Designing and Assessing Adaptive Dependable Distributed Systems: Putting the Model in the Loop

William H. Sanders

University of Illinois at Urbana-Champaign
whs@crhc.uiuc.edu
www.crhc.uiuc.edu/PERFORM

IFIP Working Group 10.4 on Dependable Computing,
St. John, USVI, January 5, 2002

Motivation

- Modern dependable distributed systems are often intended to serve multiple users, whose (quality of service) needs are not known at design time, and may change over the course of a period of use
- Such systems are inherently complex, and cannot be designed to meet a changing specification using traditional ad hoc means
- Modeling methods and tools that can be applied to practical dependable distributed systems are needed
- Changing quality-of-service needs also suggest that systems should be created that can adapt at run time
- On-line models can effectively guide this adaptation
- Simple models have been used in the past, but advances in modeling technology and increased computation power suggests that more sophisticated models, inspired by those developed by the traditional modeling community, could be used

Stochastic Modeling Technology: State-of-the-Art

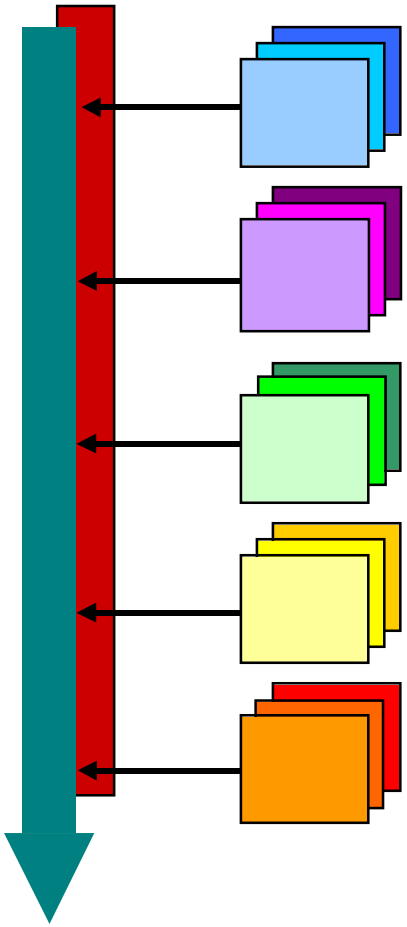
- Many model representation methods, BUT either limited in representation power, or not understandable to design engineers
  - Fault-Trees, Reliability Block Diagrams
  - Stochastic Petri nets and variants
  - Stochastic Process Algebras
- Model solution methods quite advanced
  - Simulation of dependability of sophisticated systems (non-exponential failure and repair times, complex repair policies, complex component and SW interaction) of 5- and 6-nines possible in reasonable time
  - “Structured” representation methods make generation of many extremely large-spaces (30+ million states) possible on standard workstation (1 GB main memory)
  - Transient and and steady-state analytic/numerical solution methods also can handle large systems, but may be slow

Future Research Directions

- Domain-specific and domain-independent modeling languages that are natural to system designers
- Integration with existing hardware and software design tools
- Composition and connection methods that use the structure of models in their solution, and are exact, or that give an estimate of the error they induce through their use (need to be able to build models that be quickly composed and used in a variety of circumstances)
- Model solution methods that make use of the nature of specific performance/dependability variable specifications to reduce cost of a solution - for analytic/numerical methods time, not space, will be the bottleneck!
- Methods that present result in a manner that are more useful system designers

