A Method of Improving QoS: Explorations of the Possibility
of Function Combinations in Service Compositions

by

Ziyuan Wang

A Senior Thesis

Submitted to
the Department of Information Science
the Faculty of Science, the University of Tokyo
on February 9, 2016
in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Science

Thesis Supervisor: Shinichi Honiden
Professor of Information Science

ABSTRACT

There is a growing need for web service providers to develop customised and flexible web services as quick as they can. One way to satisfy this demand is to utilise service compositions, which provide a method of consolidating several services to a richer service. Because of the uncertainty in the feasibility of the functional and QoS requirements, it is not guaranteed that service compositions succeed with suitable solutions.

One way to satisfy the given requirements is to control the given two kinds of requirements. There have been studies on methods of optimising the QoS with fixed functional requirements. However, the search space in that case is limited, which would possibly not include service compositions with better QoS whose functions are slightly different from the required one. Therefore, I focus on the possibility of improving the QoS by compromising on functions or exploring the possibility of function combinations. This enables better advice of service compositions to users. In this article, I propose a method of efficiently searching different possibilities to support users to balance functional requirements and QoS requirements.

QoS

Acknowledgements

Contents

Chapter 1

INTREODUCTION

Web service, service-based system, and service composition are significant. A lot of existing studies on QoS-aware service composition. Fixed functional requirements -¿ may lose potential improvement of QoS, difficulty for users to properly define the functional requirements because of uncertainty of the feasibility of requirements given available services Tis paper ... (summary of the remainder). Chapter 2 describes ... Chapter 3...

1.1 Background

1.2 Goals

1.3 Contributions

[1].

Chapter 2

Preliminary

This section defines some key terms of formal research of the service composition that I continue to use in this paper.

2.1 Service

A service S is a reusable system that provides functionalities which are documented in a service description. This description defines a 5-tuple $IOPEQ = (S.I, S.O, S.P, S.E, S.Q)$ where $S.I, S.O$ are abbreviated from the required input and output paremeters of the S, $S.P, S.E$ are the preconditions and effects which means the necessary condition of utilising S and the consequence after running S, and $S.Q$ is the set of the QoS attributes of the S.

In this paper, $S.P, S.E$ are ignored therefore 5-tuple are model only with 3-tuple IOQ.

2.2 QoS

multi-criteria normalized

2.3 Functional Compliance

Connectivity between two services
    def: strong/weak

2.4 Workflow

Workflow templates: sequence of service tasks Workflow: sequence of services (service instances)
    Functional requirements: (existing)

2.5 QoS optimization

this time single-objective

Chapter 3

Extended Composition Problem

: Proposal1
motivation again
(actually, not so meaningful to use too weak services)

Chapter 4

Algorithm for Extended Composition Problem

: Proposal 2

4.0.1 Functionality Graph

From [Wagner'11] def. algorithm

4.0.2 Skyline

def. algorithm

4.0.3 Naive Algorithm

4.0.4 Algorithm

Chapter 5

Experiments

5.1 Implementation

Implemented with Python.

5.2 Extended Composition Problem

Evaluation of Proposal 1 How QoS can be improved by allowing compromise in functional requirements?

5.3 Algorithm

Evaluation of Proposal 2 How fast can the proposed algorithm solve the extended problem? - with skyline vs. without skyline - with skyline vs. proposed algorithm with skyline

Chapter 6

Discussion

6.0.1 Proposal

The extended problem should be significant because in the experiment
The proposed algorithm made good improvement because in the experiment

6.0.2 Future Work

- QoS constraints - adaptive

Chapter 7

Conclusion

References

[1] E. Goto. The parametron, a digital computing element which utilizes parametric oscillation. *Proceedings of the IRE*, 47(8):1304–1316, 1959.