselyjä vaatii. Tätä kokeiltiin sekä CEI-limitykselle että B+puulle 5000 erilaisia paikkoja ja



Kuva 22: B+puun ja CEI-limityksen suoritusajan vertailu [Vla07 s.123]

alueita kattavalla kyselyalueella. Intuitiivisesti kyselyn vaatima aika on verrannollinen niiden aikavälien määrään, jotka limittyvät annetun kyselyn kanssa. Siksi on tarpeen esittää kyselyjen suoritusaika vastausjoukon koon funktiona. Kuvassa 22 on esitetty vastaavat histogrammit. Näistä nähdään CEI-pohjaisen hakemiston ylivoimaisuus; menetelmä on 2-3 kertaluokkaa nopeampi kuin kilpaileva B+puumenetelmä. Koska suoritusaika on mikrosekuntialueella, on ehdotetun hakemistojärjestelmän suorituskyky reaaliaikasovellusten vaatimalla tasolla.

6 Johtopäätökset

Tässä tutkielmassa on esitelty kattava kehys purskeiden korrelointiin. Järjestelmän tehokkuus ei perustu vain tehokkaaseen purskeiden havaitsemiseen vaan myös tehokkaaseen muistipohjaiseen hakemistoon. Hakemisto järjestää hierarkkisesti aikasarjojen tärkeät purskepiirteet purskesegmenttien muodossa ja sen jälkeen mahdollistaa hyvin tehokkaan havaittujen purskeiden limityslaskennan. Menetelmän tuottama erittäin huomattavasti parantunut vasteaika on näytetty simuloimalla ja osakekaupan tietokannoista louhitusta tiedosta on esitetty mielenkiintoisia purskekorrelaatioita. Indeksointimenetelmää on tarkoitus tulevaisuudessa soveltaa havaittujen purskeiden klusterointiin ja tietovirtojen ristikorrelaatioiden havaitsemiseen niiden purskeominaisuuksien avulla.

Lähteet

Vla07 Vlachos,M., Wu,K., Chen,S. ja Yu,P. *Correlating burst events on streaming stock market data.* Data Mining and Knowledge Discovery, March 2007, sivut 109-133.

