ΔΕΣΜΗ 2009-2010

Research Promotion Foundation

ΚΥΠΡΙΑΚΗ ΔΗΜΟΚΡΑΤΙΑ

Algorithmic Mechanisms for Reliable Internet Supercomputing

Chryssis Georgiou

Assistant Professor
University of Cyprus

March 6th, 2012
Nicosia, Cyprus

This work is supported in part by the Cyprus Research Promotion Foundation grant ΤΠΕ/ΠΛΗΡΟ/0609(ΒΕ)/05

Outline

- Motivation and preliminaries
- Framework and accomplishments
- Algorithmic Mechanism
- Application of the mechanism
  - SETI-like scenario
  - Contractor scenario
- Conclusions

Motivation and Preliminaries

K Computer

Top Three Supercomputers (2012)

- **K Computer**, RIKEN Advanced Institute for Computational Science (2011), Japan
  - CPUs: 88,128 SPARC64 VIIIfx 8-core 2.0GHz
  - 11,280 TFLOPS (11.2 PetaFLOPS)
- **Tianhe-1A,** Tianjin National Supercomputer Center, (2010), China
  - CPUs: 186,368 NUDT X5670 6-core 2.93GHz
  - 4,701 TFLOPS (4.7 PetaFLOPS)
- **Jaguar Cray XT5-HE,** Cray Inc (2009), USA
  - CPUs: 224,162 AMD x86_64 Opteron 6-core 2.6GHz

  2,331 TFLOPS (2.33 PetaFLOPS)

€ € € € € €

Computational Tasks

- Increasing demand for processing complex computational tasks
  - One-processor machines have limited computational resources
  - Powerful parallel machines (supercomputers) are expensive and are not globally available
- Intern… emerges as a viable platform for supercomputing
  - P2P, …d computing
    - e.g., EGEE Grid, … Amazon's EC2
  - Volunteer Master-Worker … me projects
    - e.g., SETI@home, AIDS@home, Folding… PrimeNet

Internet-based Computing

SETI

- Search for ExtraTerrestrial Inteligence
- Internet-based public volunteer computing project
  - Employs the BOINC software platform
  - Hosted by the Space Sciences Laboratory, at the University of California, Berkeley, USA
- Purpose: analyze radio (telescopic) signals, searching for signs of extra terrestrial intelligence
- How to use it:
  - Register your PC
  - Downloads the SETI data analyzer (screensaver mode)
  - When PC is idling, it starts analyzing data
  - When done, sends results, gets new data chunk to analyze

http://setiathome.berkeley.edu/

Arecibo Radio Telescope, Puerto Rico

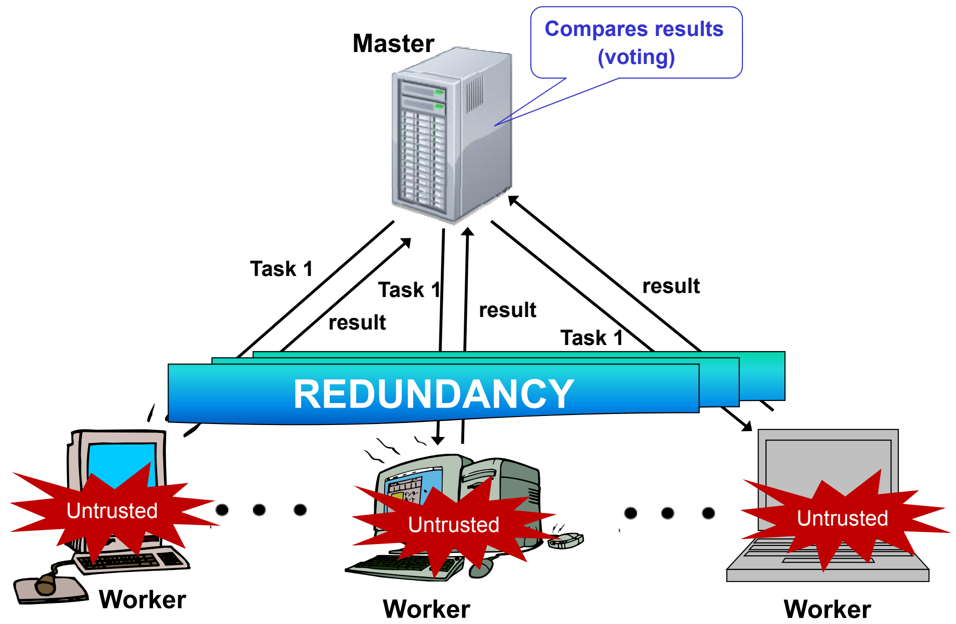
SETI@home by the numbers

- As reported in November 2009
  - 278,832 active CPUs (out of a total of 2.4 million) in 234 countries
  - 769 TFLOPs

Comparable processing power with top Supercomputers @ a fraction of the cost!

