Towards Soft Real-Time Applications on Enterprise Desktop Grids

Derrick Kondo, Bruno Kindarji, Gilles Fedak, Franck Cappello

Laboratoire de Recherche en Informatique/INRIA Futurs

Desktop Grids

- Use free compute, storage and network resources in Internet and Intranet environments
- Why use desktop grids?
  - Huge computational power at relatively low cost
    - BOINC (largest Internet desktop grid system) currently provides over 400 TeraFlop/sec from 1 million hosts!
- Used by over 40 applications from wide range of scientific domains

Motivation and Goal

- State of the art: high-throughput, task-parallel applications
- Broaden applicability of desktop grids
- Allow soft real-time applications to utilize desktop grids
  - Online tomography, sensor networks, genome sequence assembly

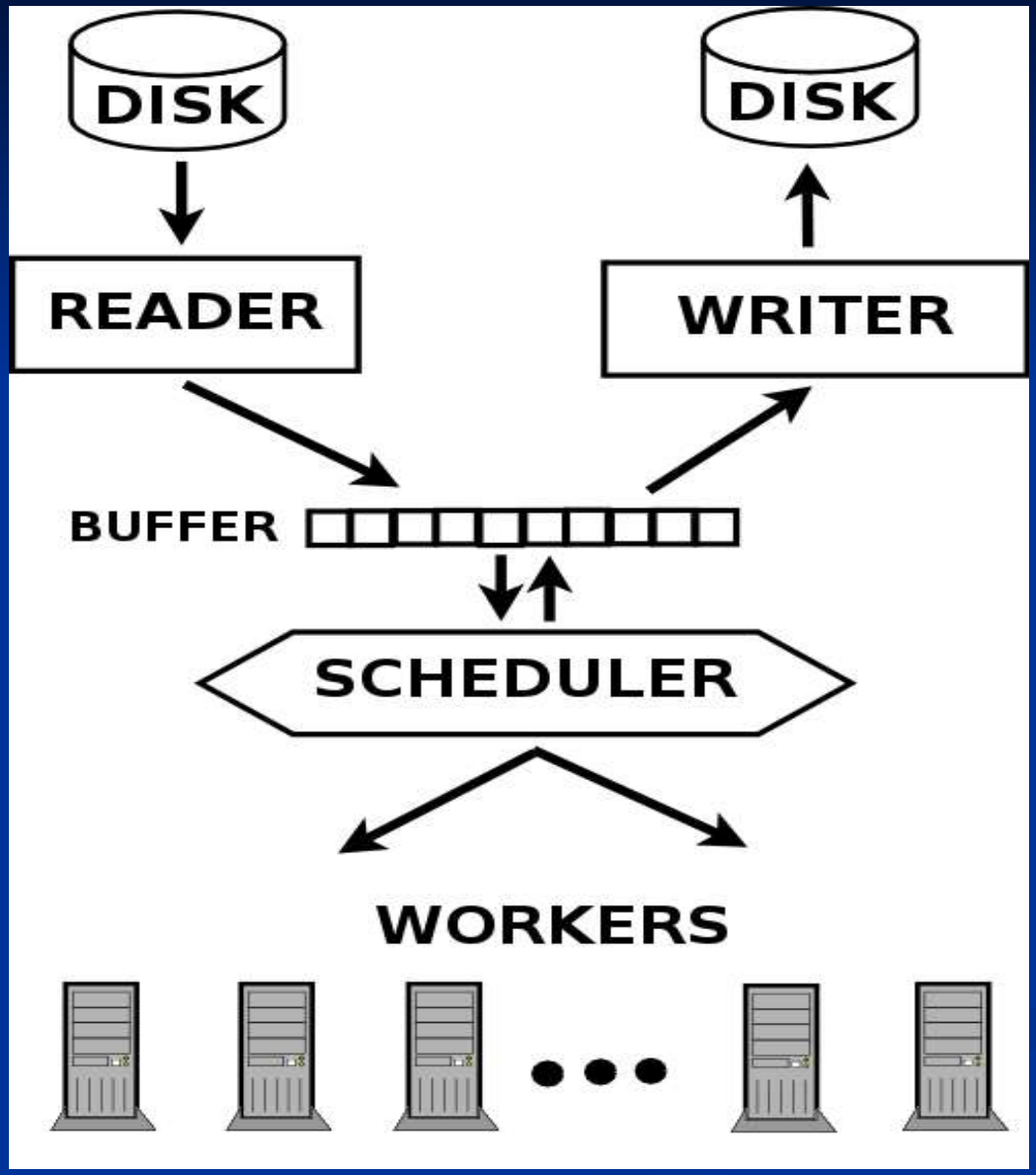
Challenge

- Volatility
  - Resources are shared with user or owner
    - Keyboard or mouse activity
    - Other user processes
    - Causes high rate of nondeterministic failures
- Heterogeneity
  - CPU, memory sizes, network