Möbius Modeling Framework

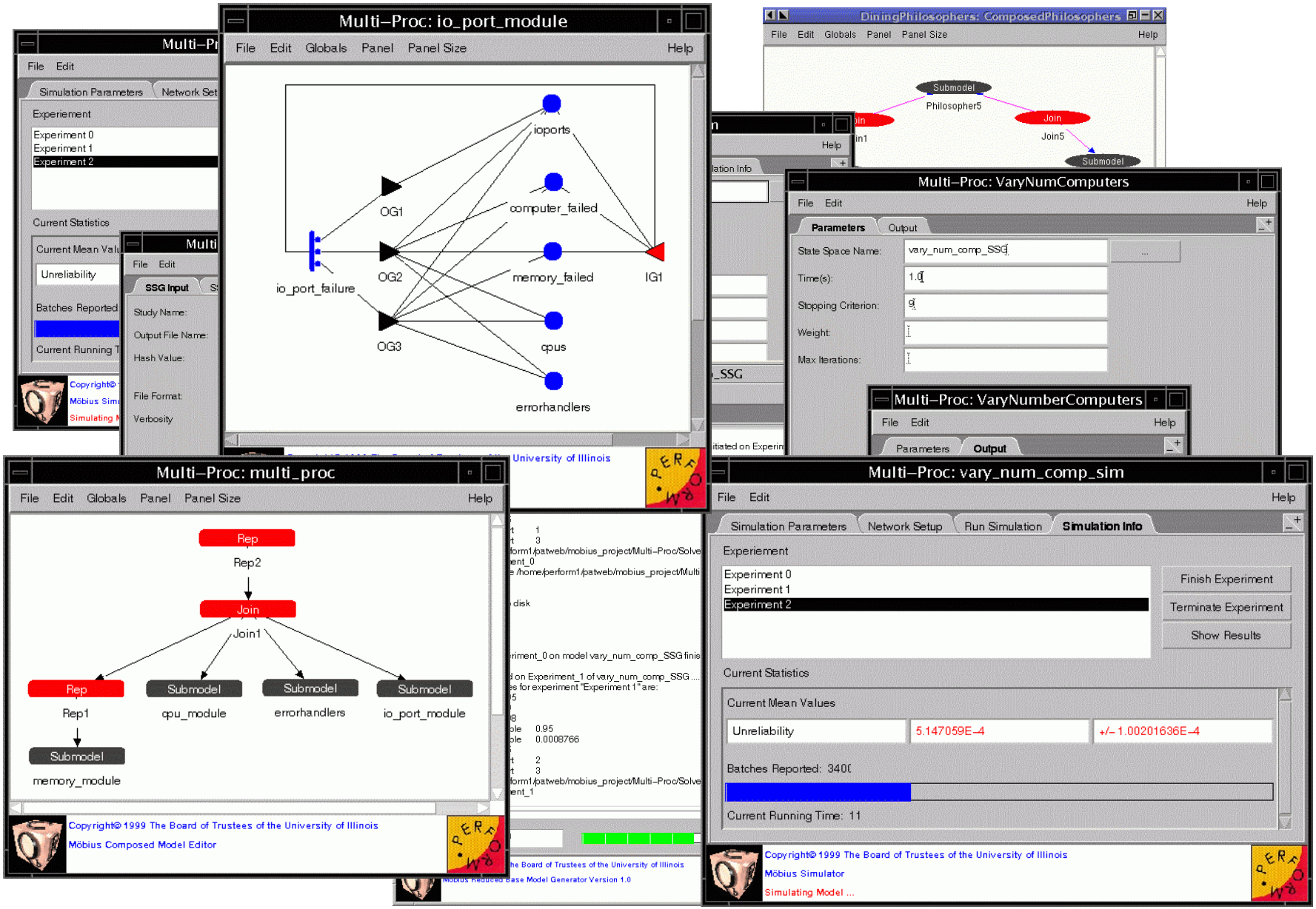
Submodel Interaction via
Abstract Functional Interface



Formalism Independent
Model Solution

Framework Component	Example Formalisms
Atomic Model	DSPN, GSPN, Markov chain, Queueing Network, SAN, SAN, SPA, other SPN extensions, Domain-specific formalism
Composed Model	Graph interconnection Kronecker Composition (SAN), Replicate/Join, SPA Domain-specific formalism
Solvable Model	Rate/Impulse reward variables Path-based reward variables Domain-specific formalism
Connected Model	Fixed-point governor Acyclic model composer
Study Specifier (generates multiple models)	Range and Set Variation Design-of-Experiments Iterator

