

TiraLabra 2015 – Toteutusdokumentti

Ohjelman yleisrakenne

Ohjelmassa testataan kahden erityyppisen algoritmin suorituskykyä ratkaista ns. knapsack problem. Ongelmassa pyritään maksimoimaan tietyn tilavuuden omaavan säkin arvo lisäämällä sinne tavaroita, joista tiedetään tilavuus ja arvo. Tavaroiden mittasuhteilla ei ole merkitystä.

Ensimmäinen algoritmi on ns. Naiivi-algoritmi, joka käy läpi, jokaisen mahdollisen tavarakombinaation suurimman löytääkseen. Sen oletettu aikavaatimus perustuen kombinatoriikkaan on $O(2^n)$, jossa n = pakettien lukumäärä. Se on käytännössä kuitenkin pienempi, jos säkin koko verrattain pienempi kuin käytettävissä olevien tavaroiden yhteenlaskettu koko. Tämä johtuu siitä että Naiivi algoritmi ei ole ihan niin naiivi, että se laskisi sellaiset haarat loppuun, jotka jo ylittäneet maksimi koon, vaan siirtyy seuraavaan tutkittavaan haaraan. Tämän algoritmin toiminta on luokassa Taytto.

Toinen algoritmi on nimeltään DP-algoritmi (dynamic programming), jonka idea on käydä läpi pakettien lukumäärän n ja säkin vetoisuuden v muodostama matriisin $n \times v$ parhaan ratkaisun kombinaation löytämiseksi. Aikavaatimus oletus on $O(nv)$. Tämän algoritmin toiminta on luokassa TayttoDP.

Taytto -luokat (em.) saavat ilmentymän luodessa Sakin ja Tavarat -arrayn. Sakki -luokka sisältää tiedon Sakin vetoisuudesta, Tavara -luokka tavarat arvosta ja koosta. Täyttö-algoritmit ajetaan kummassakin luokassa metodilla etsiMaksimiArvoJaJono(), joka tallettaa löytyneen maksimiarvon ja yhden sen toteuttavan tavarakombinaation luokkamuuttujiin. Kumpikin algoritmi samalla säkillä ja tavarakokoonpanolla palauttaa siis saman maksimiarvon. Maksimiarvon toteuttava tavarakombinaatio voi olla erilainen algoritmien välillä, jos ratkaisuja on useita. Maksimiarvon ja sen toteuttavan kombinaation etsiminen ei tulosta mitään tuloksia. Tulokset voi hakea tulostusmetodeilla, kun etsintä on ensin toteutettu.

Testi -luokalla taas mitataan näiden kahden täyttömenetelmän ajallista suorituskykyä ja tulostetaan mittaustuloksia näkyville. Testi -ilmentymä luodaan antamalla sille testattava Sakki ja Tavara -kombinaatio. Testi ilmentymä luo testitilanteita varten Taytto -ilmentymiä, joiden suorituskykyä mittaa. Testi saa syötteenä myös testin toistojen lukumäärän, joista lasketaan keskiarvo tulokseksi. Testi luokan idea on luoda testaukselle mahdollisimman stabiili ympäristö, jotta ajallinen mittaus olisi mahdollisimman häiriötöntä ja riippumaton erilaisista tekijöistä. Näitä sovellettuja menetelmiä käydään tarkemmin läpi suorituskykytestausosiossa. Testit ajetaan ajaTesti() -metodilla, joka myös tulostaa testin tulokset. Testi tarkistaa jokaisen mittauksen osalta, että algoritmien laskema maksimiarvo on sama ja katkaisee testin, jos ei ole (tulostaen kyseisen tapauksen tiedot) eli testien ajaminen samalla validoi myös algoritmien toimintaa.

Ohjelmaa ajetaan Main -luokasta SakinTaytto, jonne asetetaan testattavat tapaukset eli säkkien koot ja tavarakombinaatiot (arvo, koko).

Suorituskykytestit

Testeissä idea todeta, että DP on selvästi nopeampi kuin Naiivi. Tarkastella miten DP-toteuttaa $n \cdot v$ -ennustetta ja Naiivi eksponentiaalista (n^2) ennustetta $n:n$ kasvaessa.

Ajoin testejä kolmessa ryhmässä A, B ja C, joissa tavaroiden n oli 30, 20 ja 10. Jokaisessa ryhmässä ajettiin eri sakkikoolla tulokset. Alla testien testitapaukset ryhmittäin.