Approach

- Meet soft deadlines for task completion via server-side buffering technique
  - Characterize the factors that effect task completion
  - Construct model from characterization
  - Show evidence that supports model via trace-driven simulation

Problem Statement



H: tasks per batch

C_{in}: period that task batch is added to buffer

b: buffer size

d: task deadline

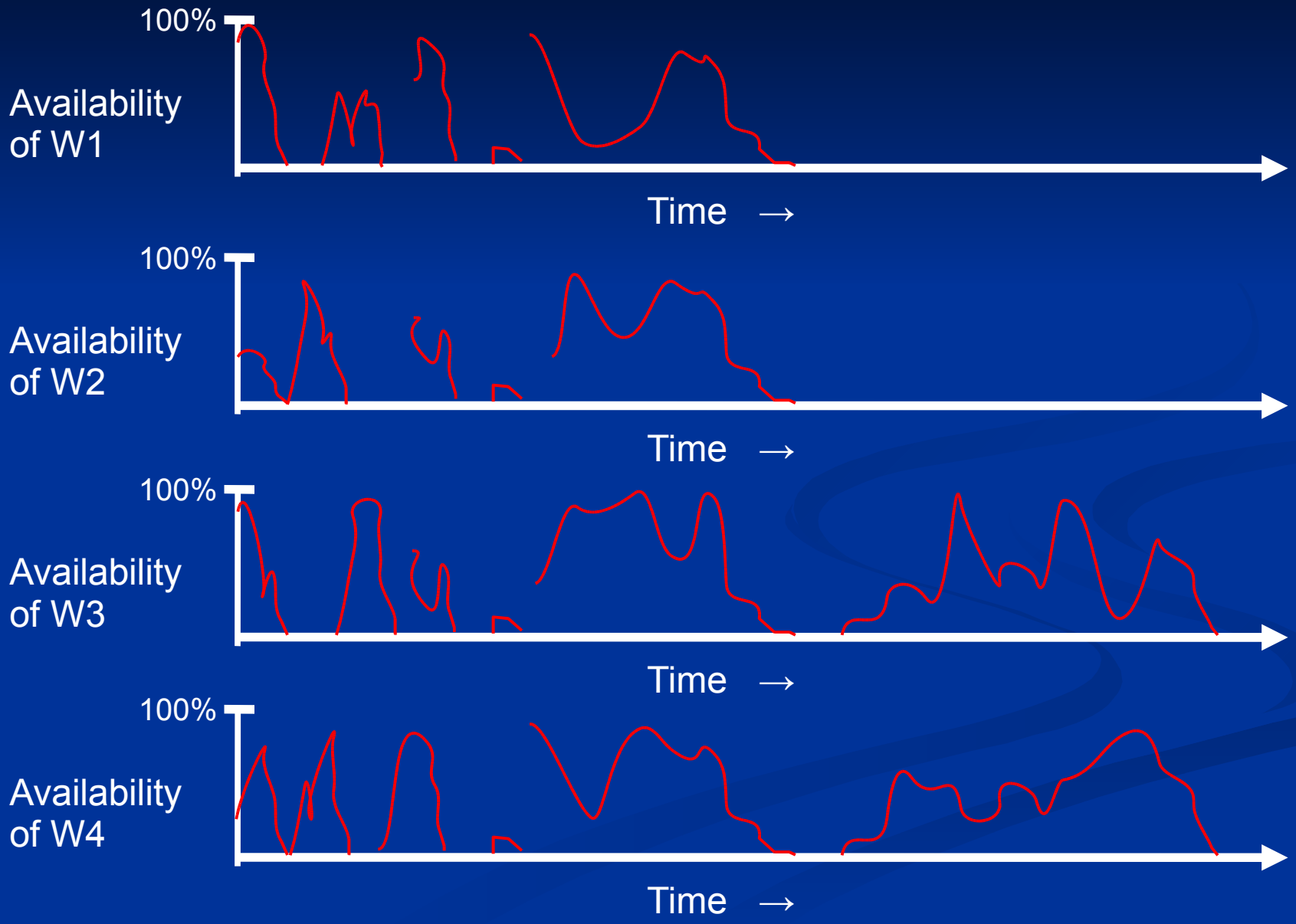
$d = b*(C_{in}/H)$

Performance metric: task success rate

Assumptions

- Tasks are independent and compute-intensive
  - Amount of input or output data is small
- Cannot cancel tasks once scheduled
  - Firewalls, NAT's, intermittant network connectivity make this difficult
- If a task fails, worker notifies scheduler as soon as possible