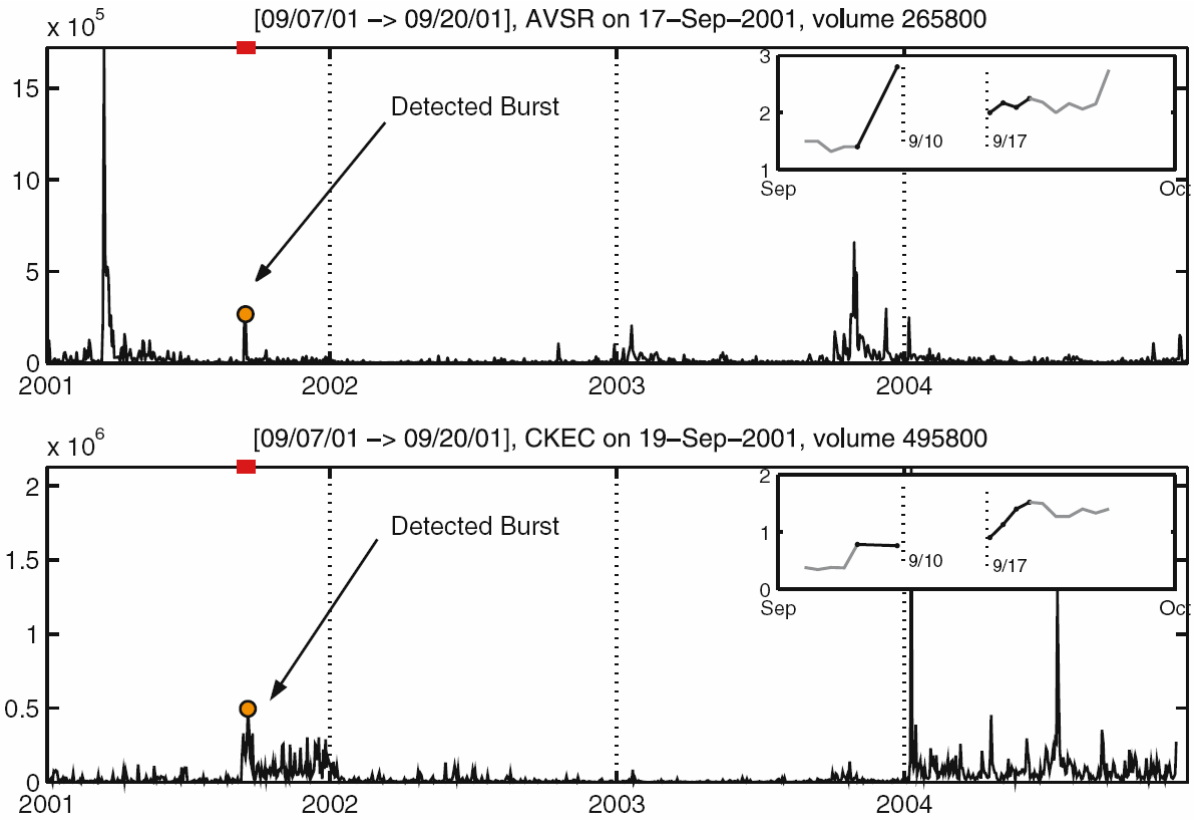
Liitteet

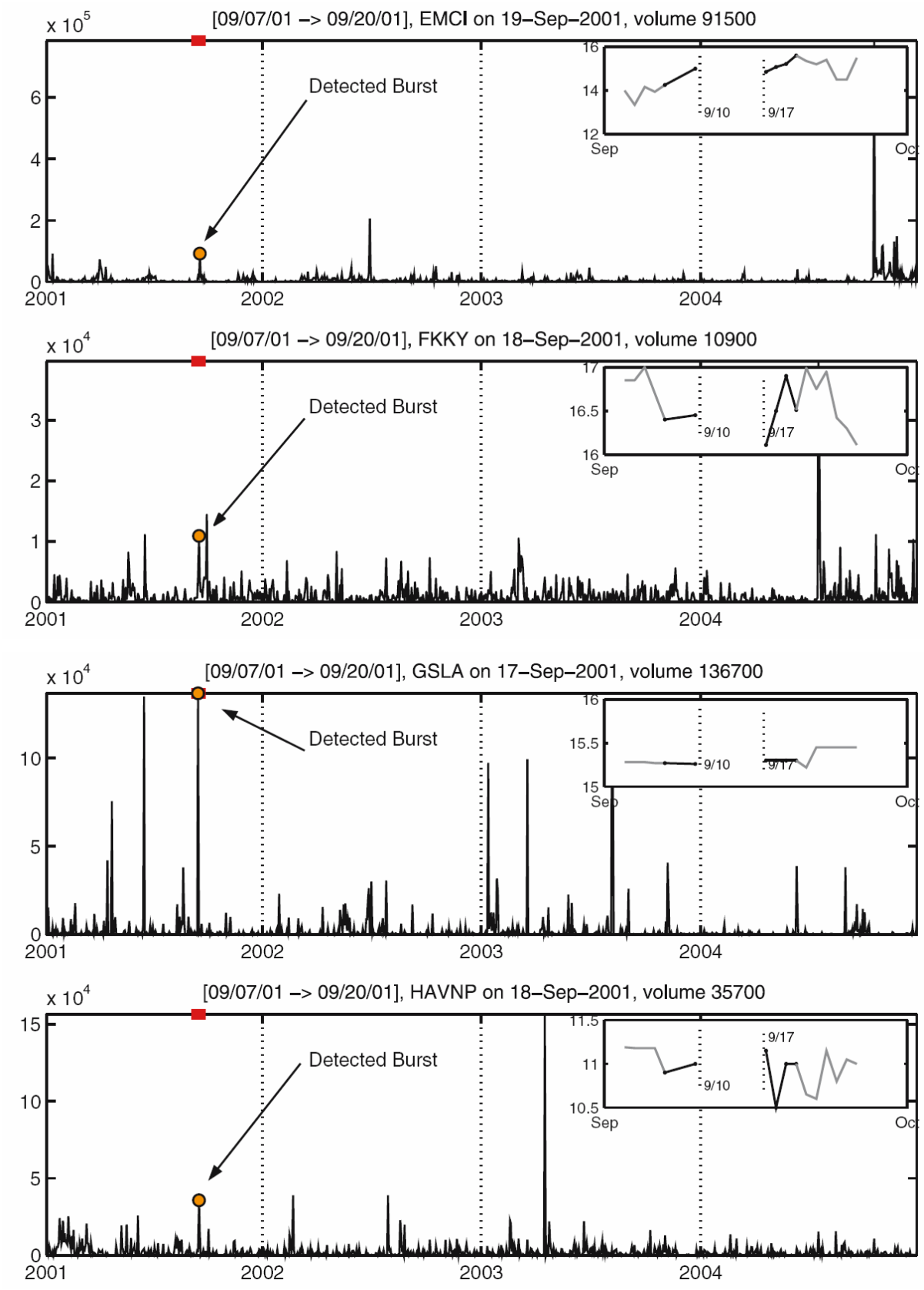
Liite 1: Osakkeita, joilla käytiin vilkkaasti kauppaa 9.11.2001 tapahtumien jälkeen [Vla07 s.129]