Great potential limited by untrustworthy entities

SETI-like Internet-based Computing



Redundant Task-Allocation

Two different approaches:

- "Classical" distributed computing: pre-defined worker behavior
  - Malicious workers fabrigate and report a bogus result
  - Altruistic workers compute and truthfully report correct result

  Malicious-tolerant voting protocols are designed

  [Sarmenta 2002, Fernandez et al 2006, Konwar et al 2006]
- Game-theoretic: workers act upon their best interest
  - Workers are Rational, i.e., they act selfishly aiming to maximize their own benefit

  Incentives are provided to induce a desired behavior

  [Yurkewych et al 2005, Fernandez et al 2008]

BUT realistically, the three types of workers may coexist!

Our Approach

Consider all worker types

- Types of workers:
  - **malicious:** always report incorrect result
  - **altruistic:** always compute and report correct result
  - **rational:** selfishly choose to be honest or a cheater
- Combine the two approaches
  - Game-theoretic approach:
    - Computations modeled as strategic games
    - Provide incentives to induce desired rationals behavior
    - Master chooses whether to audit the returned result or not
  - Classical distributed computing approach:
    - Design malice-aware voting protocols
- Objective: Reliable Internet-based Master-Worker Computing with *provable guarantees*

Background

A game consists of a set of players, a set of strategies available to those players, and a specification of payoffs (utilities) for each combination of strategies [wikipedia]

- Game Theory:
  - Players (processors) act on their self-interest
  - Rational [Golle Mironov 01] behavior: seek to increase their utility
  - Protocol is given as a game
  - Design objective is to achieve equilibrium among players

Nash Equilibrium (NE): players do not increase their expected utility by changing strategy, if other players do not change their strategy [Nash 50]

Algorithmic Mechanism Design

- Games are designed to provide necessary incentives such that rational players act "correctly"
  - Behave well: Reward
  - Otherwise: Penalize
- The design objective is to induce a desired behavior (e.g., a unique NE)

[Nisan Ronen 01]

Framework and Accomplishments

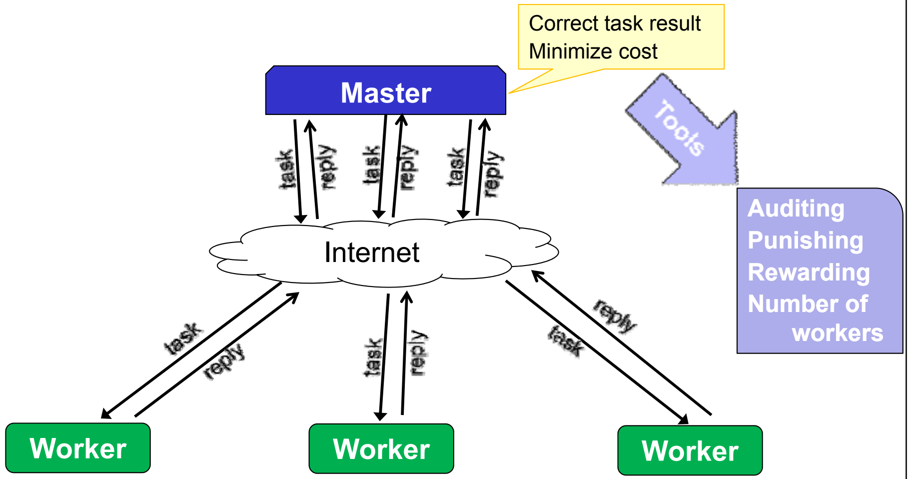
Framework

- Developed a general framework that captures the essential characteristics of existing Master-Worker platforms
  - Assuming communication between the master and the workers is reliable
  - Assuming that communication might be unreliable and workers may be unavailable
- Workers' types:
  - Unknown type of workers → Bayesian game [Harsanyi '67]
  - Known probability distribution over types

p_ρ : Rational p_μ : Malicious p_α : Altruistic

such that $p_\rho + p_\mu + p_\alpha = 1$

General Framework (reliable com.)