Möbius Graphical User Interfaces

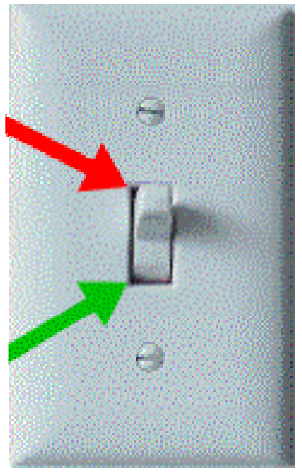


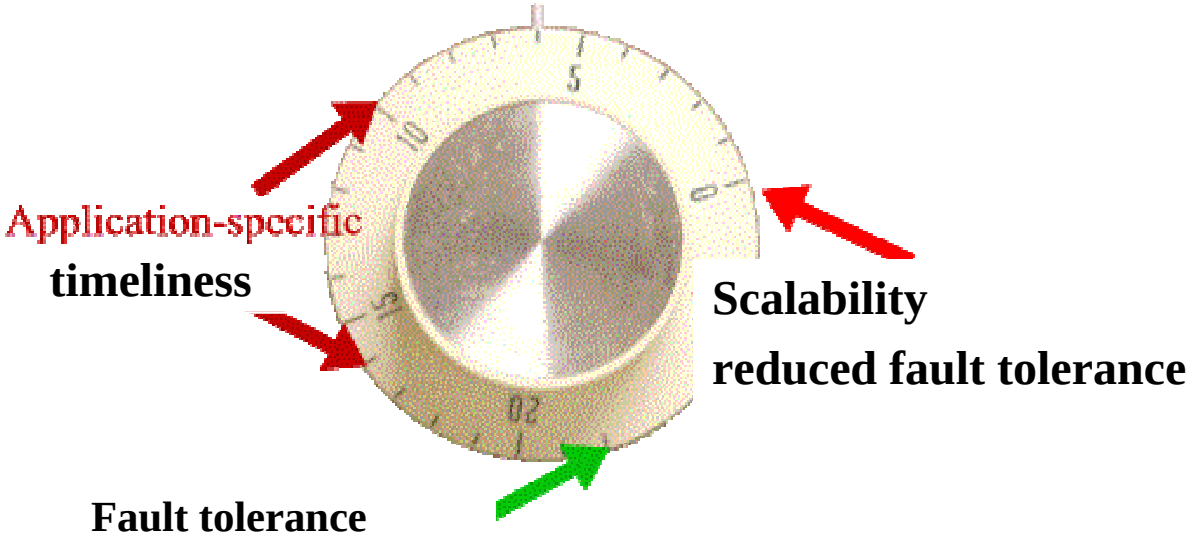
Model-based Adaptation in Dependable Distributed Systems

- Large separation between stochastic modeling and middleware design communities, the first being very formal, and the second being largely experimental, stressing the importance of prototyping and and demonstration
- Processor speeds, relative to communication times and required QoS, make on-line solution of models possible to make adaptation decisions
- Research is needed to:
  - Collect appropriate data for model input (what to collect, how to collect, how to statistically process)
  - Build appropriate models (all models are wrong, some are useful!)
  - Quickly solve models
  - Provide methods for multiple models present in a system to interact with one another, providing global adaptation policy without the overhead of centralized control or global state knowledge

An Initial Success: Replica Selection to Achieve Soft-Realtime and Dependability Constraints

- **Spatial redundancy = 1**
- **Selection metrics: [lowest response time, nearest replica, …]**
- **Tolerate unresponsiveness using temporal redundancy**
- **Spatial redundancy = N [Eternal, AQuA, …]**





- Better approach:
  Tunable redundancy

Probabilistic Model

- M : set of replicas offering a service
- Ri : response time of replica i (a random variable)
- t : response time requested by client
- K : replica subset selected to service a client
- $P_K(t)$: Prob (at least one replica in K responds by t)
- P (no timing failure) = $P_K(t)$