Testitapaukset

A

koot: $n=30$ 1,2,3,4,5,6,7,3,4,20,4,5,6,7,3,4,5,6,7,20,4,5,6,7,3,4,5,6,7,20

arvot: $n=30$ 7,6,2,1,4,3,4,3,4,20,7,6,5,4,3,2,4,3,4,20,7,6,5,4,3,2,4,3,4,20

Sakkikoot 20, 60, 100, 144, 188

B

koot: $n=20$ 1,2,3,4,5,6,7,3,4,20,4,5,6,7,3,4,5,6,7,20

arvot: $n=20$ 7,6,2,1,4,3,4,3,4,20,7,6,5,4,3,2,4,3,4,20

Sakkikoot 20, 60, 100

C

koot: $n=10$ 1,2,3,4,5,6,7,3,4,20

arvot: $n=10$ 7,6,2,1,4,3,4,3,4,20

Sakkikoot 20, 60

Alla olevassa taulukossa on tulokset jokaiselle ajolle. Ne ovat keskiarvoja 20 saman ajon tuloksista. Taulukossa siis kunkin ajon nimi, syötteet, tulokset naiiville ja odotettu aikavaatimus sekä tulokset DP:lla ja odotetut aikavaatimukset.

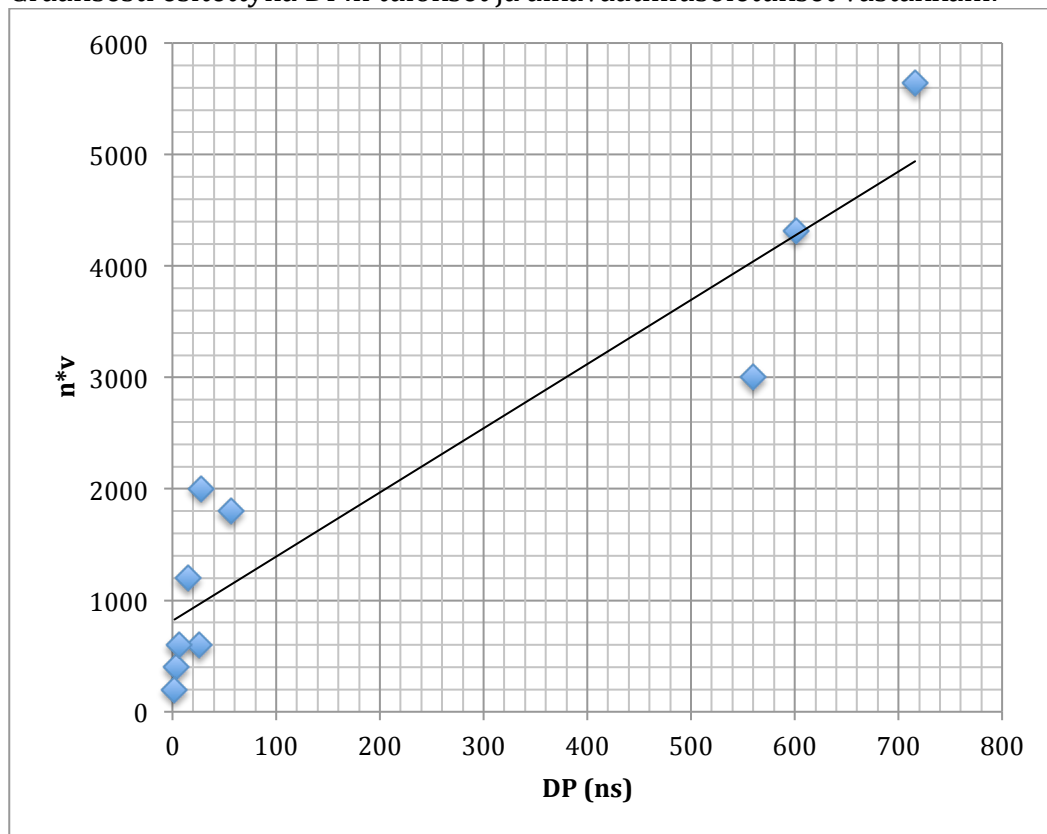
Testin nimi	Syötteet		Naiivin tulokset ja aikavaatimukset		DP:n tulokset ja aikavaatimukset	
	Sakki (v)	Tavarat (n)	Naiivi (ns)	2^n	DP (ns)	$n \cdot v$
A188	188	30	32044198	1073741824	716	5640
A144	144	30	32255846	1073741824	601	4320
A100	100	30	20993099	1073741824	560	3000
A60	60	30	2091893	1073741824	56	1800
A20	20	30	1048	1073741824	26	600
B100	100	20	31553	1048576	28	2000
B60	60	20	15207	1048576	15	1200
B20	20	20	186	1048576	4	400
C60	60	10	39	1024	6	600
C20	20	10	13	1024	2	200

Ensimmäisenä havaintona voidaan sanoa, että DP suorittaa selvästi tehokkaammin täytön. Kun $n=10$, niin ero muutamissa kertaluokissa, mutta kun $n=20$ tai 30, niin ero on useissa kymmenien potensseissa.

