A temporal framework for timely completion of cloud workflows

Yun Yang – 杨耘 (yyang@swin.edu.au)
http://www.ict.swin.edu.au/personal/yyang

SUCCESS
Swinburne University of Technology
Melbourne, Australia
SUCCESS – A Brief Introduction

Swinburne University Centre for Computing and Engineering Software Systems

- Swinburne is one of top 500 universities in the world
  - 2nd smallest one (less faculties than MIT CS Lab)

- SUCCESS has the strongest SE group in Australia
  - five full professors

- 2011 figures on two top SE journals:
  - TSE – IEEE Trans. on Software Engineering
    4 (2+2) (world total: 48)
  - ToSEM – ACM Trans. on Software Engineering and Methodology
    2 (1+1) (world total: 18)
Melbourne – Capital City of Victoria

- a very dynamic city
- population just over 4 million
- Australia’s cultural capital
- famous for parks and gardens
- “The Most Liveable City in the World”
- Welcome for (joint) PhD program etc.
Outline

- Related Publications (and Acknowledgement)
- Background
- Motivating Example and Problem Analysis
- A Probabilistic Temporal Framework
- Evaluation
- Conclusion
Acknowledgement: Assoc. Prof. Jinjun Chen; Dr Xiao Liu (two former PhD graduates)


Background: Workflow QoS

- QoS dimensions
  - time, cost, fidelity, reliability, security ...

- QoS of Cloud Services

- Workflow QoS
  - the overall QoS for a collection of cloud services
  - but not simply add up!
Temporal QoS

- System performance
  - Response time
  - Throughput

- Temporal constraints
  - Global constraints: deadlines
  - Local constraints: milestones, individual activity durations

- Satisfactory temporal QoS
  - High performance: fast response, high throughput
  - On-time completion: low temporal violation rate
Motivating Example

- **Astrophysics: pulsar searching**
  - Pulsars: the collapsed cores of stars that were once more massive than 6-10 times the mass of the Sun
  - [http://astronomy.swin.edu.au/cosmos/P/Pulsar](http://astronomy.swin.edu.au/cosmos/P/Pulsar)

- Typical scientific workflow which involves a large number of data and computation intensive activities. For a single searching process, the average data volume (not including the raw stream data from the telescope) is over 4 terabytes and the average execution time is about 23 hours on Swinburne high performance supercomputing facility ([http://astronomy.swinburne.edu.au/supercomputing/](http://astronomy.swinburne.edu.au/supercomputing/)).
Pulsar Searching

Data Collection:
- Collect data

Data Pre-processing:
- Extract Beam
- Transfer Data
- De-disperse (1200)
- De-disperse (2400)
- De-disperse (3600)

Candidate Searching:
- Seek
- Accelerate
- Get Candidates
- Eliminate candidates
- Fold to XML

Decision Making:
- Make Decision

\[ U(SW_1) = 15.25 \text{ hours} \]
\[ U(SW_2) = 5.75 \text{ hours} \]
\[ U(SW) = 24 \text{ hours} \]
Problem Analysis

- Setting temporal constraints
  - Coarse-grained and fine-grained temporal constraints
  - Prerequisite: effective forecasting of activity durations

- Monitoring temporal consistency state
  - Monitor workflow execution state
  - Detect potential temporal violations

- Temporal violation handling
  - Where to conduct violation handling
  - What strategies to be used
Ultimate Goal

- Achieving on-time completion

- Measurements:
  - Temporal correctness
  - Cost effectiveness
Temporal Framework
Temporal Framework

- **Component 1: Temporal Constraint Setting (JSS, CCPE)**
  - Forecasting workflow activity durations
  - Setting coarse-grained temporal constraints
  - Setting fine-grained temporal constraints

- **Component 2: Temporal Consistency Monitoring (TSE, ToSEM)**
  - Temporal checkpoint selection
  - Temporal verification

- **Component 3: Temporal Violation Handling (TSE, JSS)**
  - Temporal violation handling point selection
  - Temporal violation handling
Temporal Checkpoint Selection

- Requirements / Objectives

- Checkpoint: the point (e.g. activity point, time point) for conducting temporal verification

- The measurements for temporal checkpoint selection
  - Necessity: only those activity points where real temporal inconsistency states take place are selected
  - Sufficiency: there are no any omitted activity points

- Efficiency

- Effectiveness
Temporal Checkpoint Selection

- Existing Work

- Representative Checkpoint Selection Strategy (CSS)
  - Every activity as a checkpoint
  - The start activity, and add a new checkpoint after each decision activity is executed
  - User defined static activity points
  - The activity duration exceeds its maximum duration
  - The activity duration exceeds its mean duration

- Problems: necessary? Sufficient?
Temporal Checkpoint Selection

- Our Strategy

Necessary and Sufficient Checkpoint Selection Strategy

- Probability Time Redundancy
- Minimum Probability Time Redundancy
- DOMTR: Dynamically Obtaining Minimum Time Redundancy
- Theorem of Checkpoint Selection
- Proof of Necessity and Sufficiency
Temporal Verification

- Multi-states temporal violations
- Statistical recoverable and non-recoverable temporal violations
Simulation Environment

- SwinCloud

Cloud Simulation Environment

Data Centres with Hadoop

VMware

Swinburne Computing Facilities

- Astrophysics Supercomputer
  - GT4
  - SuSE Linux

- Swinburne CS3
  - GT4
  - CentOS Linux

- Swinburne ESR
  - GT4
  - CentOS Linux
CSS Types

- **CSS$_1$:** every activity as a checkpoint.
- **CSS$_2$:** start time and end time of each activity.
- **CSS$_3$:** start activity, after each decision activity.
- **CSS$_4$:** user-defined static checkpoints.
- **CSS$_5$:** $a_i$ as a checkpoint if $R(a_i) > D(a_i)$.
- **CSS$_6$:** $a_i$ as a checkpoint if $R(a_i) > M(a_i)$.
- **CSS$_7$:** $a_i$ as a checkpoint if $R(a_i) > M(a_i) + \text{a minimum proportional time redundancy at } a_i$.
- **CSS$_8$:** $a_i$ as a checkpoint if $R(a_i) > M(a_i) + \text{a minimum time redundancy at } a_i$. 
Evaluation

Total Unnecessary Checkpoints Selected by Each CSS

Unnecessary Checkpoints vs Workflow Size (Activities)

- CSS_2
- CSS_1
- CSS_0
- CSS_5
- CSS_7
- CSS_4
- CSS_3
- CSS_MTR
Evaluation

Total Checkpoints Omitted by Each CSS

Omitted Checkpoints

Workflow Size (Activities)

CSS_2
CSS_1
CSS_0
CSS_5
CSS_7
CSS_4
CSS_3
CSS_MTR
Evaluation

![Graph showing verification computation units vs. number of upper bound constraints]
Evaluation

Adjustment Point Selection

Number of Adjustment Points vs. Number of Workflow Activities

- 0% Noise
- 10% Noise
- 20% Noise
- 30% Noise
Conclusion

- On-time completion of scientific workflows
- Lifecycle support of temporal QoS
- A probabilistic temporal framework
  - Setting temporal constraints
  - Monitoring temporal consistency
  - Handling temporal violation
- Necessary and sufficient checkpoint selection
Future Work

- Instance intensive business workflows
- Fast response time vs. high system throughput
- Resource management in cloud computing environment
  - Service level agreement (SLA) management
  - Cloud resource reservation
  - Dynamic Scheduling
End - Q&A

- Thanks for your attention!