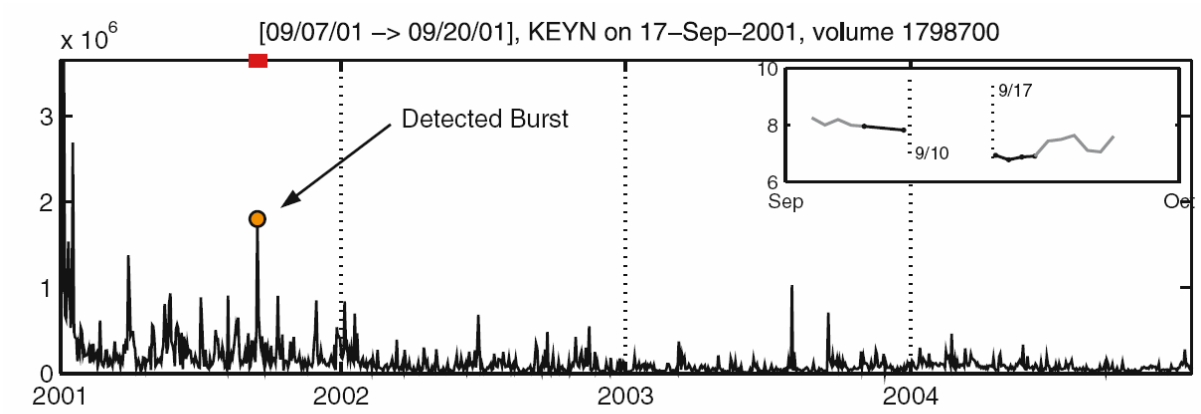
Table 2 Some of the stocks that exhibited high trading volume after the events of 9/11/2001