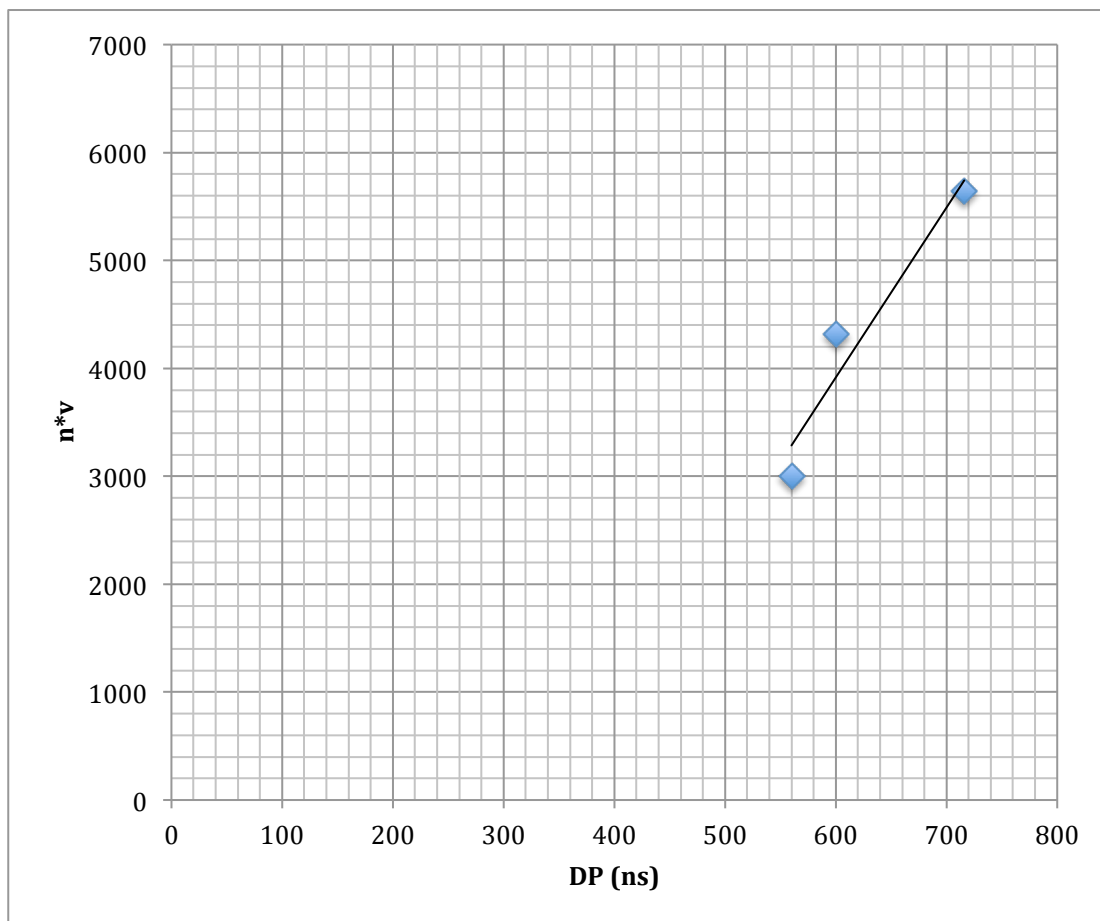
Seuraavassa taulukossa hieman **DP-algoritmin tuloksia** tarkemmin:

Testin nimi	Syötteet		Naiivin tulokset ja aikavaatimukset		DP:n tulokset ja aikavaatimukset	
	Sakki (v)	Tavarat (n)	Naiivi (ns)	2^n	DP (ns)	$n \cdot v$
A188	188	30	32044198	1073741824	716	5640
A144	144	30	32255846	1073741824	601	4320
A100	100	30	20993099	1073741824	560	3000
A60	60	30	2091893	1073741824	56	1800
A20	20	30	1048	1073741824	26	600
B100	100	20	31553	1048576	28	2000
B60	60	20	15207	1048576	15	1200
B20	20	20	186	1048576	4	400
C60	60	10	39	1024	6	600
C20	20	10	13	1024	2	200

