



Aalto-yliopisto
Perustieteiden
korkeakoulu

Scheduling of Genetic Analysis Workflows on Grid Environments (valmiin työn esittely)

Arttu Voutilainen

20.4.2015

Ohjaaja: FT Lauri Eronen (Biocomputing Platforms Ltd.)

Valvoja: Prof. Harri Ehtamo

Työn saa tallentaa ja julkistaa Aalto-yliopiston avoimilla verkkosivuilla. Muilta osin kaikki oikeudet pidätetään.

Tausta

DNA:n kartoitus halpenee jatkuvasti

- Raakaa dataa saadaan todella paljon (1000 yksilön genomien sekvenointi: 16TB)
- Datan analysointi on laskennallisesti raskasta (BWA/GATK yhdellä suorittimella > 20 vuotta)

Analysointi on mahdollista jakaa osiin, joita suoritetaan rinnakkain ja peräkkäin

- Täytyy valita suoritusjärjestys
- Aikataulutus (scheduling): $M : W \rightarrow R \times t$
- Tiedonsiirto resurssien välillä oleellista

NP-täydellinen ongelma!

- Ratkaisuna erilaisia heuristiikkoja ja meta-heuristiikkoja (Min-Min, geneettinen algoritmi,...)

Tavoitteet

Vertailla eri ajoitusmenetelmien (scheduler) sopivuutta annettuihin esimerkkitapauksiin

- Ajoitukseen kuluva aika (scheduling time)
- Aikataulun kokonaispituus (makespan)

Valita ajoitusmenetelmä, jonka avulla työt voitaisiin suorittaa mahdollisimman tehokkaasti ja nopeasti

Ei oteta huomioon QoS-rajoituksia

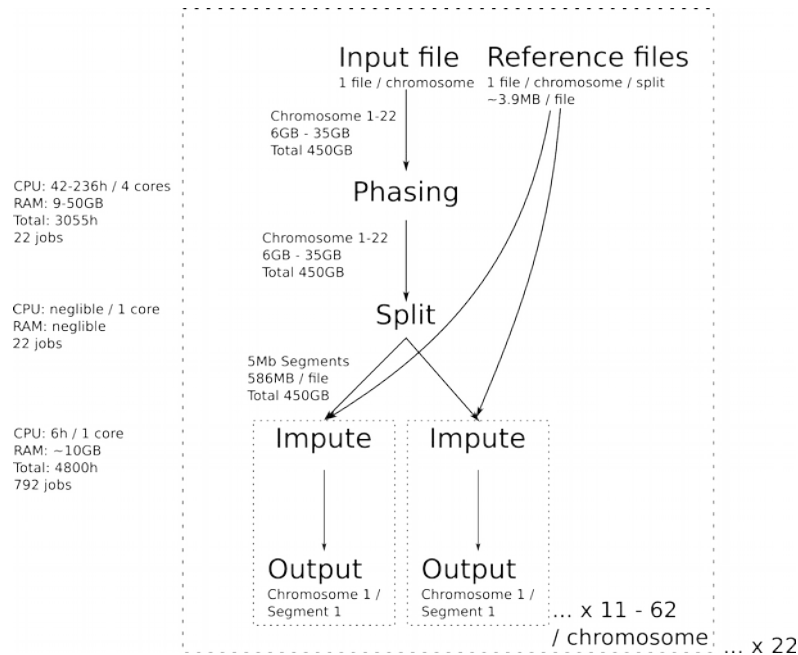
Ei oteta kantaa työnkuvauksien suunnitteluun tai virheiden hallintaan