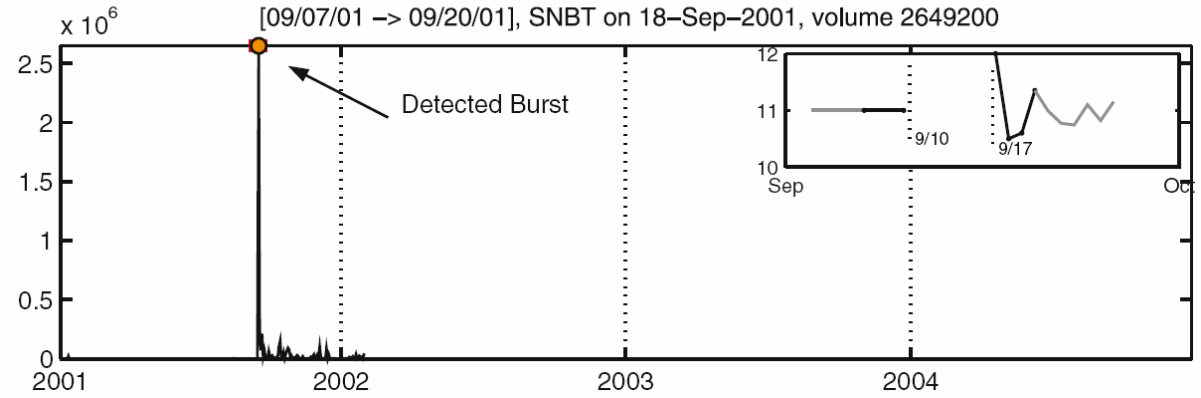
Symbol	Name (Description)	Price
AVSR	Avistar Communications	28% ↓
BEAV	Be Aerospace Inc	65% ↓
CKEC	Carmike cinemas	97% ↑
EMCI	EMC Insurance	4% ↑
ESLT	ELBIT Systems LTD (defense electronics supplier)	11% ↑
FKKY	Frankfort First Bancorp	0.3% ↑
FLYI	Atlantic Coast Airlines Holdings, Inc	35% ↓
HAVNP	Haven Capital Trust	4.5% ↓
INSU	Insituform Technologies (pipe tunneling)	38% ↓
KEYN	Keynote Systems (e-business services)	11% ↓
LIFE	Lifeline Systems (Medical Emergency Response)	1.5% ↓
MAIR	Mair Holdings (Airline Subsidiary)	36% ↓
MRCY	Mercury Computer Systems (defense electronics)	44.8% ↑
NICE	NICE Systems (Air traffic Control Systems)	25% ↑
PCLN	Priceline	60% ↓
PRCS	Praecis Pharmaceuticals	41% ↓
SKYW	Skywest Inc	61 % ↓
STNR	Steiner Leisure (Spa & Fitness Services)	51 % ↓
STNJ	Sterling Bank	2.5% ↓
TSBK	Timberland Bancorp, Inc.	7% ↓

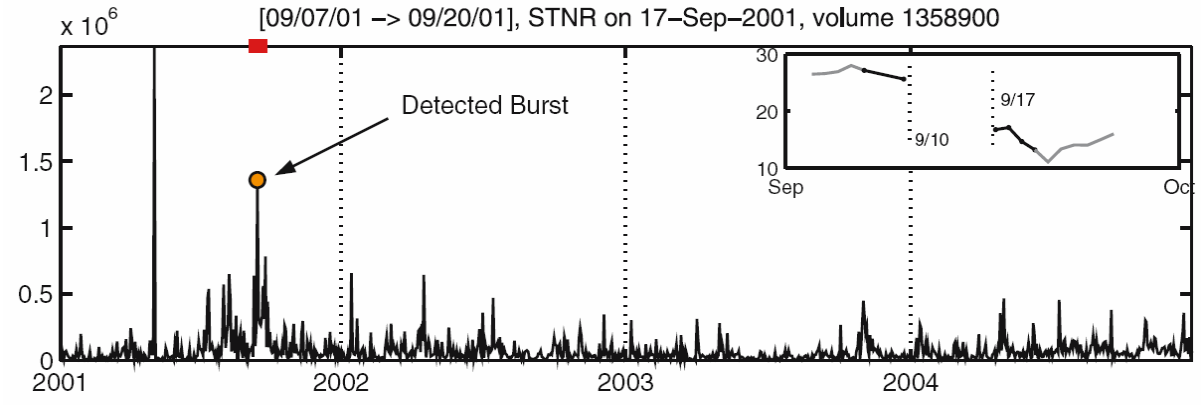
Liite2: Lisää esimerkkejä osakkeista, joihin 9.11.2001 tapahtuma vaikutti [Vla07 s.129]