Graafisesti esitettynä DP:n tulokset ja aikavaatimukset vastakkain:

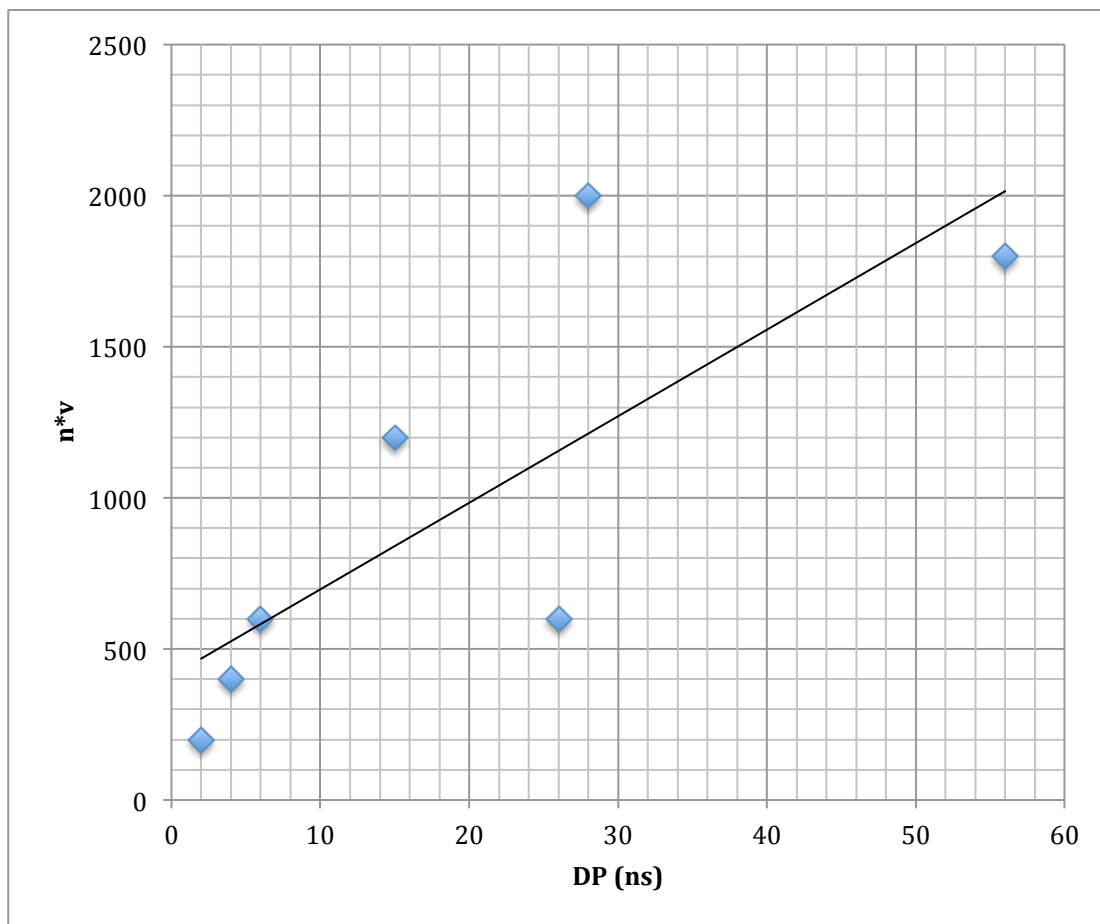


Muodostuu kaksi rykelmää: alas vasemmalle ja ylös. Voidaan sanoa, että tulokset ja aikavaatimukset eivät asetu erityisen lineaarisesti. Suurimmat 3 säkkikokoa kun $n=30$ erottuvat muusta joukosta selvästi. Tarkastellaan niitä pelkästään:

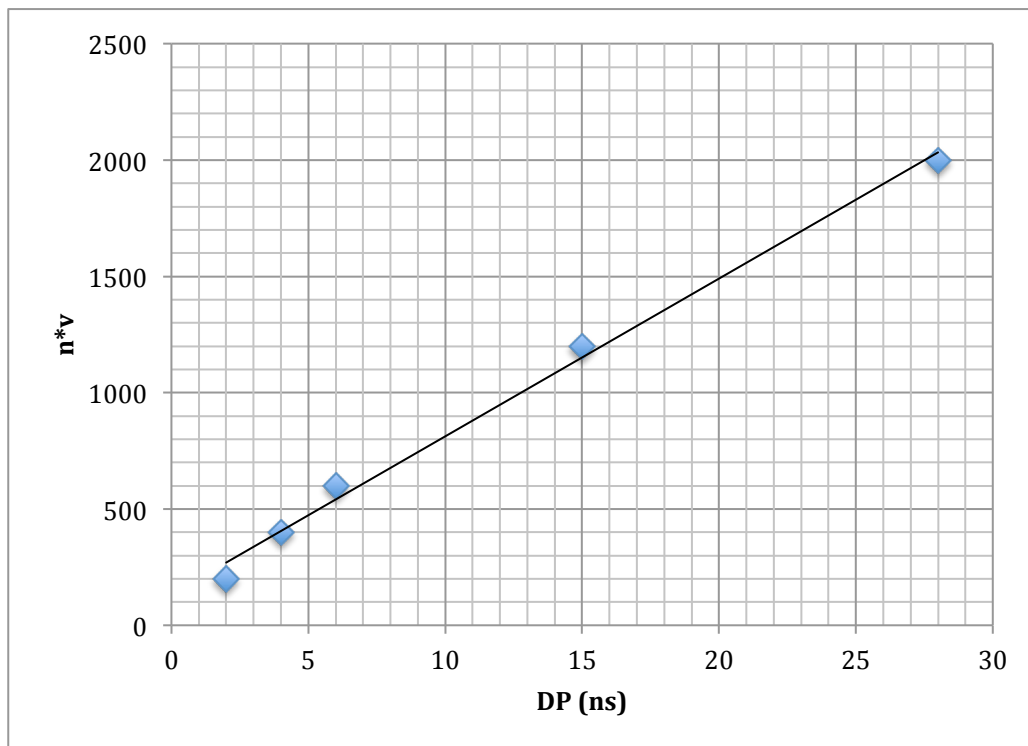


Nyt nähdään, että nämä kolme asettuvat suurin piirtein samalle suoralle, vaikka eroavat muista tuloksista paljon.

Tarkastellaan pienempien n:n rykelmää erikseen:

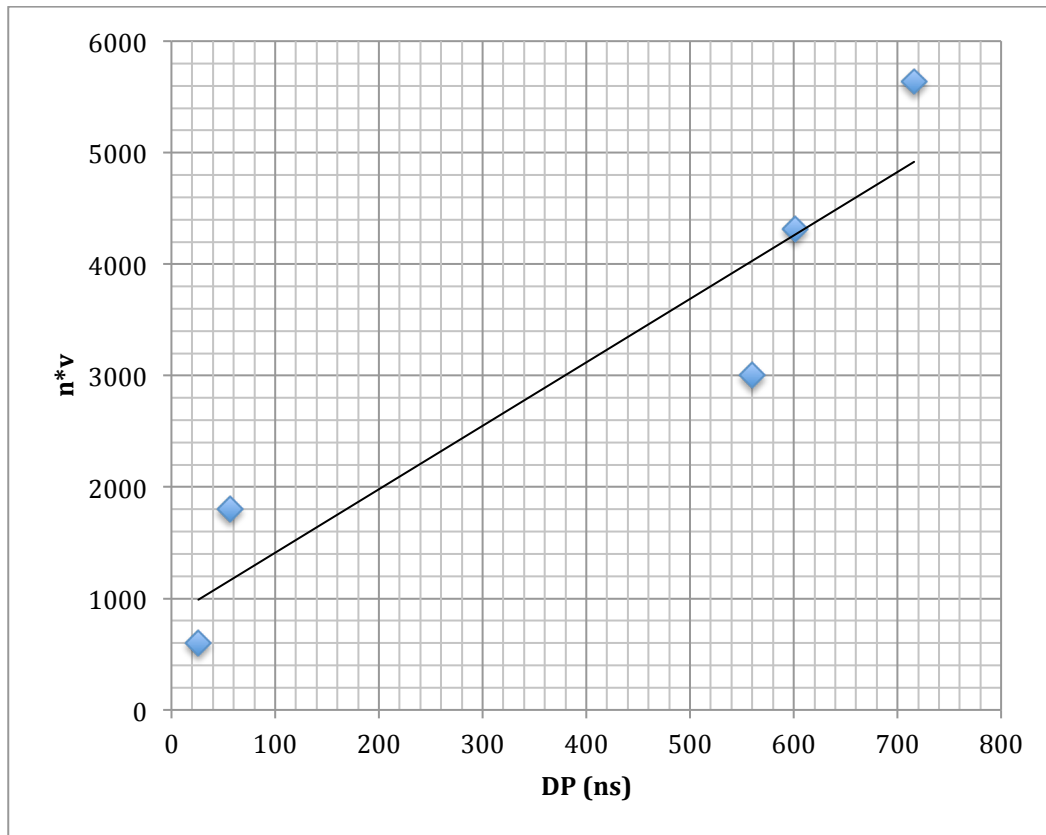


Tässä poikkeuksia ovat kaksi jäljellä olevaa $n=30$ tulosta. Tarkastellaan ilman niitä eli tuloksia, joissa $n=20$ tai 10:



Nyt asettuvat suurin piirtein samalle suoralle.