Characterization



Characeterization Of Aggregate Compute Power

$p_i^{t,t+\delta}$: Compute power during [t and $t+\delta$] for worker i

$$P^{t,t+\delta} : \sum_{i=1}^{N} p_i^{t,t+\delta} \text{where } N \text{ is the \# of workers}$$

- H_0: Aggregate compute power $P^{t,t+\delta}$ follows a normal distribution
- Central Limit Theorem: sum of a set of variates from any distribution with finite mean and variance tends toward a normal distribution

Trace Data

- San Diego Supercomputer Center (SDSC)
  - Submitted compute-intensive application to real desktop grid with 200 hosts over a 1-month period
  - Logged number of operations completed per 10 second period to file
  - Assembled files to yield availability trace
- UC Berkeley (UCB) [Arpaci95]
  - Logged keyboard/mouse activity and processes for about 85 hosts over a 2-week period

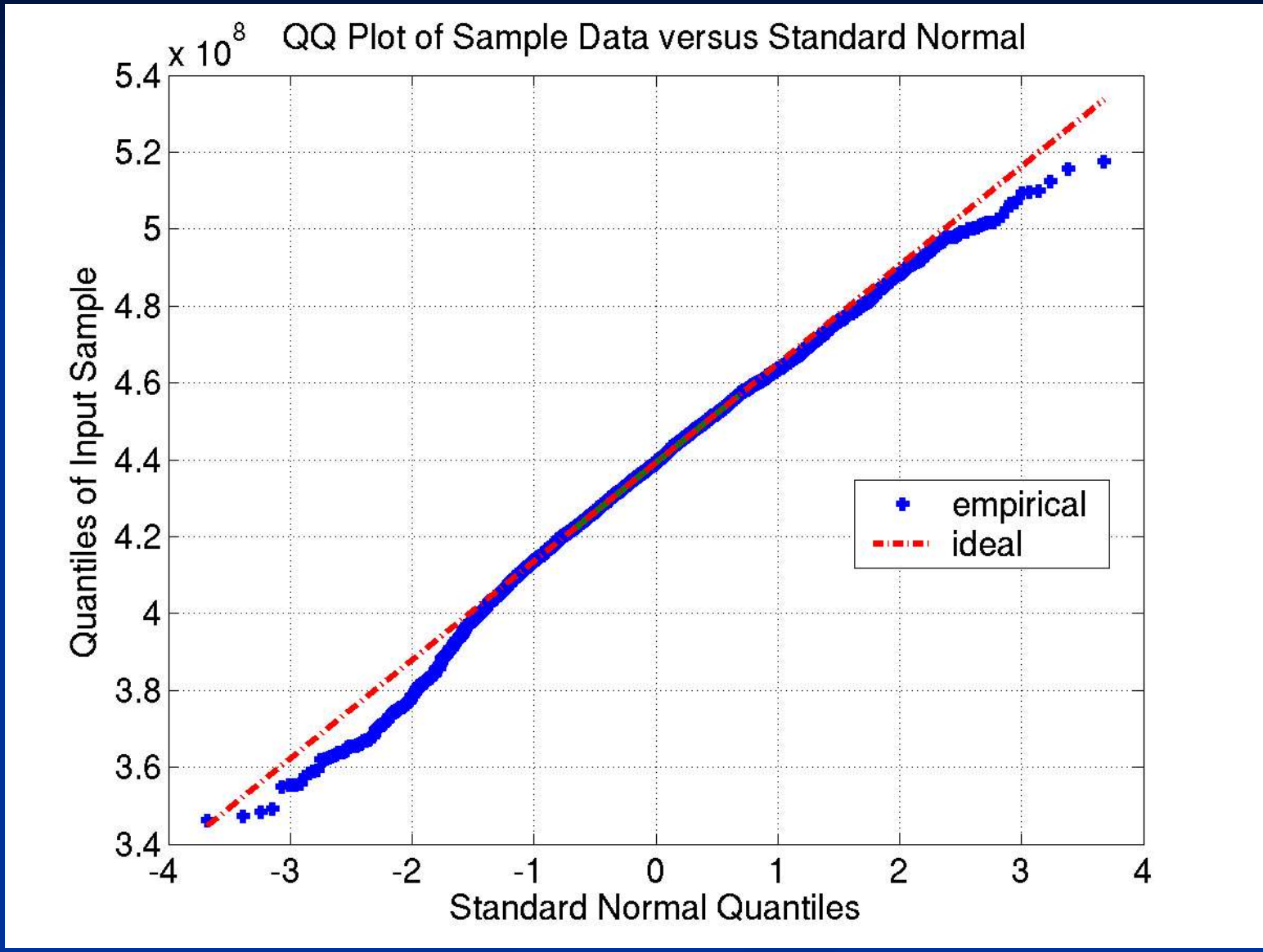
Parameter Estimation

$p_i^{t,t+\delta}$: Compute power during [t and $t+\delta$] for worker i

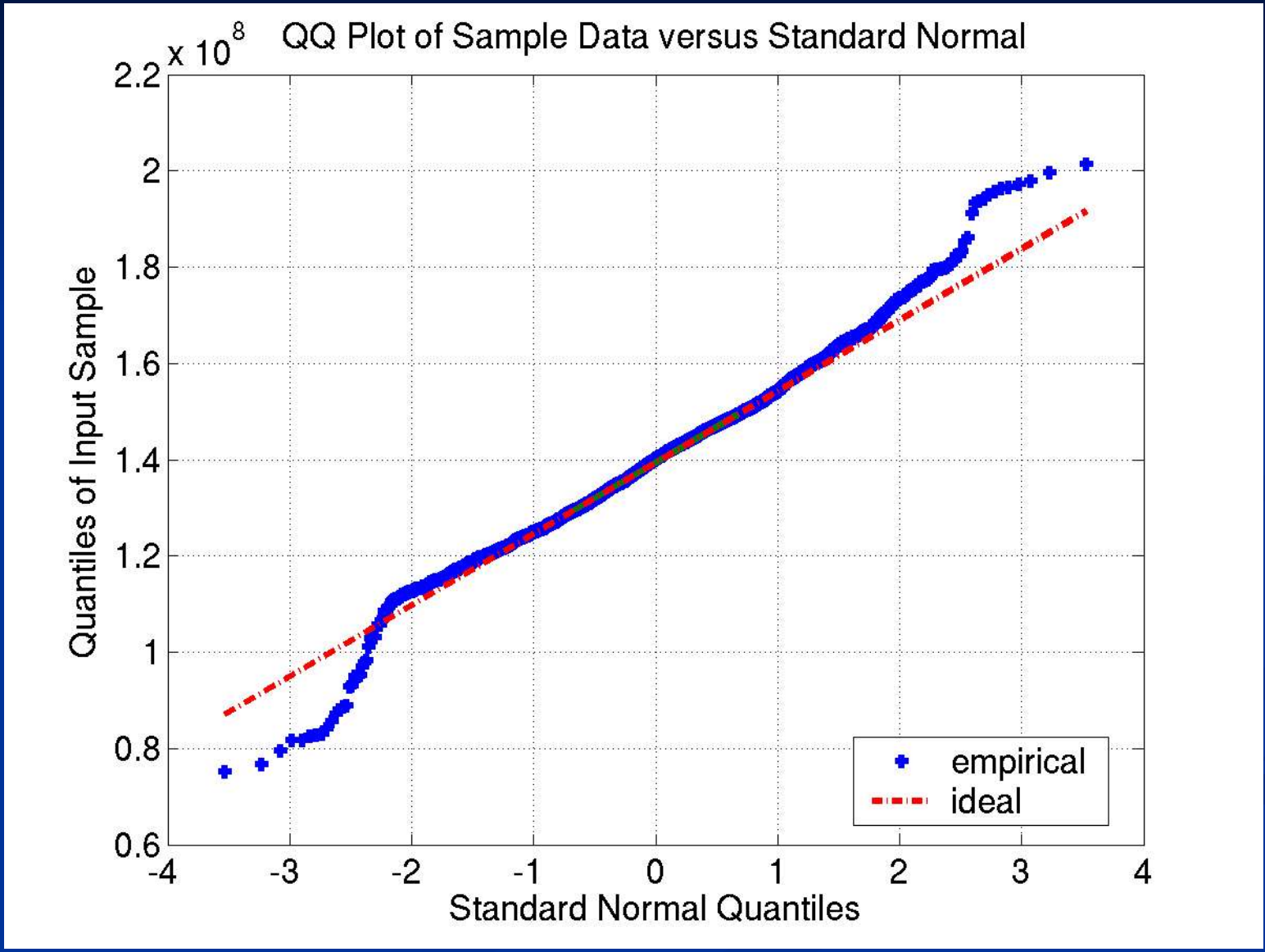
$$P^{t,t+\delta} : \sum_{i=1}^{N} p_i^{t,t+\delta} \text{ where } N \text{ is the \# of workers}$$

