Adaptive Self-Optimizing Resource Management for the Grid

Chen-Khong Tham
eletck@nus.edu.sg

Gokul Poduval
gokul@nus.edu.sg

Computer Networks & Distributed Systems (CNDS) Laboratory
Dept of Electrical & Computer Engineering (ECE)
National University of Singapore

Motivation

- Jobs pass through several network domains to get to
  Grid resource (=computational server)
- Network routes and conditions change
  - Signing SLAs with every possible intermediate domain is cumbersome
  - Without SLAs, domain administrators will need to constantly keep track of resource allocations due to varying conditions to meet delay (QoS) req'mts
- Available resources at Grid resource change due to time-varying load and allocations

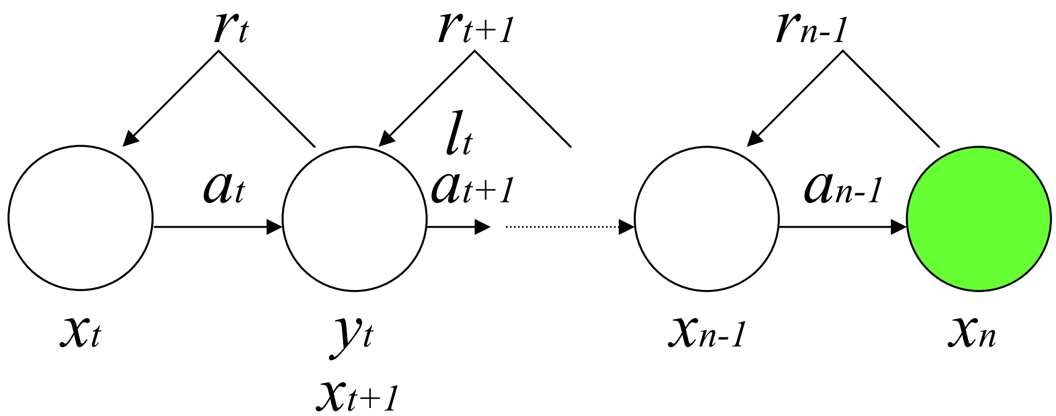
Advantages of Self-Optimization

- Real-time control of resource allocation can lead to better results in meeting deadlines (QoS)
- Fact: Each domain has different traffic characteristics
- Learning system will be able to adapt and respond to these different conditions

Our approach

- Treat computational and network resource allocation problems in Grid as Markov Decision Process (MDP) problems
- Using Reinforcement Learning (RL) techniques to learn resource allocation strategies without external supervision
- Key advantage over other learning systems:
  - able to cope well with uncertainty
  - external supervision, system model and configuration not required even during training phase

MDP, Q-Learning algorithm



$$Q_{t+1}(x,a) = \begin{cases} Q_t(x,a) + \eta_t[r_t + \gamma Q_t(y_t, l_t) - Q_t(x,a)] \\ \qquad\qquad \text{if } x = x_t \text{ and } a = a_t, \\ Q_t(x,a) \qquad \text{otherwise.} \end{cases}$$

Action selection

$$P(a|x_t) = \frac{e^{\beta Q_t(x_t,a)}}{\sum_{l \in A(x_t)} e^{\beta Q_t(x_t,l)}}$$

Our approach

- Allows users to select appropriate Grid resources to meet deadlines (QoS)
- Allows service providers to allocate appropriate amounts of CPU and network resources to different classes of Grid users
- Coordination among resource managers

RL techniques for Resource Mgmt

- We studied two RL techniques adapted to perform resource management on the Grid:
  - Watkins Q(λ)
  - Semi-Markov Average Reward Technique (SMART)

Watkins Q(λ)

- Off-policy Temporal Difference (TD) learning method
- Uses the following update rules

$$\delta \leftarrow R_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)$$

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \delta e(s_t, a_t)$$

SMART

- RL technique to solve Semi-Markov Decision Process (SMDP) problems
- SMDPs are MDP problems where the sojourn times or event intervals are not fixed but drawn from a general distribution
  - in Grid case, an event occurs when a job starts or ends