Tässä vielä kaikki kun $n=30$:



DP:sta yhteenvetona:

$n*v$ -toteutuu hyvin, kun $n=10$ tai 20 . Kun $n=30$, niin suurimmat säkkikoot saavat yli 10 kertaa suuremmat tulokset, kuin $n*v$ ennustaisi. Tähän ehkä syynä se, että ajettaessa vierekkäin Naiivin algoritmin kanssa muisti joutuu koetukselle, kun isommalla n :llä ja säkkikoolla Naiivi -varaa muistia paljon.

DP:n suoritusaikojen tarkempi tarkastelu vaatisi, että ne ajettaisiin ilman vertailevaa ajoa Naiivin kanssa, jotta olosuhteet esim. muistin suhteen olisivat stabiilimmat.

Naiivissa algoritmissa säkkikoko vaikuttaa tuloksiin siten, että verrattain pienet säkkikokojen täyttö on useita kertaluokkia nopeampaa kuin säkkikoot, jotka yli puolet tavaroiden yhteenlasketusta koosta. Tähän syynä, se että algoritmi osaa loetaa haaran käsittelyn, jos se on ylittänyt jo sallitun kokorajan. Alla olevassa taulukossa rajatuilla alueilla tämä nähdään selvästi:

Testin nimi	Syötteet		Naiivin tulokset ja aikavaatimukset		DP:n tulokset ja aikavaatimukset	
	Sakki (v)	Tavarat (n)	Naiivi (ns)	2^n	DP (ns)	n*v
A188	188	30	32044198	1073741824	716	5640
A144	144	30	32255846	1073741824	601	4320
A100	100	30	20993099	1073741824	560	3000
A60	60	30	2091893	1073741824	56	1800
A20	20	30	1048	1073741824	26	600
B100	100	20	31553	1048576	28	2000
B60	60	20	15207	1048576	15	1200
B20	20	20	186	1048576	4	400
C60	60	10	39	1024	6	600
C20	20	10	13	1024	2	200

Kun säkkikoko on verrattain suuri esim. 60, niin nähdään Naiivin algoritmin eksponentiaalinen kasvu suoritukselle n:n kasvaessa:

Testin nimi	Syötteet		Naiivin tulokset ja aikavaatimukset		DP:n tulokset ja aikavaatimukset	
	Sakki (v)	Tavarat (n)	Naiivi (ns)	2^n	DP (ns)	n*v
A60	60	30	2091893	1073741824	56	1800
B60	60	20	15207	1048576	15	1200
C60	60	10	39	1024	6	600

Suorituskyky testien suunnittelua

Olen ajanut erilaisia testejä pilvin pimein ja löytänyt seuraavia havaintoja: Vaihdelu tulosten välillä on ollut erittäin suurta samalla testisyötteellä. Tuloksista nähdään, että DP algoritmi on selvästi tehokkaampi, mutta ongelma on ollut tarkentaa kuinka paljon. Haarukka pahimmillaan ollut 20 - 100 kertaa tehokkaampi. Tähän näyttäisi vaikuttavan pitkälti se miten testejä ajetaan, johtuen mahdollisesti ajoympäristön vaikeasti ennustettavista vaihteluista ja mm. siitä millaisen painoarvon testin ensimmäiset ajot saavat. Olen nyt haarukoinut menetelmän, jolla vaihtelu on jo pienempää, mutta vaatii vielä muutamia tarkennuksia. Kun riittävä taso saavutettu, voin aloittaa avrsinaiset suorituskykytestit erilaisill apkeettien määrä ja jakauma sekä säkkien koko vaihtoehdoilla.