Tarkastellaan vain yhden työnkulun ajoittamista

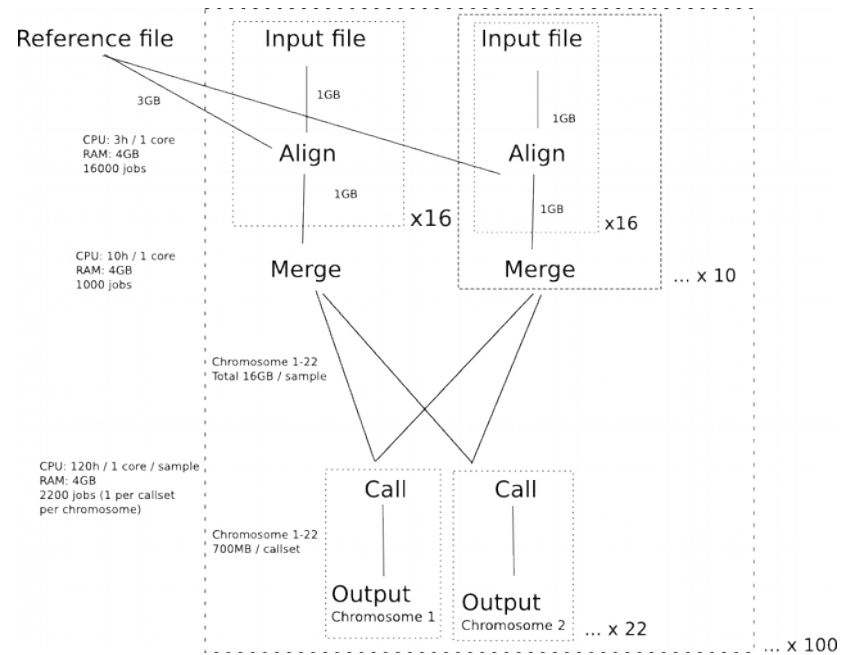
Yksinkertaistuksia resurssienhallinnassa (rajattomat levyt, ei huomioida yhtäaikaisten siirtojen vaikutusta kaistaan,...)

Työnkulut

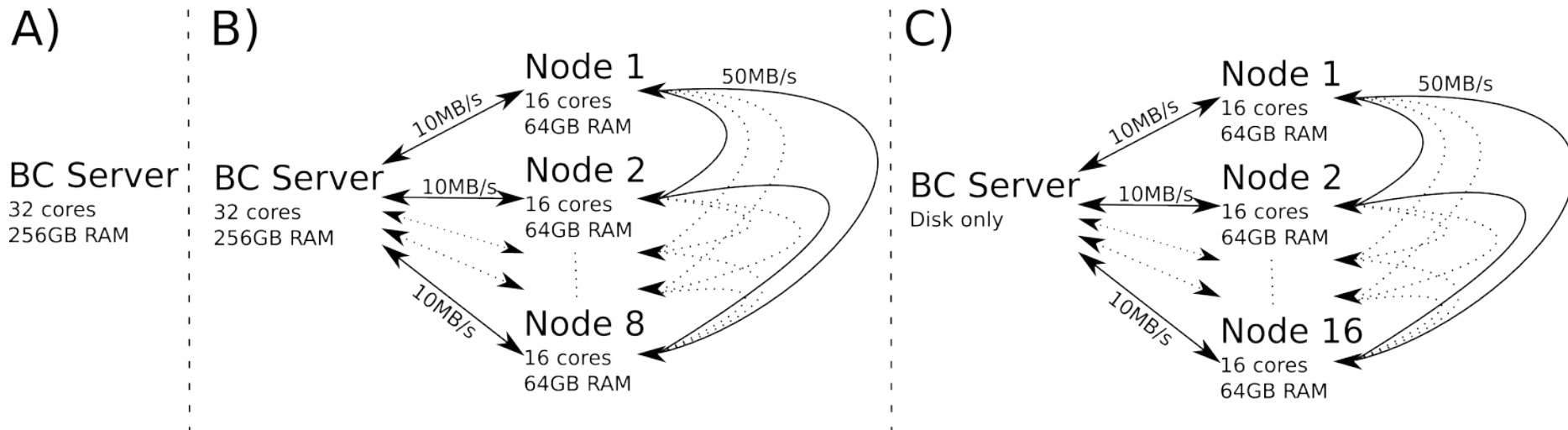
Imputation Puuttuvien arvojen täydentäminen



BWA/GATK Datan pelkistäminen eroavaisuuksiin



Suoritusympäristöt



Tulokset – Eksakti ratkaisu

“Job Shop”-ongelma, myös NP-täydellinen

- JSP samankaltainen kuin tämä, mutta huomattavasti yksinkertaisempi

JSP ratkaistavissa eksaktisti vielä kun laajuus on 10×10

- mutta 15×15 ei enää järkevästi

BWA/GATK: 100 rinnakkaista työtä, jotka jakaantuvat 10-160 rinnakkaiseen työhön

Tulokset – Kirjallisuuskatsaus

Taksonomia: “centralized with performance-driven scheduling strategy” (*Yu and Buyya [2005]*)