Untrusted: malicious or selfish

Reply with an incorrect result (cheat)

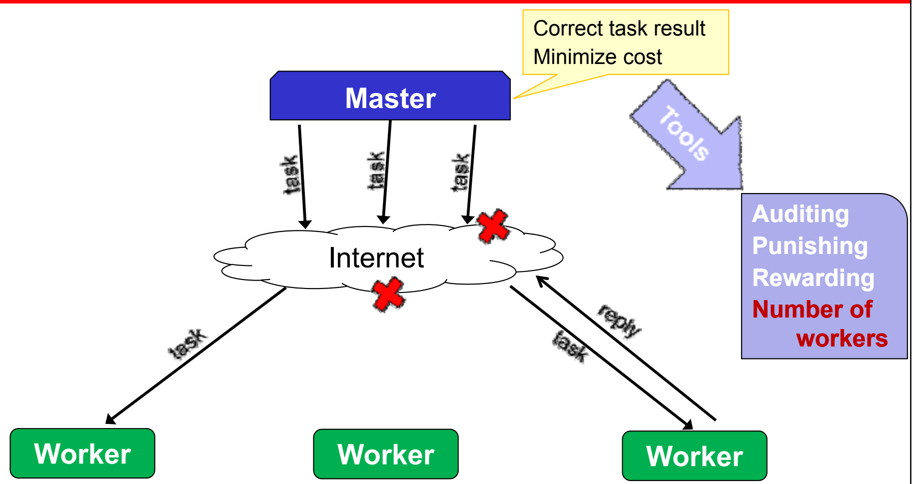
Accomplishments (reliable com.)

Reliable communication assumed

- Designed an algorithmic mechanism
  - Provides, when necessary, incentives to rational workers to act correctly so that
    - Master obtains correct task result (whp)
    - Despite malicious workers actions
- Analyzed the mechanism under two existing Internet-based Master-Worker settings
  - SETI-like volunteer computing systems
  - Profit-seeking Internet-based computational service

  Provide clear tradeoffs between reliability and cost under different system parameters

General Framework (unreliable com.)



Untrusted: malicious or selfish

Cheat or abstain (do not reply)

Accomplishments (unreliable com.)

Unreliable communication, worker unavailability

- Designed two algorithmic mechanisms
  - Provides, when necessary, incentives to rational workers to act correctly so that
    - Master obtains correct task result (whp)
    - Despite malicious workers actions and network unreliability

  Both mechanisms are useful in different situations

  When communication is reliable, we get the mechanism of the reliable communication case
- Analyzed the mechanisms under the two mentioned application-examples
  - Provides clear tradeoffs between reliability, cost and network unreliability

ALGORITHMIC MECHANISM
[RELIABLE COMMUNICATION]

Master Protocol

- Master assigns a task to n workers and collect replies
- Rational workers cheat with probability p_C
- Master audits the responses with probability $p_{\mathcal{A}}$
- If master audits
  - rewards honest workers and
  - penalizes the cheaters
- If master does not audit
  - Accepts value returned by majority of workers
  - Rewards/penalizes according to one of four reward models

$\mathcal{R}_m$	the master rewards the majority only
$\mathcal{R}_a$	the master rewards all workers
$\mathcal{R}_\emptyset$	the master does not reward any worker
$\mathcal{R}_\pm$	the master rewards the majority and penalizes the minority

Payoff parameters

Payoff parameters

$WP_{\mathcal{C}}$	worker's punishment for being caught cheating
$WC_{\mathcal{T}}$	worker's cost for computing the task
$WB_{\mathcal{Y}}$	worker's benefit from master's acceptance
$MP_{\mathcal{W}}$	master's punishment for accepting a wrong answer
$MC_{\mathcal{Y}}$	master's cost for accepting the worker's answer
$MC_{\mathcal{A}}$	master's cost for auditing worker's answers
$MB_{\mathcal{R}}$	master's benefit from accepting the right answer

Note that it is possible that $WB_{\mathcal{Y}} \neq MC_{\mathcal{Y}}$

Master's Goals

- Obtain the correct task result with a parameterized probability: $P_{wrong} \leq \varepsilon$
- Then increase its utility (benefit): U_M
- Depending on the type distribution, the master might or might not rely on rational workers
- The master must choose the auditing probability $p_{\mathcal{A}}$ in such a way, to "force", when needed, the rational workers to act correctly ($p_C = 0$)
- We computed the equilibrium conditions under general payoffs values and system parameters

