

PhD thesis: Integrity issues on the cloud

Jakub Muszyński

Computer Science and Communications (CSC) Research Unit, University of Luxembourg, Luxembourg

Yearly Team Meeting 2011
October 18th



BOINC¹ platform characteristics

- Based on volunteer computing
 - ↪ Use idle cycles of voluntarily shared machines
 - ↪ Heterogeneous platform with high volatility
 - ↪ Reward system (credit points, hall of fame etc.)
- Largest and most powerful distributed computing systems
 - ↪ 2,000,000 users / 6,000,000 machines / 4 PetaFlops

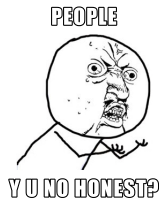
¹Berkeley Open Infrastructure for Network Computing

BOINC¹ platform characteristics

- Based on volunteer computing
 - ↳ Use idle cycles of voluntarily shared machines
 - ↳ Heterogeneous platform with high volatility
 - ↳ Reward system (credit points, hall of fame etc.)
- Largest and most powerful distributed computing systems
 - ↳ 2,000,000 users / 6,000,000 machines / 4 PetaFlops

Incentives attract cheaters!

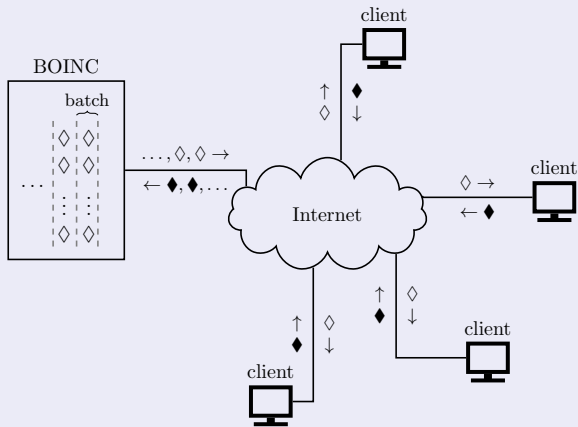
- Seek to obtain rewards with little/no contribution
- **Cheating faults** (or *fault by value*) modelled by result falsification
 - ↳ typically done via software tampering / man-in-the-middle attack
 - ↳ sub-class of byzantine failures, more complex than *crash-faults*



¹Berkeley Open Infrastructure for Network Computing

BOINC platform overview

Typical work-flow



ABFT for cheating faults

Algorithm-Based Fault-Tolerance (ABFT)

- Error detection/correction technique tailored to the algorithm
 - ↳ **Inherent tolerance of a limited number of faults**
 - ↳ Permits resilience to byzantine failures
 - ↳ Avoid overhead of checkpoint/restart methods

- Mostly (if not only) applied against *crash-faults* [Huang&al.84],[Chen Dongarra 96]

EAs and Fault-tolerance

- **Observation** of natural resilience of EAs against destructive faults
 - ↪ More or less *crash faults* model
 - ↪ Experiments conducted several GA/GP contexts
 - [DEPCOS06]: Ant problem analysis (sequential GP)
 - [AHS08]: cGA to deal with hardware components failure
 - [GECCO07]: coarse-grain dEA for GA benchmarks problems.
(analysis of the Fmodal or the Schwefel's function)

[DEPCOS06]

FF De Vega. *A Fault Tolerant Optimisation Algorithm based on Evolutionary Computation*. 2006

[AHS08]

A. Morales-Reyes & al. *Towards Fault-Tolerant Systems based on Adaptive Cellular Genetic Algorithms*, 2008

[GECCO07]

J.I Hidalgo & al. *Is the island model fault tolerant?*, 2007

Cheating characteristics

- Cheaters know everything
 - ↳ Every cheater wants to be maximally effective in his actions
- Model – Bernoulli process with (unknown) probability p_{cheat}
- For every task T in the system, exists its cheated version T'
 - ↳ Prototype of T' unaffected by cheating
 - ↳ Result of T' altered with worst impact on the evolutionary process without consuming more computing power than T (**lazy cheater**)
i.e.

$$W_1(T') \leq W_1(T) \text{ and } W_\infty(T') \leq W_\infty(T) \quad (1)$$

Current focus

- Theoretical proofs of convergence of [distributed] Evolutionary Algorithms with cheaters in the network
 - ↳ Based on Markov chains and transition kernels
 - ↳ Publication in a journal in preparation

Future work

- Analysis of different cheaters models
- Sensitivity analysis for Evolutionary Algorithms with cheaters
- Similar analysis for cellular automata

Thank you for your attention...