Algoritmit: Myopic, Min-Min, Max-Min, Suffrage, HEFT, Hybrid, GRASP, Geneettinen Algoritmi (*Yu et al. [2008]*)

Myopic vs. HEFT vs. GA: HEFT antoi parhaat tulokset (*Wieczorek et al. [2005]*)

Algoritmeissa paljon säädettävää: HEFT:n eri rank-vaihtoehtojen välillä jopa 40% eroja (*Zhao and Sakellariou [2003]*)

Tulokset – Kirjallisuuskatsaus

Lyhin valmistumisaika (LVT)

Myopic: Valitse työ, aseta resurssille jolla pienin LVT

Min-Min: Valitse työ jonka LVT pienin

Max-Min: Valitse työ jonka LVT suurin

Suffrage: Valitse työ jolla suurin ero LVT:n ja seuraavan valmistumisajan välillä

HEFT: Arvota työt seuraajien ja vaatimusten perusteella

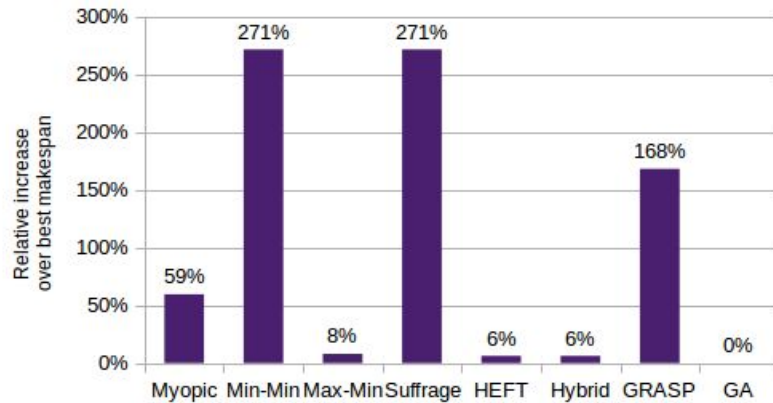
Hybrid: Jaa arvotetut työt ryhmiin, joiden sisällä ei riippuvuuksia

GRASP: Muodosta satunnaisia aikatauluja valitsemalla (työ,resurssi)-pareja jotka eivät ole liian huonoja ja pyri optimoimaan lopputulos, iteroi

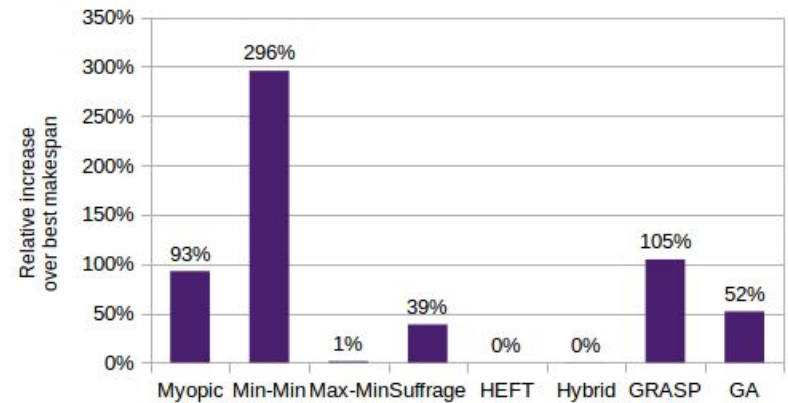
GA: Luo satunnaisia aikatauluja, iteroi geneettisiä operaattoreita

Tulokset – Imputation: Aikataulut

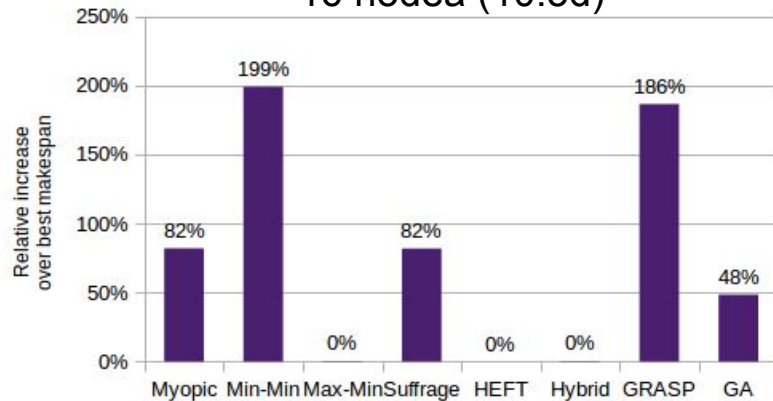
Vain BC Server (25.0d)