Equilibrium Conditions

Guaranteeing : $P_{wrong} \leq \varepsilon$ While maximizing U_M

Pr(master obtains wrong answer):

$$P_{wrong} = (1-p_{\mathcal{A}})\mathbf{P}_q^{(n)}(\lceil n/2 \rceil, n)$$

E(utility of master):

$$U_M = p_{\mathcal{A}}(MB_{\mathcal{R}} - MC_{\mathcal{A}} - n(1-q)MC_{\mathcal{Y}}) +$$
$$(1-p_{\mathcal{A}})(MB_{\mathcal{R}}\mathbf{P}_q^{(n)}(0, \lfloor n/2 \rfloor) - MP_{\mathcal{W}}\mathbf{P}_q^{(n)}(\lceil n/2 \rceil, n) + \gamma)$$

where

$$\gamma = \begin{cases} -MC_{\mathcal{Y}}(\mathbf{E}_{1-q}^{(n)}(\lceil n/2 \rceil, n) + \mathbf{E}_q^{(n)}(\lceil n/2 \rceil, n)) & \mathcal{R}_{\mathrm{m}} \text{ and } \mathcal{R}_{\pm} \text{ models} \\ -nMC_{\mathcal{Y}} & \mathcal{R}_{\mathrm{a}} \text{ model} \\ 0 & \mathcal{R}_{\emptyset} \text{ model} \end{cases}$$

$$\mathbf{E}_p^{(n)}(a,b) = \sum_{i=a}^{b} \binom{n}{i} i p^i (1-p)^{n-i}, p \in [0,1]$$

Mechanism Design

Master protocol to choose $p_{\mathcal{A}}$

- Free rationals (master does not rely on rational workers)
  - Case 1: probability of malicious workers p_μ very large, high $p_{\mathcal{A}}$
    $p_{\mathcal{A}} \leftarrow 1 - \varepsilon / \mathbf{P}_{p_\mu + p_\rho}^{(n)}(\lceil n/2 \rceil, n)$
  - Case 2: probability of altruistic workers p_α big
    $p_{\mathcal{A}} \leftarrow 0$
  - Case 3: rationals probability of being honest $p_{\mathcal{H}}$ is 1, even if $p_{\mathcal{A}} = 0$
    $p_{\mathcal{A}} \leftarrow 0$
- Guided rationals (enforce the behavior of rational workers $p_{\mathcal{C}} = 0$)

$$p_{\mathcal{A}} \leftarrow \begin{cases} 1 - \frac{WP_{\mathcal{C}} + WB_{\mathcal{Y}} - WC_{\mathcal{T}}}{WP_{\mathcal{C}} + WB_{\mathcal{Y}}(\mathbf{P}_{p_\mu+p_\rho}^{(n-1)}(\lfloor n/2 \rfloor, n-1) + \mathbf{P}_{p_\mu+p_\rho}^{(n-1)}(\lceil n/2 \rceil, n-1))} & \mathcal{R}_{\mathrm{m}} \\ \frac{WC_{\mathcal{T}}}{WP_{\mathcal{C}} + WB_{\mathcal{Y}}} + \psi, \text{ for any } \psi > 0 & \mathcal{R}_{\mathrm{a}} \,\&\, \mathcal{R}_{\emptyset} \\ 1 - \frac{WP_{\mathcal{C}} + WB_{\mathcal{Y}} - WC_{\mathcal{T}}}{(WP_{\mathcal{C}} + WB_{\mathcal{Y}})(\mathbf{P}_{p_\mu+p_\rho}^{(n-1)}(\lfloor n/2 \rfloor, n-1) + \mathbf{P}_{p_\mu+p_\rho}^{(n-1)}(\lceil n/2 \rceil, n-1))} & \mathcal{R}_{\pm} \end{cases}$$

- if $U_M(p_{\mathcal{A}}, q) < U_M(1-\varepsilon, p_\mu + p_\rho)$ then $p_{\mathcal{A}} \leftarrow 1 - \varepsilon$

Optimality

Theorem: In order to achieve $P_{wrong} \leq \varepsilon$, the only feasible approaches are either to enforce a NE where $p_C = 0$ or to choose $p_{\mathcal{A}}$ so that $P_{wrong} \leq \varepsilon$ even if all rational workers cheat.

Computational Issues

- The mechanism for the master to choose appropriate values of $p_{\mathcal{A}}$ involves
  - Simple arithmetic calculations
  - Computing binomial probabilities
  - Verification of conditions for NE

  All these computations can be carried out using well-known numerical tools of polynomial cost.