- Measured $P^{t,t+\delta}$ at thousands of different values of t with δ=60 sec.
- Maximum likelihood estimation to determine parameter fit
  - SDSC: $P^{t,t+\delta}$ ~ N ($1.4*10^8$, $1.56*10^7$)
  - UCB: $P^{t,t+\delta}$ ~ N ($4.8*10^8$, $2.6*10^7$)

UCB Quantile-Quantile Plot



SDSC Quantile-Quantile Plot



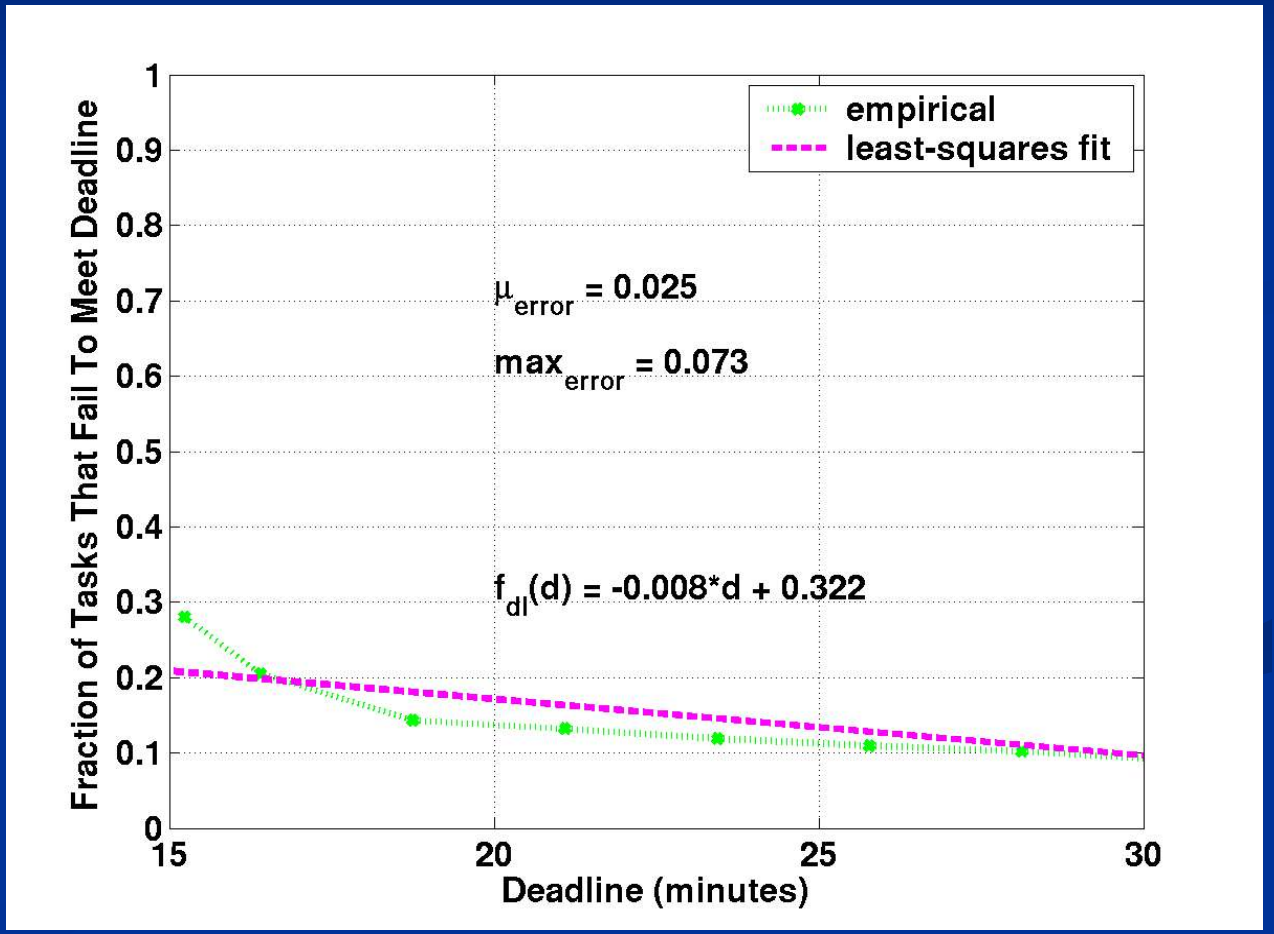
Kolmogorov-Smirnov Goodness-of-fit Test

- Quantitative test
  - Intuition: reflects maximum difference between observed and expected cumulative distribution function (CDF)
- UCB: mean p-value: 0.466
- SDSC: mean p-value: 0.448

Characterization of Deadline Failure Rates

$f_{dl}(d)$: Fraction of tasks that fail to meet deadline d