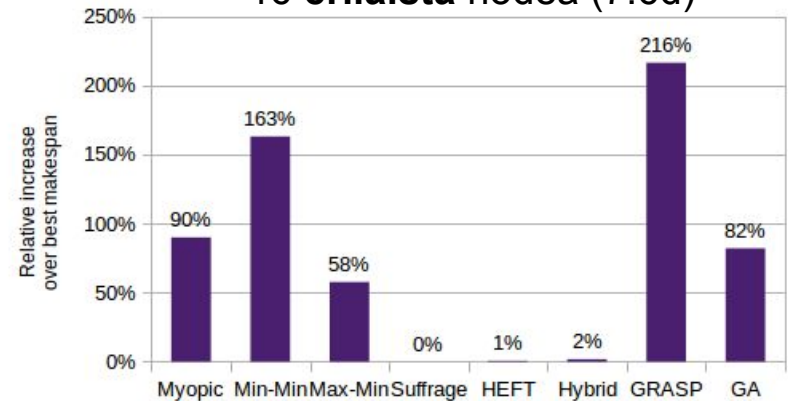
BC Server + 8 nodea (10.6d)



16 nodea (10.5d)

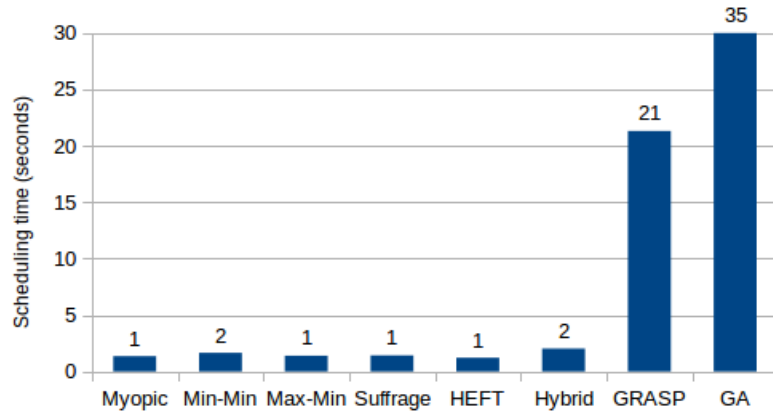


16 erilaista nodea (7.6d)