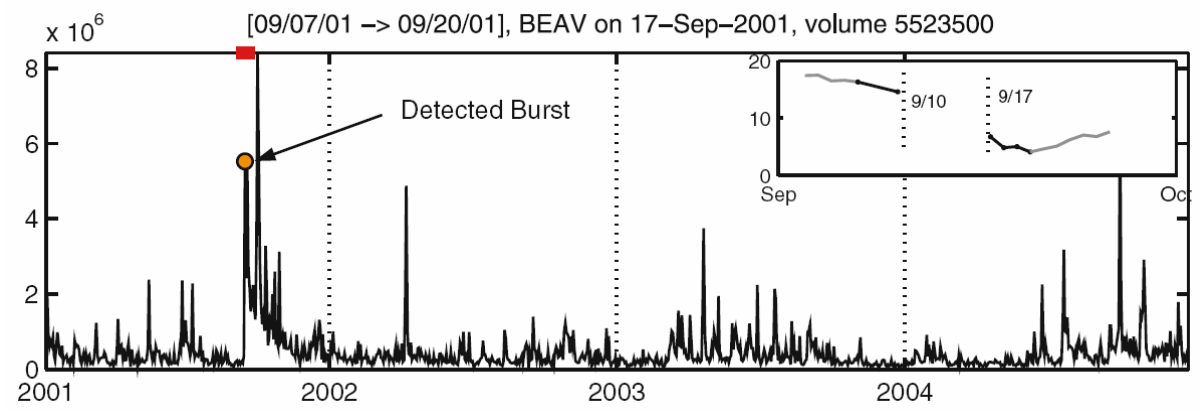


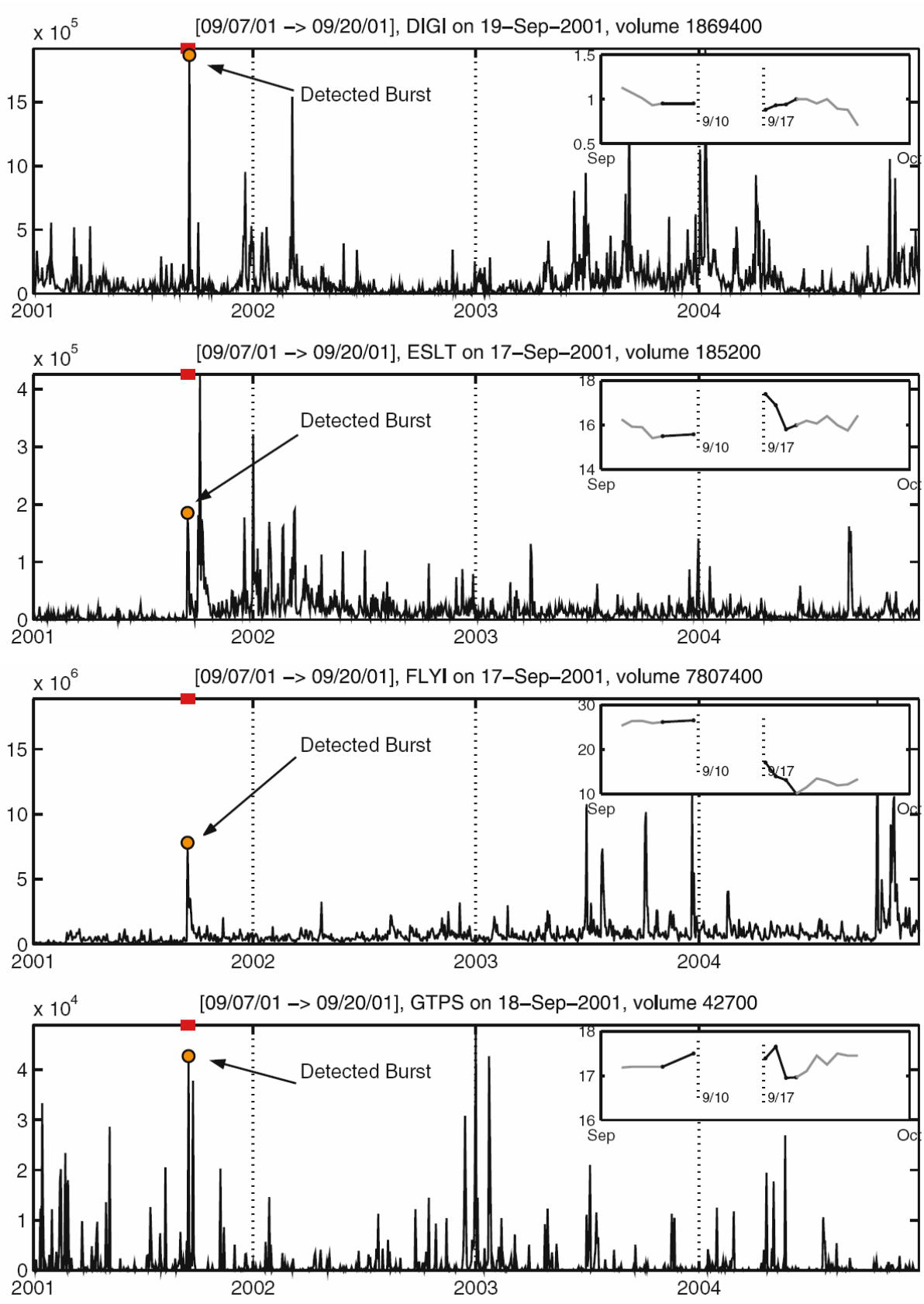

x 10^5
[09/07/01 –> 09/20/01], EMCI on 19–Sep–2001, volume 91500
Detected Burst
9/10
9/17
Sep
Oct
2001
2002
2003
2004
x 10^4
[09/07/01 –> 09/20/01], FKKY on 18–Sep–2001, volume 10900
Detected Burst
9/10
9/17
Sep
Oct
2001
2002
2003
2004
x 10^4
[09/07/01 –> 09/20/01], GSLA on 17–Sep–2001, volume 136700
Detected Burst
9/10
9/17
Sep
Oct
2001
2002
2003
2004
x 10^4
[09/07/01 –> 09/20/01], HAVNP on 18–Sep–2001, volume 35700
Detected Burst
9/10
9/17
Sep
Oct
2001
2002
2003
2004