Tähän astinen testidata raakana alla:

Millisekunneissa mittaus:

10000000:n ajon summa Naiivilla: 24417, DP:lla: 18364

10000000:n ajon summa Naiivilla: 26366, DP:lla: 19428

10000000:n ajon summa Naiivilla: 26819, DP:lla: 19469

10000000:n ajon summa Naiivilla: 28503, DP:lla: 20496

Millisekunnit osoittautuivat liian karkeuksiksi, kun suuri osa DP-testeistä sai tulokseksi 0, joka aiheutti hajontaa paljon.

Tässä erinäistä data testiajoista:

100:n ajon keskiarvo (ns) Naiivilla: 2602, DP:lla: 174
100:n ajon keskiarvo (ns) Naiivilla: 1863, DP:lla: 257
1000:n ajon keskiarvo (ns) Naiivilla: 811, DP:lla: 36
1000:n ajon keskiarvo (ns) Naiivilla: 551, DP:lla: 30
1000:n ajon keskiarvo (ns) Naiivilla: 753, DP:lla: 37
1000:n ajon keskiarvo (ns) Naiivilla: 555, DP:lla: 30
1000:n ajon keskiarvo (ns) Naiivilla: 551, DP:lla: 28
1000:n ajon keskiarvo (ns) Naiivilla: 670, DP:lla: 35
1000:n ajon keskiarvo (ns) Naiivilla: 559, DP:lla: 29
1000:n ajon keskiarvo (ns) Naiivilla: 1184, DP:lla: 53

1:n ajon keskiarvo (ns) Naiivilla: 15315, DP:lla: 729
1:n ajon keskiarvo (ns) Naiivilla: 13178, DP:lla: 1573
1:n ajon keskiarvo (ns) Naiivilla: 22060, DP:lla: 1054
1:n ajon keskiarvo (ns) Naiivilla: 25831, DP:lla: 654
1:n ajon keskiarvo (ns) Naiivilla: 12836, DP:lla: 1174
1:n ajon keskiarvo (ns) Naiivilla: 10370, DP:lla: 764
1:n ajon keskiarvo (ns) Naiivilla: 16719, DP:lla: 698
1:n ajon keskiarvo (ns) Naiivilla: 10441, DP:lla: 678
1:n ajon keskiarvo (ns) Naiivilla: 27826, DP:lla: 7918
1:n ajon keskiarvo (ns) Naiivilla: 8441, DP:lla: 805
1:n ajon keskiarvo (ns) Naiivilla: 19304, DP:lla: 825

2:n ajon keskiarvo (ms) Naiivilla: 26135, DP:lla: 561
2:n ajon keskiarvo (ns) Naiivilla: 19124, DP:lla: 795
2:n ajon keskiarvo (ns) Naiivilla: 5943, DP:lla: 5043
2:n ajon keskiarvo (ns) Naiivilla: 8221, DP:lla: 756
2:n ajon keskiarvo (ns) Naiivilla: 6972, DP:lla: 513
2:n ajon keskiarvo (ns) Naiivilla: 8833, DP:lla: 608
2:n ajon keskiarvo (ns) Naiivilla: 7860, DP:lla: 724
2:n ajon keskiarvo (ns) Naiivilla: 13840, DP:lla: 554
2:n ajon keskiarvo (ns) Naiivilla: 13690, DP:lla: 787
2:n ajon keskiarvo (ns) Naiivilla: 8738, DP:lla: 3405

100:n ajon keskiarvo (ns) Naiivilla: 1364, DP:lla: 312
100:n ajon keskiarvo (ns) Naiivilla: 1797, DP:lla: 189
100:n ajon keskiarvo (ns) Naiivilla: 2008, DP:lla: 196
100:n ajon keskiarvo (ns) Naiivilla: 1247, DP:lla: 198
100:n ajon keskiarvo (ns) Naiivilla: 1000, DP:lla: 199
100:n ajon keskiarvo (ns) Naiivilla: 1177, DP:lla: 171
100:n ajon keskiarvo (ns) Naiivilla: 1020, DP:lla: 171
100:n ajon keskiarvo (ns) Naiivilla: 2552, DP:lla: 196
100:n ajon keskiarvo (ns) Naiivilla: 1381, DP:lla: 190
100:n ajon keskiarvo (ns) Naiivilla: 1604, DP:lla: 160

No random

1:n ajon keskiarvo (ns) Naiivilla: 8384, DP:lla: 445

1:n ajon keskiarvo (ns) Naiivilla: 8586, DP:lla: 657
1:n ajon keskiarvo (ns) Naiivilla: 11379, DP:lla: 444
1:n ajon keskiarvo (ns) Naiivilla: 9966, DP:lla: 433
1:n ajon keskiarvo (ns) Naiivilla: 15011, DP:lla: 586
1:n ajon keskiarvo (ns) Naiivilla: 38830, DP:lla: 641
1:n ajon keskiarvo (ns) Naiivilla: 8841, DP:lla: 391
1:n ajon keskiarvo (ns) Naiivilla: 16526, DP:lla: 381
1:n ajon keskiarvo (ns) Naiivilla: 9132, DP:lla: 444
1:n ajon keskiarvo (ns) Naiivilla: 12912, DP:lla: 471