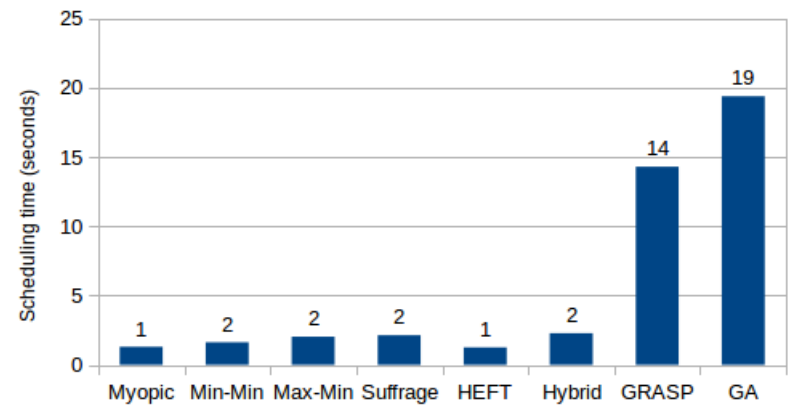


Tulokset – Imputation: Ajoaika

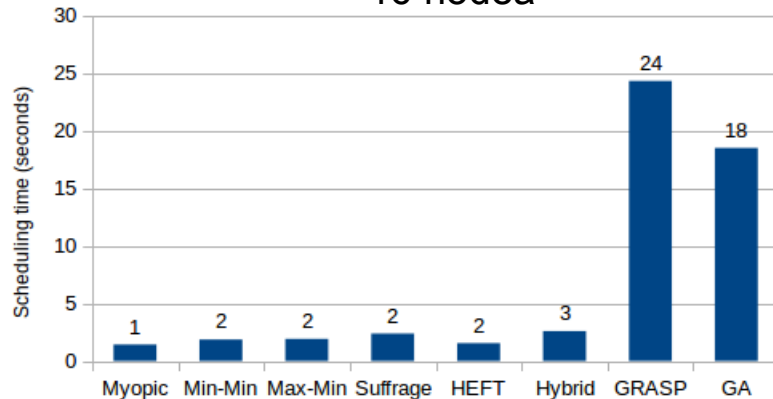
Vain BC Server



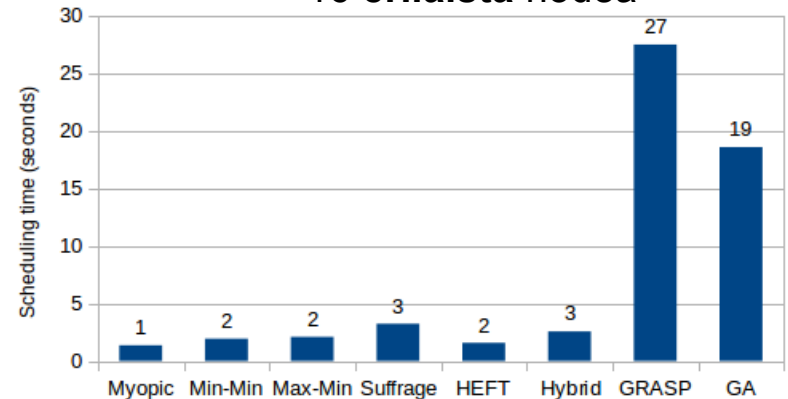
BC Server + 8 nodea



16 nodea

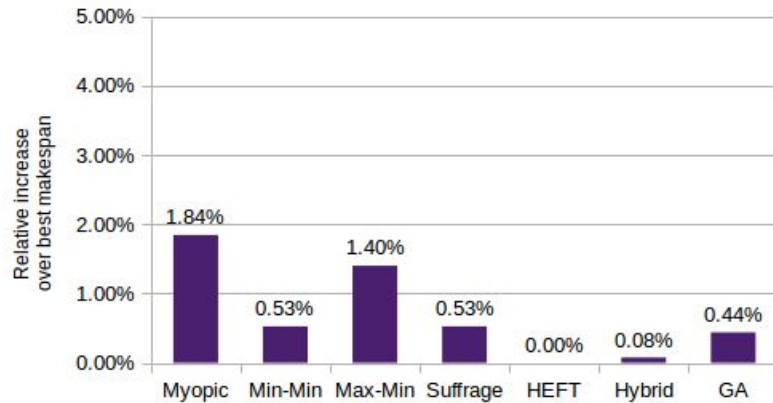


16 erilaista nodea

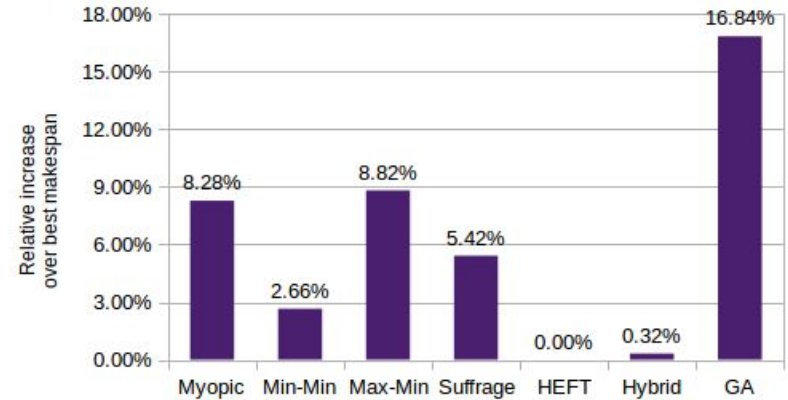