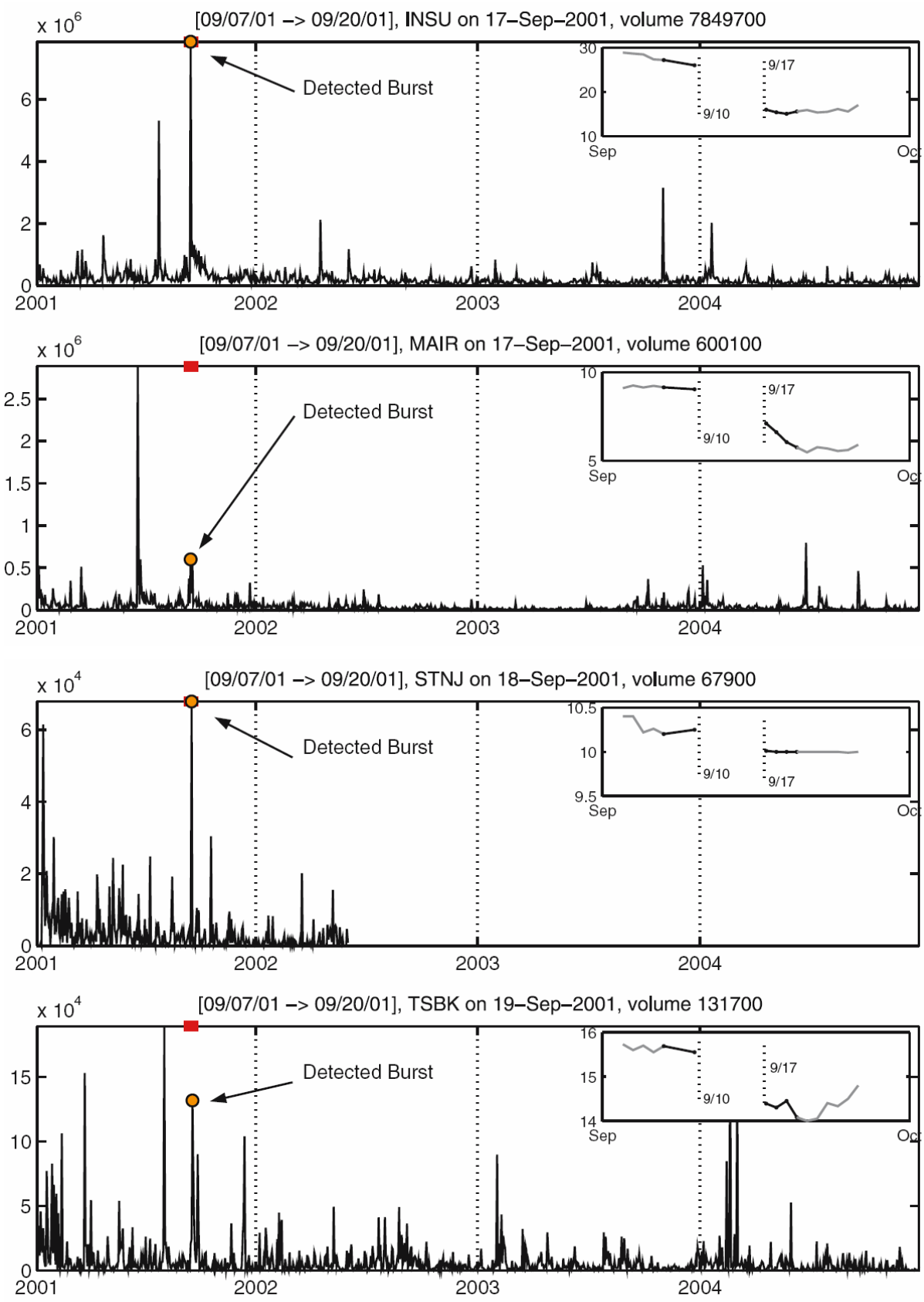

x 10^6
[09/07/01 –> 09/20/01], KEYN on 17–Sep–2001, volume 1798700
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004


x 10^6
[09/07/01 –> 09/20/01], SNBT on 18–Sep–2001, volume 2649200
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004


x 10^6
[09/07/01 –> 09/20/01], STNR on 17–Sep–2001, volume 1358900
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004


x 10^6
[09/07/01 –> 09/20/01], BEAV on 17–Sep–2001, volume 5523500
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004


x 10^5
[09/07/01 –> 09/20/01], DIGI on 19–Sep–2001, volume 1869400
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004
x 10^5
[09/07/01 –> 09/20/01], ESLT on 17–Sep–2001, volume 185200
Detected Burst
x 10^6
[09/07/01 –> 09/20/01], FLYI on 17–Sep–2001, volume 7807400
Detected Burst
x 10^4
[09/07/01 –> 09/20/01], GTPS on 18–Sep–2001, volume 42700
Detected Burst


[09/07/01 –> 09/20/01], INSU on 17–Sep–2001, volume 7849700
x 10^6
Detected Burst
9/10
9/17
Sep
2001
2002
2003
2004
[09/07/01 –> 09/20/01], MAIR on 17–Sep–2001, volume 600100
x 10^6
Detected Burst
9/10
9/17
Sep
[09/07/01 –> 09/20/01], STNJ on 18–Sep–2001, volume 67900
x 10^4
Detected Burst
9/10
9/17
Sep
[09/07/01 –> 09/20/01], TSBK on 19–Sep–2001, volume 131700
x 10^4
Detected Burst
9/10
9/17
Sep