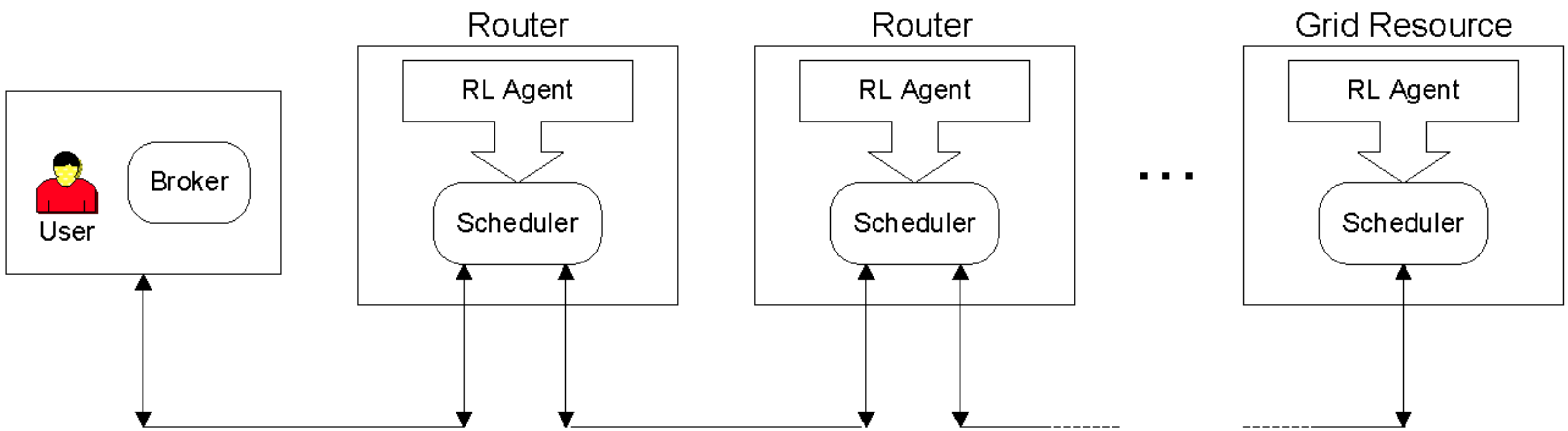
$$V_{new}(s,a) = (1-\alpha_m)V_{old}(s,a) + \alpha_m \left(R(s,s',a) - \mathcal{AR}_m \tau(s,s',a) + \max_{a'} V_{old}(s',a') \right)$$

$$\mathcal{AR}_m = (1-\beta_{m-1})\mathcal{AR}_{m-1} + \beta_{m-1}\frac{T(m-1)\mathcal{AR}_{m-1} + R(s,s',a)}{T(m)}$$

RL-based Resource Management

- Each user has a broker (UB), which decides the Grid resource (GR) where each job should be run
- The broker makes the decision using one of the RL techniques described earlier
- Network and computational nodes also run RL agents to decide the resource allocation levels

