SELECTING THE BEST COMMUNICATION SERVICE IN FUTURE NETWORK ARCHITECTURES

Rahamatullah Khondoker *, *Paul Mueller*

Integrated Communication Systems
University of Kaiserslautern
Kaiserslautern, Germany
{khondoker, pmueller}@informatik.uni-kl.de

Kpatcha Bayarou

Fraunhofer Institute for Secure
Information Technology (FhG-SIT)
Darmstadt, Germany
kpatcha.bayarou@sit.fraunhofer.de

ABSTRACT

As the number of future network architectural approaches increases, the possibility of offering many similar services with different qualities of service is increasing. Therefore, it will be required to select a suitable, or the best, service from the set of alternative services. This paper proposes a matching process and an adapted analytic hierarchy process to accomplish this task. The matching process is used to determine if a service is suitable. When more than one suitable service is available, the adapted analytic hierarchy process is used to select the best service.

Keywords— Future Internet, NGN, service-orientation, service description, network architectures, service selection

1. INTRODUCTION

In todays Internet, protocols are tightly coupled with the application, which results in difficulties in automatically switching between the functionalities based on the application requirements. Traditionally, an email application uses TCP, a Voice over IP (VoIP) application uses UDP, some video streaming applications use SCTP. However, a video application cannot just switch between UDP and SCTP based on its variety of demands.

For introducing flexibility in network architectures and enabling innovations, several projects like GENI, FIND, G - Lab, PL - Lab, AKARI, have been funded in USA, Europe and Asia. The results of these projects are a set of future network architectures like Autonomic Network Architecture (ANA) [1], Netlet-based Node Architecture (NENA) [2], eXpressive Internetwork Architecture (XIA) [3], Service-Oriented Network Architectures (SONATE) [4] and Recursive InterNetwork Architecture (RINA) [5].

Some of these approaches are based on communication services. Here we consider only communication services not web services. A communication service can represent a fine-grained functionality like an algorithm for forward error correction (e.g., hamming code) or compression (e.g., Huffman tree) or it can even represent a coarse-grained functionality like the functionality of the TCP/IP network stack or an access technology like WiFi.

Most of future network architectural approaches need to use a suitable service, or to select the best service, if there more than one suitable service is available. Selection of a suitable service can be done by matching the description of the offered services with the application requirements. This match can result in a several suitable services. Now, the question is, which suitable service should be selected and used? The answer is that we should select the best one, as we do in our day to day life.

Selecting the best service using a single selection criterion is trivial. For example, if there are two communication services where one offers 100ms end-to-end delay and another offers 200ms, then we should obviously select the one with the lowest delay.

However, communication services have multiple selection criteria such as delay, throughput, loss ratio, jitter and cost. That is why, selecting the best communication service is a Multi-Criteria Decision Making problem (MCDM). For solving such a problem, several Multi-Criteria Decision Analysis (MCDA) approaches are used in managerial science like Multiple Attribute Utility Theory (MAUT), Analytic Hierarchy Process (AHP), ELECTREIII and Evamix [6].

We used AHP to select the best service for two reasons, firstly, it supports relative prioritization and, secondly, there is a way to check the consistency of the evaluation measures.

The main requirement for using AHP is to assign pairwise priority both for the requirements and for the offers. However, as offerings are decoupled from the application requirements, a mapping mechanism is required from the measured values of the offerings to the pairwise priority assignment scale. We use a mapping mechanism based on monotonic interpolation and extrapolation.

The outline of the paper is as follows. We present a service selection model in section 2. The components of communication service selection using the analytic hierarchy process are discussed in section 4. In a service-oriented network architecture, offerings are decoupled from the application, for

*Main corresponding author. He is a PhD student at the department of computer science in the University of Kaiserslautern. Moreover, he is affiliated with the Fraunhofer Institute for Secure Information Technology located in Darmstadt, Germany

this reason a mapping mechanism is necessary to map from the measured value of the offers to the pairwise prioritization scale. We propose a mapping mechanism using monotonic interpolation and extrapolation in section 4.3.2. We implemented and evaluated the selection process using a maximum of six selection criteria and six services which is discussed in section 5. After that, related work for future network architectural approaches and service selection is presented in section 6. Section 7 concludes the paper.

2. SERVICE SELECTION MODEL

A model for fine-grained service selection and composition is shown in figure 1. The main aim of the process is to create a protocol graph (i.e., a network stack) for a network connection. To achieve this goal, it takes the requirements from the application, constraints from the network, policies from the network or system administrator, and the offered services from the network. Considering all of these inputs, it composes the protocol graph of building blocks (the implementation of a protocol or a mechanism). Automatic selection of a suitable, or the best, fine-grained functionality is required during the composition process.