Tulokset – BWA/GATK: Aikataulut

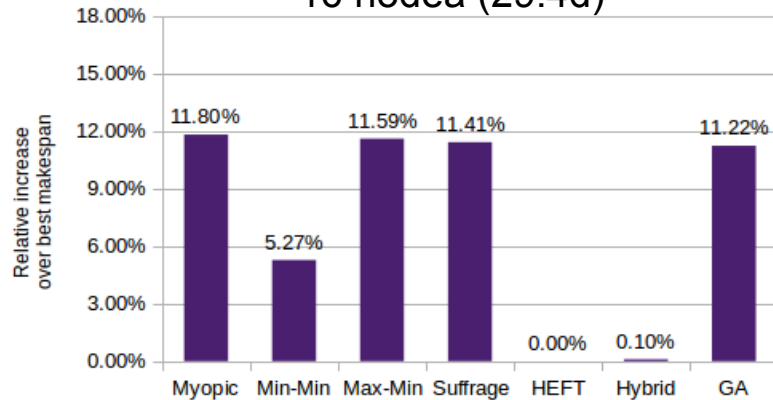
Vain BC Server (232.1d)



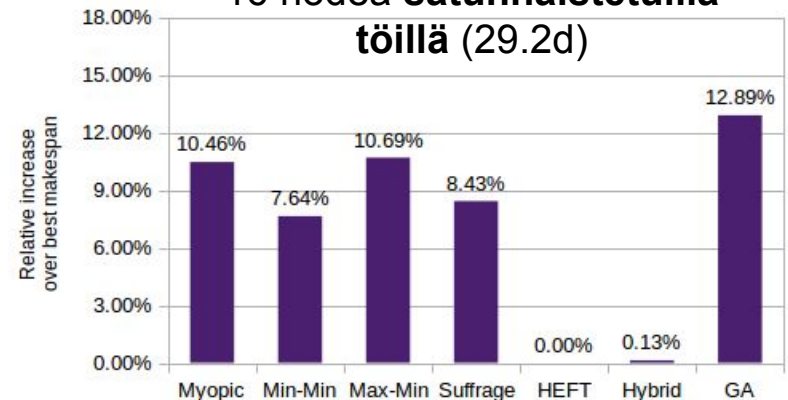
BC Server + 8 nodea (46.7d)



16 nodea (29.4d)

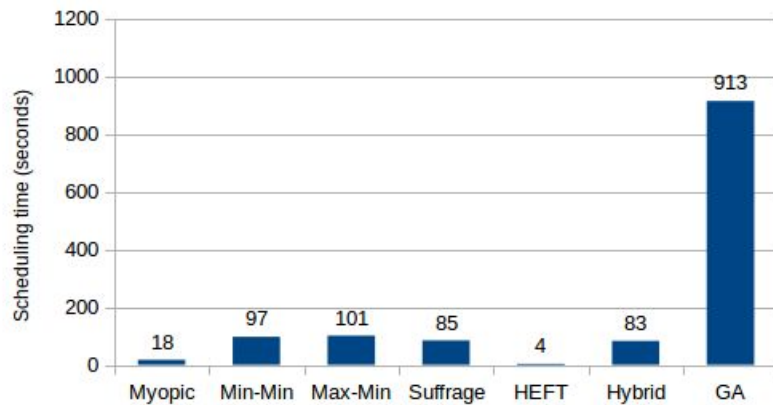


16 nodea **satunnaistetuilla töillä** (29.2d)