RL-based Resource Management



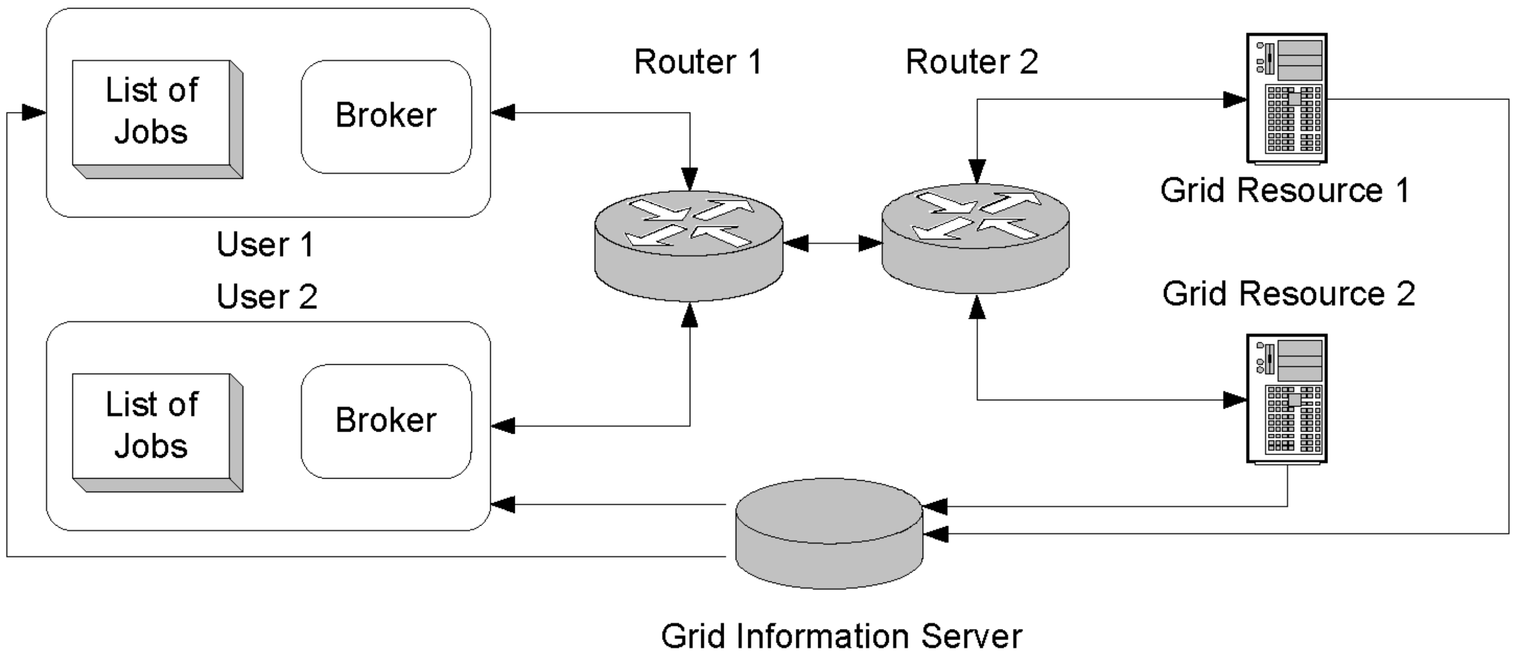
RL-based Resource Management

- Each Grid job has a deadline
- Brokers generate a positive reward if a job is completed and returned within the deadline, else a negative reward is generated
- This reward is assigned to itself and the agents at network and Grid resources for them to train themselves

RL-based Resource Management

- WFQ and Rate-Controlled Static Priority (RCSP) schedulers are used at routers to achieve provisioning and reservation respectively
- GPS and CPU reservation techniques are used at Grid nodes to achieve provisioning and reservation respectively

Simulation Setup



Simulation

- Enhanced GridSim grid simulation software was used
- *Unknown to RL agent:*
  - *All network links set to 1 Mbps, 5 ms propagation delay and MTU of 1 KB*
  - *Grid Resource 1 (GR1) and Grid Resource 2 (GR2) have processing capacities of 250 and 350 MIPS respectively*
- Simulations run for 25,000 s

Job Characteristics

Class	Job Size (MI)	Data Size (bytes)	Mean Generation Delay (s)	Deadline (s)
1	150	50,000	2.5	4
2	300	100,000	2.5	15

- Class 1 jobs model jobs requiring fast response time
- Class 2 jobs model jobs requiring large amount of processing, but not necessarily instant response (bulk jobs)
- User 1 always sends Class 1 jobs
- User 2 always sends Class 2 jobs
- → Fairly highly loaded
- Insight: complex relationship between desired QoS and amount of (different types of) resources to allocate at different load levels and usage patterns (unless guaranteed service/reservations/over-provisioning are used)