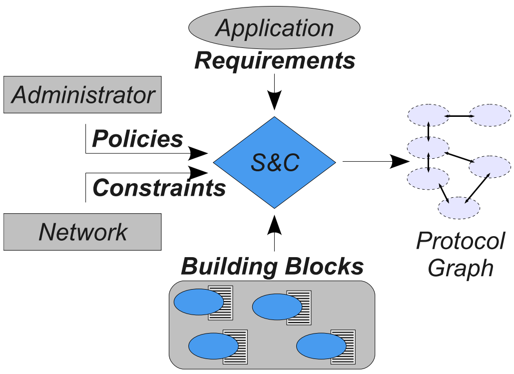

Figure 1. A model for fine-grained service selection and composition

A model for coarse-grained service selection is shown in figure 2. The three main entities in this model are the service consumer, the service provider and the service broker. The service broker selects a suitable, or the best service, from the services offered by the different service providers by considering the requirements specified by (or chosen from the predefined specification) an application developer through an application programming interface (API). Service providers like SONATE and NENA frameworks can be categorized based on their composition approaches. Services can be offered by conventional providers like TCP / IP, UDP / IP and SCTP / IP. Services can be composed during design time, deployment time, partial runtime and runtime. In compound approaches, services are composed during design time, potentially assisted by software. In this approach, the selection of an appropriate compound service is done during runtime by a service broker. The template approach is an example of partial runtime composition, where the placement of functionalities is done during design time and a suitable, or the best, mechanism is chosen during runtime. Services can also be provided by a dynamic selection and composition provider where the selection and composition of the protocol graph is done during runtime.

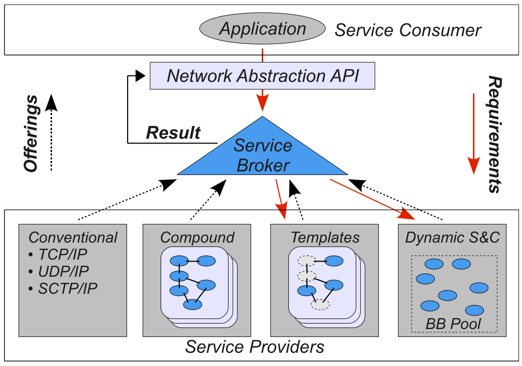

Figure 2. A model for coarse-grained service selection in a service-oriented network architecture

Partial runtime and dynamic selection and composition providers cannot register their services to the broker until they get the application's requirements and perform their composition. Other providers can register their service to the broker beforehand.

The service broker returns a suitable, or the best, service to the application through the API.

3. TERMINOLOGY: CRITERIA FOR SERVICE SELECTION

The criteria that are used to select a suitable, or the best, communication service are specified by the field expert. The assumption here is that, an experienced VoIP application developer knows the criteria that should be considered for his application. Even though functional criteria are also considered in service selection, we considered here only the following quality of service criteria:

1. Delay: Delay is defined as the elapsed time to transfer a packet from the sender's application to the destination receiver's application across the network. Delay is measured by seconds or fraction of seconds. [7]

2. Jitter: Variation in delay of packets arriving in the destination.

3. Energy Consumption: The power that is required to process a packet is called energy consumption. Energy consumption is usually measured in Joules (J).

4. Data Length: The length of a packet consisting of a payload of data and a header is called data length (sometimes called packet length).

5. Loss Rate: When a transmitted packet does not successfully arrive at its destination, it is called a lost packet. The loss rate is the ratio of the number of lost packets and the total number of sent packets. Loss rate can also be called Packet Loss Ratio (PLR). [7]

6. Throughput: The average rate of successful data delivery over a wired or wireless communication is called throughput. Throughput is measured in bits per second, in short, bits/sec or bps. [7]

4. COMPONENTS FOR SERVICE SELECTION

Service selection is a process to select a suitable service, or the best service if more than one suitable service is available. The components of our service selection approach are:

1. Description of application requirements and network offerings
2. Matching process
3. Analytic Hierarchy Process
4. Network abstraction API

4.1. Description of application requirements and network offerings

Service selection requires the description of application requirements, network and administrator constraints, and network offerings. This requirement can be fulfilled by the description language for communication services of future network architectures [8]. All of these requirements, constraints and offerings can be described by using the construct $\{effect\ operator\ attribute\}$.

An effect is a single outcome of an execution of algorithm or protocols, sometimes called building blocks. Effects can be functional and non-functional. Functional effects are the effects which are required for proper functioning of a building block. For example, the effect $LossRatio$ can be used by the retransmission building block to know how many packets to retransmit. Non-functional effects, on the other hand, are the effects which might not be necessary for functioning. For example, the processing time of a building block can be seen as an example of such an effect.

An attribute is the value of an effect. For example, 0% can be seen as an attribute of the effect packet loss.

An operator connects an effect to an attribute. The packet loss offering of a retransmission building block can be written as $\{LossRatio = 0\%\}$.