- Determine $f_{dl}(d)$ using random incidence for 15-minute tasks for UCB



Failure Rate as a Function of Buffer Size

$$f_{dl}(d) = -0.008 * d + 0.322$$

Since $d = f(b) = (C_{in}/H) * b$,

$$f_{dl}(f(b)) = (-0.008 * C_{in} * b)/H + 0.322$$

Aggregate Compute Power as a Function of Buffer Size

$$b \geq ((H * s)/(C_{in} * P^{t,t+\delta})) * H$$

where s is the size of the task in ops

$$\Longleftrightarrow$$

$$b \geq (H^2 * s)/(C_{in} * P^{t,t+\delta})$$

$$Pr(P^{t,t+\delta} \geq \alpha) = Pr(P^{t,t+\delta} \geq (H^2 * s)/(C_{in} * b))$$

Modelling Task Success Rate

$$Pr(P^{t,t+\delta} \geq \alpha) = Pr(P^{t,t+\delta} \geq (H^2 * s)/(C_{in} * b))$$

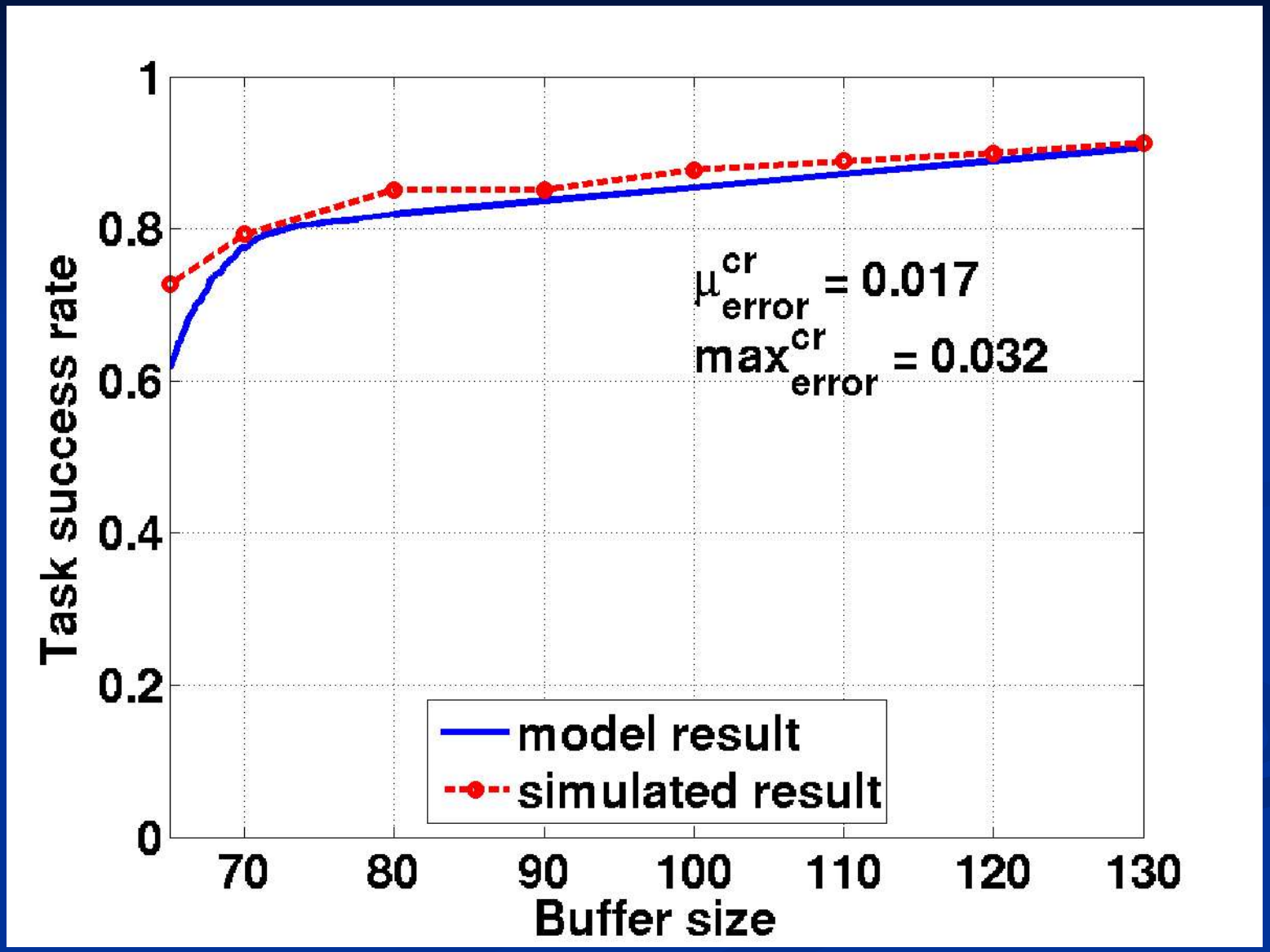
$$f_{dl}(f(b)) = (-0.008 * C_{in} * b)/H + 0.322$$