- Together with the task, the master sends a ***certificate*** (p_A, *payoffs, n, reward*) of the uniqueness of the desired NE to the workers

PUTTING THE MECHANISM INTO ACTION

Volunteering Computing (SETI-like)

- Each worker
  - Incurs in no cost to perform the task: $WC_{\mathcal{T}} = 0$
  - Obtains a benefit: $WB_{\mathcal{Y}} > 0$
    (recognition, prestige – top contributors list)
- Master
  - Incurs in a (possibly small) cost to reward a worker (advertise participation): $MC_{\mathcal{Y}} > 0$
  - May audit results at a cost: $MC_{\mathcal{A}} > 0$
  - Obtains a benefit for correct result: $MB_{\mathcal{R}} > MC_{\mathcal{Y}}$
  - Suffers a cost for wrong result: $MP_{\mathcal{W}} > MC_{\mathcal{A}}$

Mechanism Instantiation

Instantiating the mechanism designed on these conditions the master can choose $p_{\mathcal{A}}$ and n so that U_M is maximized for $P_{wrong} \leq \varepsilon$ for any given worker-type distribution, reward model, and set of payoff parameters in the SETI scenario.

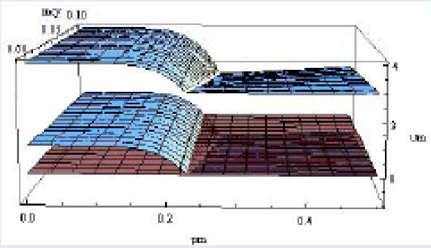
$$U_M \approx \max\{MB_{\mathcal{R}} - MC_{\mathcal{A}} - n(1-p_\mu)MC_{\mathcal{Y}},$$
$$MB_{\mathcal{R}}\mathbf{P}_{p_\mu}^{(n)}(0, \lfloor n/2 \rfloor) - MP_{\mathcal{W}}\mathbf{P}_{p_\mu}^{(n)}(\lceil n/2 \rceil, n) + \gamma\}$$

$$\gamma = \begin{cases} -MC_{\mathcal{Y}}(\mathbf{E}_{1-p_\mu}^{(n)}(\lceil n/2 \rceil, n) + \mathbf{E}_{p_\mu}^{(n)}(\lceil n/2 \rceil, n)) & \text{for the } \mathcal{R}_m \text{ and } \mathcal{R}_\pm \text{ models.} \\ -nMC_{\mathcal{Y}} & \text{for the } \mathcal{R}_a \text{ model.} \\ 0 & \text{for the } \mathcal{R}_\emptyset \text{ model.} \end{cases}$$

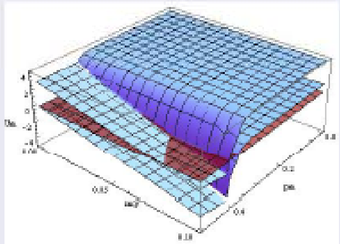
Plots

- Plots illustrating trade-off between reliability and cost
- Parameters' value:
  - $MC_{\mathcal{A}} = 1$, normalizing parameter
  - $MP_{\mathcal{W}} = 100$
  - Different values, don't change qualitatively the results
- 3D plots : Graphical characterization of the master's utility
  - $p_\mu \in [0, 0.5]$ ($p_\mu < 0.1$ in empirical evaluations on SETI-like system, Einstein@home, Estrada, Taufer and Anderson 09.)
  - $MC_{\mathcal{Y}} \in [0, 0.1]$, small maintenance cost of contribution list

Examples



- $\mathcal{R}_\emptyset$, $n = 15$
- Upper plane $MB_\mathcal{R} = 4$, lower plane $MB_\mathcal{R} = 1$, red plane $U_M = 0$
- Master audits around $p_\mu = 0.2$



- $\mathcal{R}_\emptyset$, $n = 75$
- Upper plane $MB_\mathcal{R} = 4$, lower plane $MB_\mathcal{R} = 1$, red plane $U_M = 0$
- Master audits around $p_\mu = 0.4$

Contractor Scenario

- Each worker
  - Incurs in a cost for computing: $WC_\mathcal{T} > 0$
  - Receives payment for computing the task (not volunteers): $WB_\mathcal{Y} = MC_\mathcal{Y} > 0$
  - Must have economic incentive: $U_i > 0$
- Master
  - Pays each worker an amount: $MC_\mathcal{Y} > 0$
  - Receives a benet (from consumers for the provided service): $MB_\mathcal{R} > MC_\mathcal{Y}$
  - May audit and has a cost for wrong result:

$$MP_\mathcal{W} > MC_\mathcal{A} > 0$$

Mechanism Instantiation

Instantiating the mechanism designed on these conditions the master can choose $p_\mathcal{A}$ and n so that U_M is maximized for $P_{wrong} \le \varepsilon$ for any given worker-type distribution, reward model, and set of payoff parameters in the Contractor scenario.