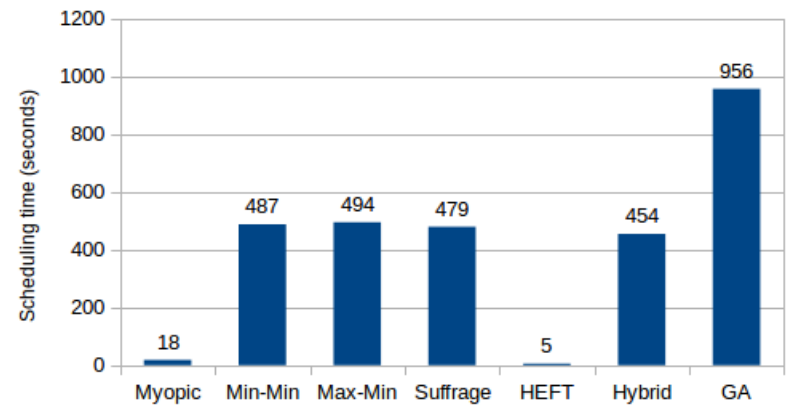


Tulokset – BWA/GATK: Ajoaika

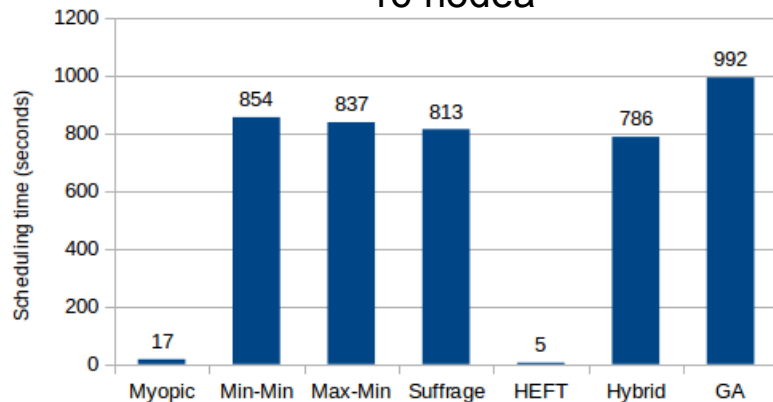
Vain BC Server



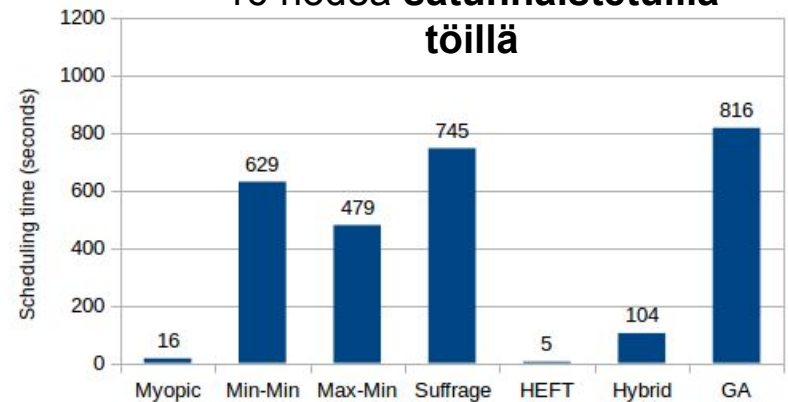
BC Server + 8 nodea



16 nodea

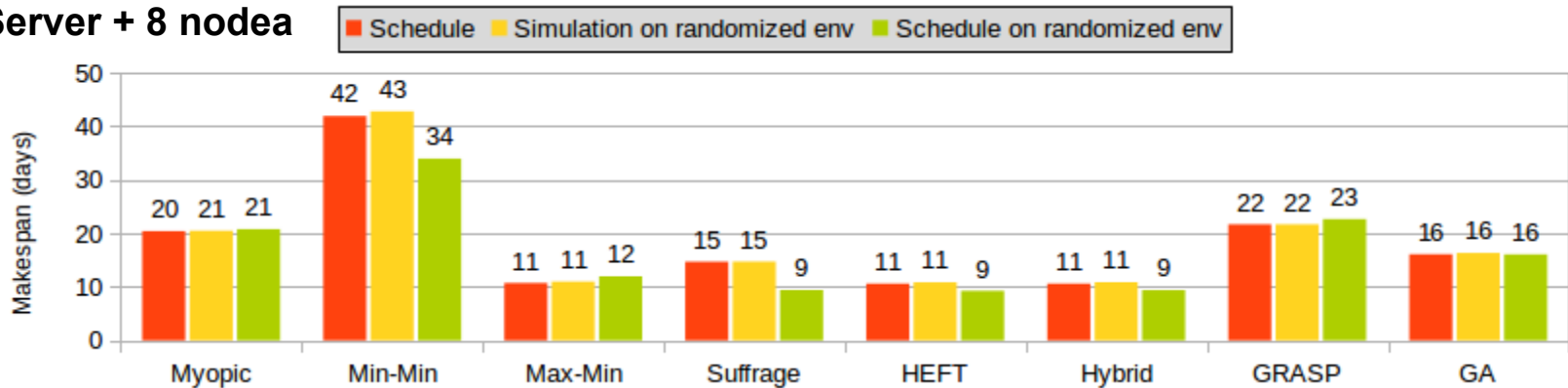


16 nodea satunnaistetuilla töillä

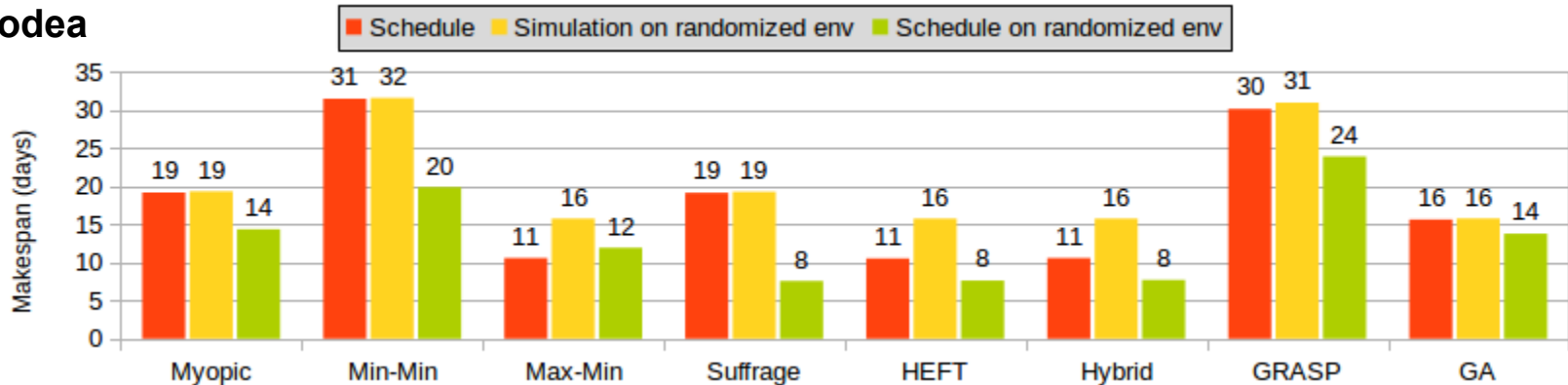


Tulokset – Impute: Suorituskykytietojen vaikutus

BC Server + 8 nodea



16 nodea



Johtopäätökset

Sopivan menetelmän käyttö on tärkeää

- Lyhimmän ja pisimmän aikataulun ero jopa satoja prosentteja

Ajoitusmenetelmien hyvyys riippuu työnkulusta ja ympäristöstä

- Imputella Max-Min parempi kuin Min-Min, BWA/GATK:lla toisin päin
- Suffrage huono yhdellä resurssilla, hyvä heterogeenisillä resursseilla

Heuristiikat ovat nopeita

- Impute (~800 työtä): < 5s, BWA/GATK (~20 000): < 15min

Metaheuristiikat ovat hitaita eivätkä tuo lisäarvoa

- Todennäköisesti osittain implementaatiosta johtuen

HEFT (Heterogenous Earliest Finish Time) on **nopein** ja antaa **parhaan** (tai hyvin lähellä parasta olevan) tuloksen

Tietolähteet