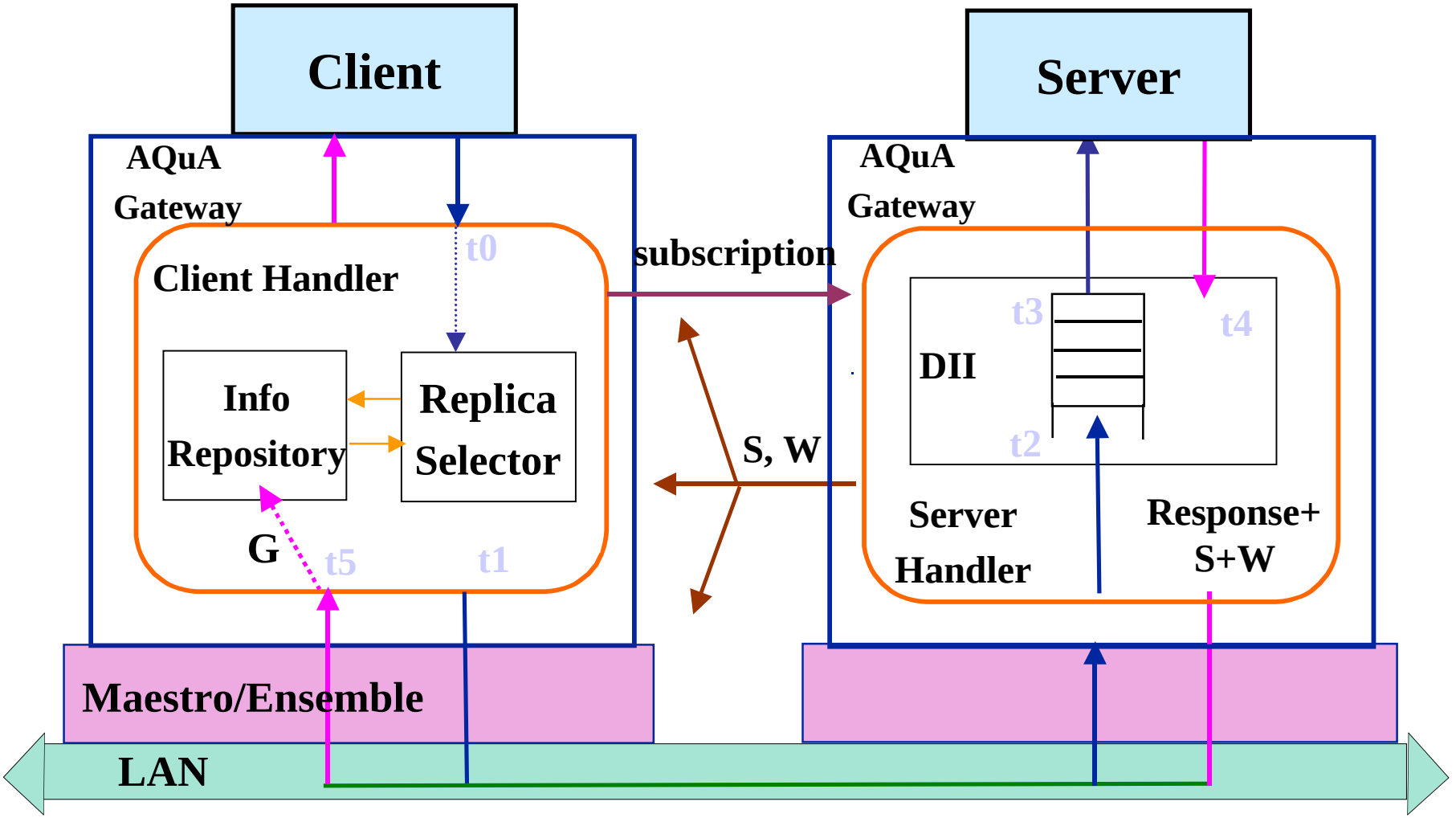
$\Rightarrow$1- P(no replica in K responds in time t)