Simulation Experiments

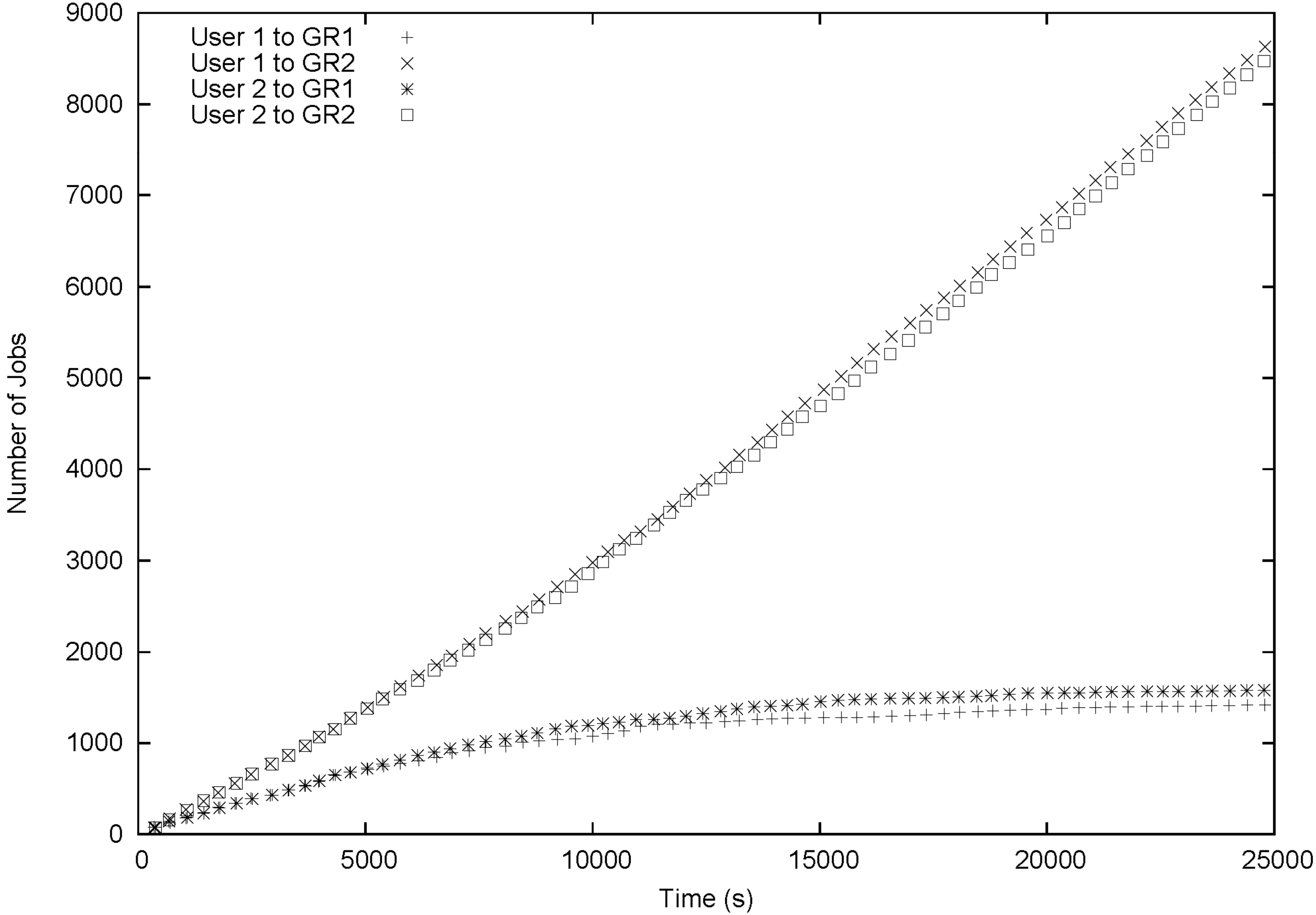
- Two schemes used for benchmarking Watkins Q(λ) and SMART:
  Round Robin (RR) and Exponential Averaging (EA)
- RR: Jobs are sent alternatively to GR1 and GR2
- EA: Jobs are sent to the resource with lower exponential average

$$EA_{it} = \alpha * \rho_{it} + (1-\alpha) * EA_{i(t-1)}$$

Results

Class	RR (s)	EA (s)	QL (s)	SMART (s)
1	2.68	2.61	2.06	2.02
2	5.63	21.28	4.65	4.42

- Table shows average response times (total time to submit, execute and get results)
- SMART performs better than QL, which performs better than RR and EA
- RR: resource managers are unable to select the appropriate resource allocation levels
- EA: there is a large delay in switching because of averaging; actions are taken too late
- Next slide shows job distribution for QL scheme


User 1 to GR1
User 1 to GR2
User 2 to GR1
User 2 to GR2
Number of Jobs
Time (s)
0
1000
2000
3000
4000
5000
6000
7000
8000
9000
0
5000
10000
15000
20000
25000

Results

- Previous slide showed the job distribution when user brokers use Q(λ)
- Initially, both GR1 and GR2 receive equal share of the load
- GR1 starts getting congested due to lower processing capability, a situation to which the RL algorithm adapts
- Subsequently, more jobs are sent to GR2

Conclusion

- RL techniques are able to dynamically perform resource allocation in response to changing traffic conditions
- Future Work
  - Next step is to extend the formulation to resource pricing situation
    - resource managers charge prices based on factors such as congestion level, revenue generation etc.
    - users select resources to meet QoS requirements and budget constraints
  - More advanced coordination schemes

Thank You – Questions?