$$S(b) \geq Pr(P^{t,t+\delta} > \alpha)(1 - f_{dl}(f(b))$$

$$\Longleftrightarrow$$

$$S(b) \geq Pr(P^{t,t+\delta} > (H^2 * s)/(C_{in} * b))(1 - ((-0.008 * C_{in} * b)/H + 0.322))$$

15m-Task Success Rate Versus Buffer Size on UCB



Related Work

- Characterization
  - Passive measurement techniques for measuring CPU or host availability [Wolski, Dinda]
    - Ignore effects of keyboard and mouse activity
    - Susceptible to OS idiosyncracies
    - Correspond to a different definition of availability
- Soft real-time scheduling in shared, undedicated environments [Dinda]
  - Use different definition of availability (do not consider task failures)

Summary

- Characterized desktop grids
  - Showed that aggegate compute power can be modelled as a normal distribution
  - Showed that deadline task failure rate can be modelled as a linear function
- Created closed-form model of task success rate as a function of buffer size
- Compared accuracy with simulated results

Current and Future Work

- Characterization of *Internet* desktop grids: XtremLab (http://xtremlab.lri.fr)
- Approach
  - Create BOINC-based project to gather availability measurements
  - Generative and predictive aggregate and per host models
- Status
  - Running since March, 2006
  - Over 7,000 participating hosts

Current and Future Work Continued

- Enable soft real-time applications in BOINC with emphasis on scheduling
- Approach
  - Provide simulation framework by which to test, evaluate, and compare novel desktop grid schedulers: SimBOINC
    - Based on SimGrid [Legrand]
    - Graft BOINC scheduling code
    - Will utilize models from XtremLab
- Status: alpha version available
  - Will be used to test BOINC scheduler itself

Current and Future Work Continued

- Enable soft real-time applications in BOINC with emphasis on scheduling
- Approach
  - Provide simulation framework by which to test, evaluate, and compare novel desktop grid schedulers: SimBOINC
    - Based on SimGrid [Legrand]
    - Graft BOINC scheduling code
    - Will utilize models from XtremLab
- Status: alpha version available
  - Will be used to test BOINC scheduler itself

Ensuring Worker Request Rates for Soft Real-Time Applications

- Want batch of M tasks to complete with time T
- Each tasks takes Y time units
- Maximum time to send out M tasks is $W = T - Y$
- Let R be the number of requests that arrive within period W
- Find the average host request period z_h so that $R \geq M$ requests arrive in the period of length W with 95% probability

15m-Task Success Rate Versus Buffer Size on UCB