$$\Rightarrow 1 - \prod_{i \in K} P(R_i > t)$$

$$\Rightarrow 1 - \prod_{i \in K} (1 - F_{R_i}(t))$$

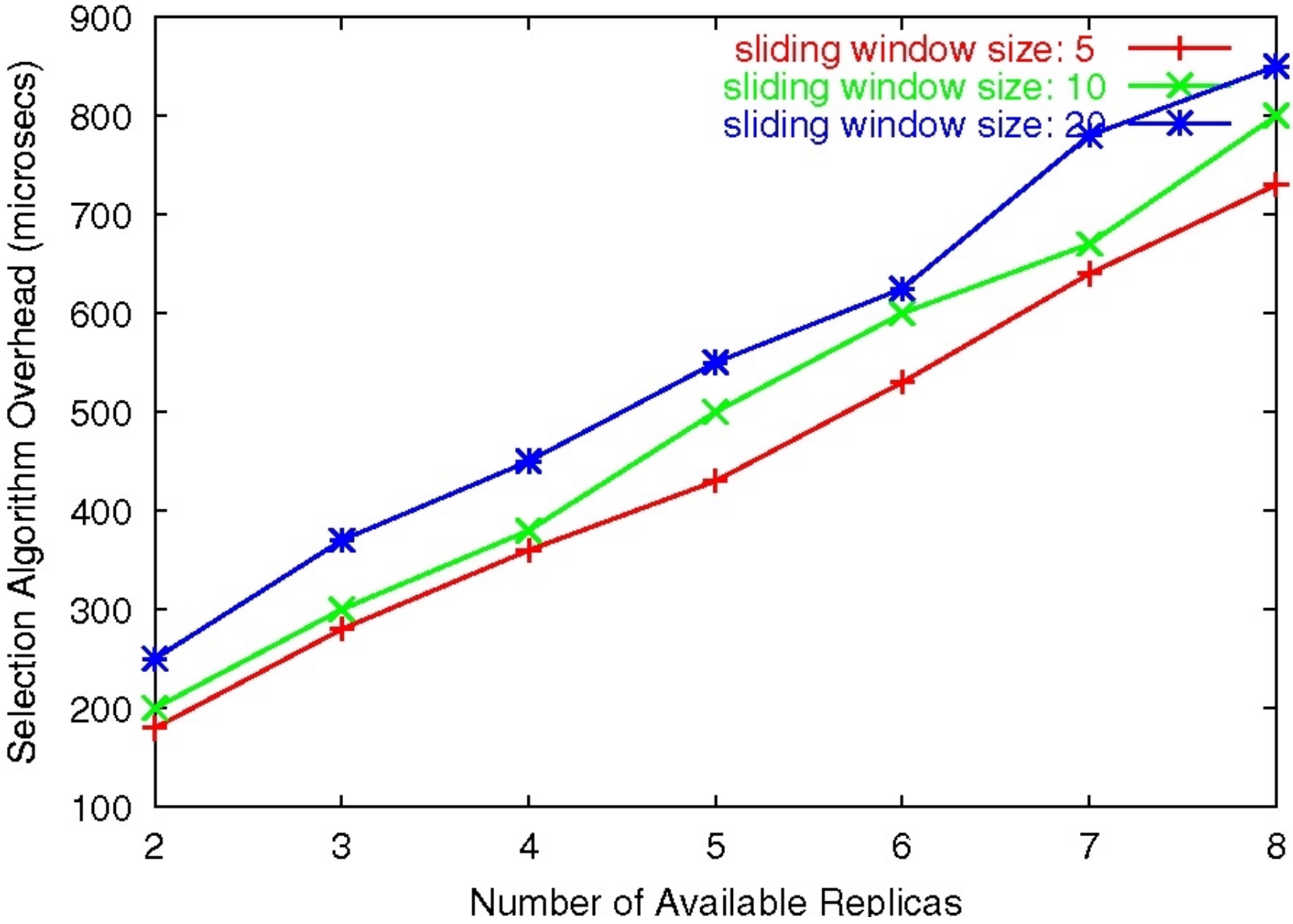
$\Rightarrow F_{R_i}$: response time distribution function of replica i