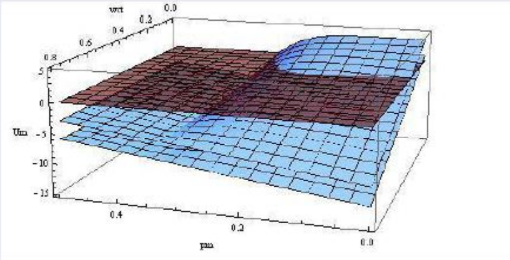
$$U_M \approx \max \Big\{ MB_\mathcal{R} - MC_\mathcal{A} - n(1-p_\mu)S,$$
$$\frac{WC_\mathcal{T}}{S}(MB_\mathcal{R} - MC_\mathcal{A} - n(1-p_\mu)S)$$
$$+ \left(1 - \frac{WC_\mathcal{T}}{S}\right)\left(MB_\mathcal{R}\mathbf{P}^{(n)}_{p_\mu}(0, \lfloor n/2 \rfloor) - MP_\mathcal{W}\mathbf{P}^{(n)}_{p_\mu}(\lceil n/2 \rceil, n)\right) \Big\}$$

(for $\mathcal{R}_\emptyset$ reward model)

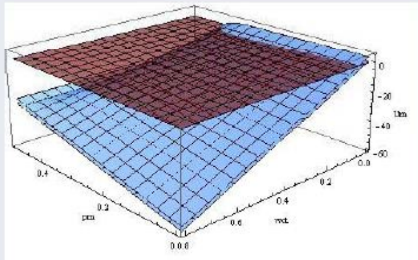
Plots

- Parameters' value:
  - $MC_\mathcal{A} = 1$, normalizing parameter
  - $MP_\mathcal{W} = 100$
  - $S = 0.8$
  - Different values, don't change qualitatively the results
- 3D plots : Graphical characterization of the master's utility
  - $p_\mu \in [0, 0.5]$ ($p_\mu < 0.1$ in empirical evaluations on SETI-like system, Einstein@home, Estrada, Taufer and Anderson 09.)
  - $WC_\mathcal{T} \in [0, S]$

Examples



- $\mathcal{R}_\emptyset$, $n = 15$
- Upper plane $MB_\mathcal{R} = 4$, lower plane $MB_\mathcal{R} = 1$, red plane $U_M = 0$
- Master audits around $p_\mu = 0.35$



- $\mathcal{R}_\emptyset$, $n = 75$
- Upper plane $MB_\mathcal{R} = 4$, lower plane $MB_\mathcal{R} = 1$, red plane $U_M = 0$
- Master audits around $p_\mu = 0.48$

CONCLUSIONS

Summary

- Combined
  - Classical distributed computing approach WITH
  - Game-theoretic approach

  towards reliable Master-Worker Internet-based Task computing under
  - Malicious, altruistic and rational workers
  - Communication uncertainty and worker unavailability
- Mechanisms trade reliability (ε) and cost (U_M) (and network unreliability)

Added Value

- As an example: instantiation of such mechanism in two real-world scenarios

BOINC-based systems (such as SETI@home) send the same task to three (3) workers. Our analysis identifies rigorously, for any given system parameters, the best allocation that BOINC-based systems could deploy.

The analysis on the contractor scenario opens the way for commercial Internet-based supercomputing where a company, given specific system parameters, could calculate its profit (if any) before agreeing into providing a proposed computational service.

Ongoing and Future Work

- Consider task execution over multiple rounds over workers that their behavior changes over time
  - View the computations in the Master-Worker framework as Evolutionary Games
- Reinforcement learning
  - The Master uses knowledge gained in past rounds to
    - decrease of its probability of error in future rounds
    - increase its utility in future rounds
  - The workers use prior knowledge to increase their utility
- Worker reputation
  - Measure the workers' reputation based on prior behavior and use it as an additional incentive for rational workers to act correctly.

Presentation available at:

http://www.cs.ucy.ac.cy/ric/dissemination.html

For further questions:

chryssis@cs.ucy.ac.cy