- Ian Foster and Carl Kesselman. ***The Grid 2: Blueprint for a new computing infrastructure.*** Morgan Kaufmann, 2003.
- H. Topcuoglu, S. Hariri, and M. Wu. **Performance-effective and low-complexity task scheduling for heterogeneous computing.** *Parallel and Distributed Systems, IEEE Transactions on*, 13(3): 260--274, 2002.
- J. D. Ullman. **Np-complete scheduling problems.** *Journal of Computer and System Sciences*, vol. 10, pages 384--393, 1975.
- M. Wiecek, R. Prodan, and T. Fahringer. **Scheduling of scientific workflows in the askalon grid environment.** *ACM SIGMOD Record*, 34(3): 56--62, 2005.
- J. Yu and R. Buyya. **A taxonomy of scientific workflow systems for grid computing.** *Sigmod Record*, 34(3): 44, 2005.
- J. Yu, R. Buyya, and K. Ramamohanarao. **Workflow scheduling algorithms for grid computing.** In *Metaheuristics for scheduling in distributed computing environments*, pages 173--214. Springer, 2008.
- H. Zhao and R. Sakellariou. **An experimental investigation into the rank function of the heterogeneous earliest finish time scheduling algorithm.** In *Euro-Par 2003 Parallel Processing*, pages 189--194. Springer, 2003.