Timing Fault Handler

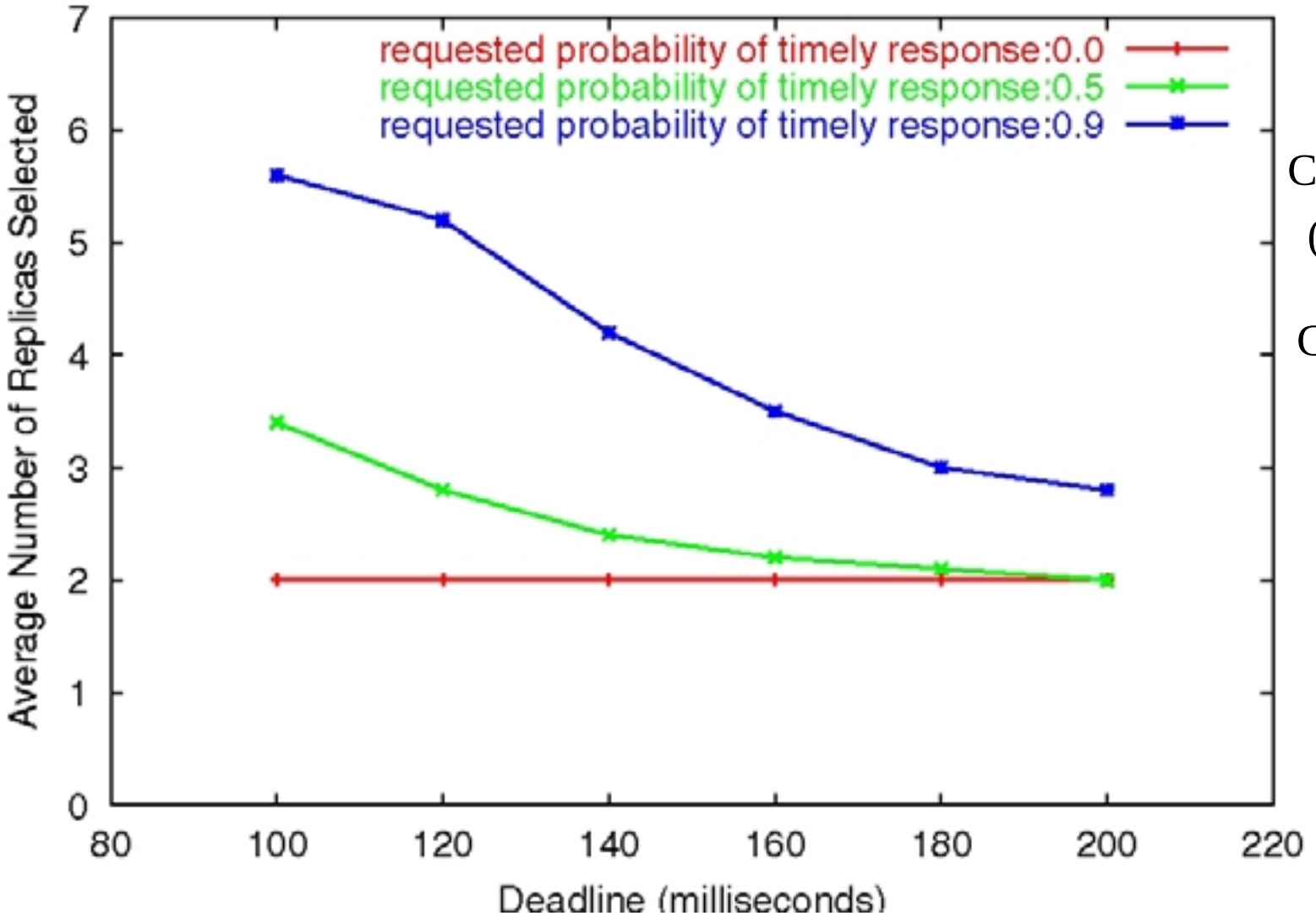


S = t4 - t3 W = t3 - t2 G = t5 –t1 – S –W Timing Failure = t5 - t0 > t

Overhead of Replica Selection Algorithm



Variation of Redundancy Level

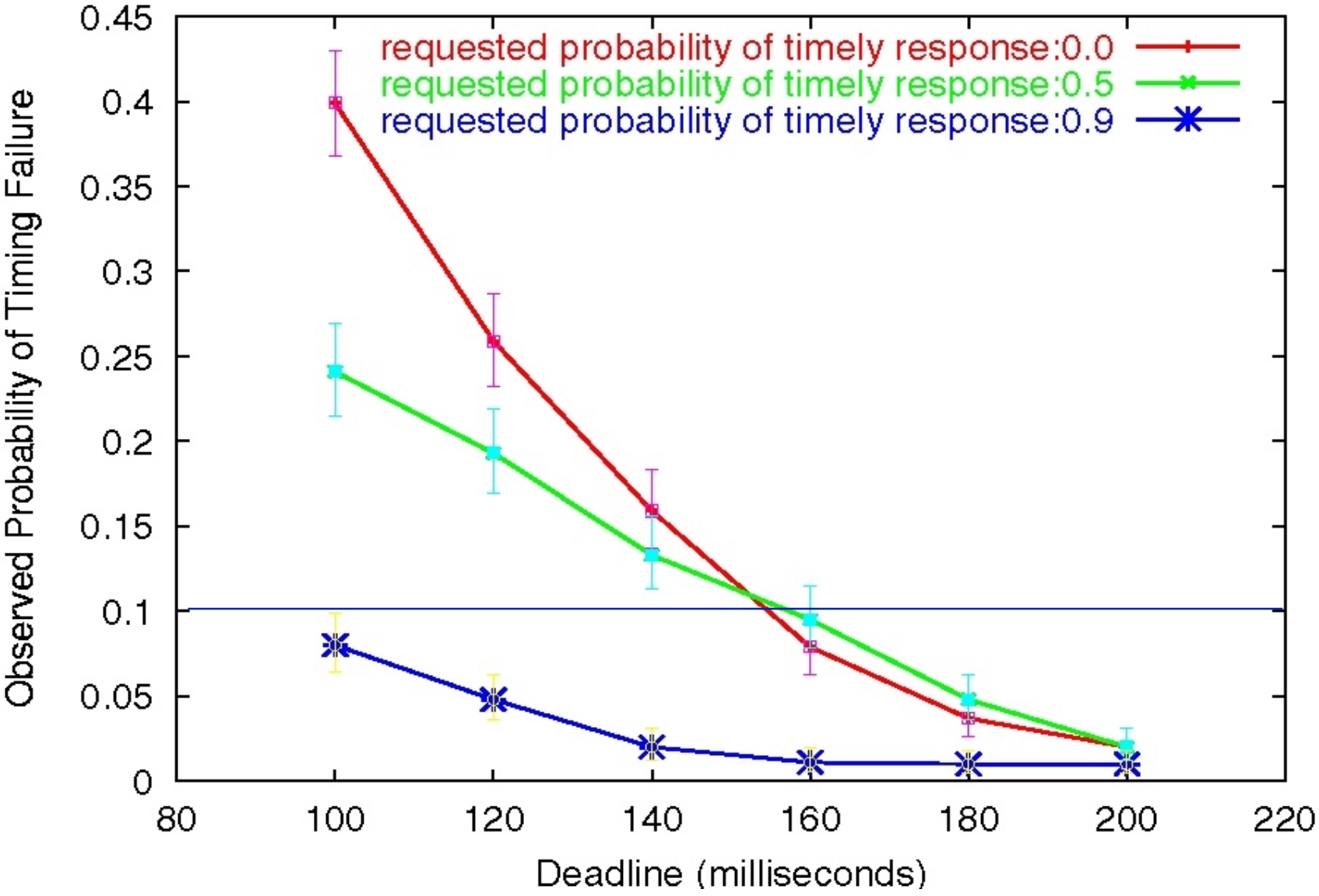


Client1 QoS
(200 ms,0)

Client2 QoS
(x, x)

•More stringent the QoS specification, the higher the redundancy

Experimental Validation of Model



Challenge Summary

- Much progress has been made in stochastic modeling theory, but LARGE gap remains between what can be done by an expert modeler, and can be done by typical system architect

⇒ Research is needed make modeling technology accessible to designers by 1) integrating modeling technology with design tools, 2) creating domain-specific modeling formalisms, 3) creating composition and connection techniques, and 4) finding ways to present results in a manner useful to designers

- Model-based adaptation is feasible, and has the potential to significantly improve quality of service provided to an application

⇒ Research is need to identify and create appropriate 1) measurement strategies, 2) models, 3) model interaction approaches, and 4) model solution methods