2:n ajon keskiarvo (ns) Naiivilla: 17290, DP:lla: 323
2:n ajon keskiarvo (ns) Naiivilla: 17643, DP:lla: 311
2:n ajon keskiarvo (ns) Naiivilla: 9916, DP:lla: 437
2:n ajon keskiarvo (ns) Naiivilla: 18190, DP:lla: 385
2:n ajon keskiarvo (ns) Naiivilla: 15709, DP:lla: 368
2:n ajon keskiarvo (ns) Naiivilla: 6596, DP:lla: 350
2:n ajon keskiarvo (ns) Naiivilla: 10085, DP:lla: 867
2:n ajon keskiarvo (ns) Naiivilla: 17836, DP:lla: 1526
2:n ajon keskiarvo (ns) Naiivilla: 8496, DP:lla: 460
2:n ajon keskiarvo (ns) Naiivilla: 8875, DP:lla: 515
2:n ajon keskiarvo (ns) Naiivilla: 7660, DP:lla: 1778
2:n ajon keskiarvo (ns) Naiivilla: 11877, DP:lla: 355
2:n ajon keskiarvo (ns) Naiivilla: 28311, DP:lla: 378

10:n ajon keskiarvo (ns) Naiivilla: 4656, DP:lla: 247
10:n ajon keskiarvo (ns) Naiivilla: 4999, DP:lla: 237
10:n ajon keskiarvo (ns) Naiivilla: 11347, DP:lla: 235
10:n ajon keskiarvo (ns) Naiivilla: 7047, DP:lla: 272
10:n ajon keskiarvo (ns) Naiivilla: 4777, DP:lla: 269
10:n ajon keskiarvo (ns) Naiivilla: 4994, DP:lla: 261
10:n ajon keskiarvo (ns) Naiivilla: 4453, DP:lla: 248
10:n ajon keskiarvo (ns) Naiivilla: 8425, DP:lla: 792
10:n ajon keskiarvo (ns) Naiivilla: 22323, DP:lla: 358
10:n ajon keskiarvo (ns) Naiivilla: 5470, DP:lla: 305

100:n ajon keskiarvo (ns) Naiivilla: 1034, DP:lla: 177
100:n ajon keskiarvo (ns) Naiivilla: 2523, DP:lla: 315
100:n ajon keskiarvo (ns) Naiivilla: 926, DP:lla: 176
100:n ajon keskiarvo (ns) Naiivilla: 1701, DP:lla: 208
100:n ajon keskiarvo (ns) Naiivilla: 2477, DP:lla: 328
100:n ajon keskiarvo (ns) Naiivilla: 1032, DP:lla: 158
100:n ajon keskiarvo (ns) Naiivilla: 1152, DP:lla: 165
100:n ajon keskiarvo (ns) Naiivilla: 1428, DP:lla: 169
100:n ajon keskiarvo (ns) Naiivilla: 1082, DP:lla: 161
100:n ajon keskiarvo (ns) Naiivilla: 1516, DP:lla: 189

Random sisätoisto 1000

2:n ajon keskiarvo (ms) Naiivilla: 964, DP:lla: 33
2:n ajon keskiarvo (ms) Naiivilla: 802, DP:lla: 36
2:n ajon keskiarvo (ms) Naiivilla: 452, DP:lla: 23
2:n ajon keskiarvo (ms) Naiivilla: 546, DP:lla: 28
2:n ajon keskiarvo (ms) Naiivilla: 445, DP:lla: 24
2:n ajon keskiarvo (ms) Naiivilla: 445, DP:lla: 25
2:n ajon keskiarvo (ms) Naiivilla: 608, DP:lla: 28
2:n ajon keskiarvo (ms) Naiivilla: 1197, DP:lla: 66

1000:n ajon keskiarvo (ms) Naiivilla: 844, DP:lla: 22
1000:n ajon keskiarvo (ms) Naiivilla: 321, DP:lla: 8
1000:n ajon keskiarvo (ms) Naiivilla: 300, DP:lla: 7
1000:n ajon keskiarvo (ms) Naiivilla: 278, DP:lla: 7