This simple construct can be used to express the requirements of an application. For example, the error correction demand of an email application can be expressed as $\{ErrorCorrection = True\}$.

The usage of an effect in the description is mandatory. But, the usage of an operator and an attribute is optional. For example, the error correction demand can be described as $\{ErrorCorrection\}$ by omitting an operator and an attribute.

This construct allows the description of the network offerings. For example, the packet loss offering of a forward error correction algorithm can be expressed as $\{LossRatio = 0\%\}, \{Delay = low\}, \{Bandwidth = high\}$.

A network or administrator constraints can be expressed by using the construct. For example, for using a certain network, authentication must be performed $\{Authentication = True\}$.

This construct supports to describe both fine-grained and coarse-grained functionality in a similar way. For example, the ProcessingTime of a single building block or a protocol graph can be expressed by using the same construct.

4.2. Matching process

Suitable services are chosen by matching the offered effects with the required effects. For example, an application can support the maximum end-to-end delay of 100 ms which is expressed by $\{end-to-endDelay <= 100ms\}$ whereas a protocol graph offers $\{end-to-endDelay = 80ms\}$. The broker can select the protocol graph as a suitable service.

For matching application requirements with the network offerings, each effect must be uniquely identified. This necessitates developing a taxonomy of effects to describe communication service illustrated in the ITU-T paper [9]. This taxonomy facilitates an application developer to specify effects either in a generic manner or in a specific way. For example, an application developer can ask for the Security effect in general, $\{Security = True\}$, or it can ask for the data origin authentication effect, $\{Data-Origin-Authentication = True\}$, to be more precise.

As the values of the offered effects are measured or precalculated values, mostly, they contain the operator is equal to $(=)$.

But, the required effects might contain other operators like less than $(<)$, less than or equal to $(<=)$, greater than $(>)$ and greater than or equal to $(>=)$.

An application might also express its requirements as an interval. For example, a video streaming application might express its packet loss requirement as $\{LossRatio <= 3\%\}$.

The application can work when the packet loss is between 0% and 3%.

During the selection process of fine-grained or coarse-grained functionalities, several of them can be determined as suitable services when they match the requirements from the application. In that case, the best service should be selected and used. We adapted Analytic Hierarchy Process (AHP) for doing this task.

4.3. Analytic Hierarchy Process (AHP) for service selection

Selecting the best service using a single selection criterion is trivial. For example, if there are two communication services where one offers 100ms end-to-end delay and another offers 200ms, then we should obviously select the one with less delay.

However, communication services have multiple selection criteria such as delay, throughput, loss ratio, jitter and cost. That is why, selecting the best communication service is a Multi-Criteria Decision Making problem (MCDM). For solving such a problem, several Multi-Criteria Decision Analysis (MCDA) approaches are used in managerial science like Analytic Hierarchy Process (AHP) [10], ELECTREIII [11], Evamix [12], Multiple Attribute Utility Theory (MAUT) [13], Multi - Objective - Programming (MOP), Goal Programming (GP) [14], NAIADE [15] and Regime [16].

We used AHP to select the best service for two reasons, firstly, it uses an absolute scale to derive priorities that also belong to the relative absolute scale (like probabilities) that can be combined like the real number system. secondly, there is a way to check the consistency of the evaluation measures.

4.3.1. Adaptation of Analytic Hierarchy Process (AHP) for service selection

The Analytic Hierarchy Process (AHP) needs to be adapted for selecting the best communication service automatically.

AHP is a process designed for assisting human decision making which is used in many application areas like social, personal, education, manufacturing, political, engineering, industry and government [17]. Basically, AHP is used for determining priorities of different alternatives. The details of the AHP process is beyond the scope of this text.

To use AHP in communication service selection, the following steps are performed

1. Define the goal and the selection criteria for achieving the goal
2. Priority assignment of the selection criteria as an application requirement
3. Priority assignment of the criteria for the offered services

The first step is to define the goal, which is to select the best communication service, and the selection criteria to achieve that goal. The selection criteria are actually a set of required effects. Examples of selection criteria are delay, throughput, loss rate, jitter, MTU and cost. Both functional and non-functional criteria can be selected.

After determining the selection criteria, the next step is to assign pairwise priority between the selection criteria. One of the reasons of pairwise priority assignment is that it is easier for a person to take two criteria and to assign priority one over the other. It is initially difficult for a new application developer to assign pairwise priority. But, the efficiency of the priority assignment process can be improved with the experience of the application developer.

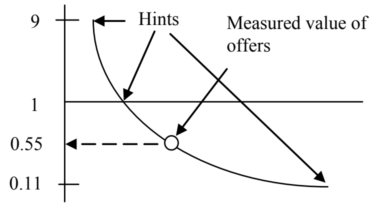

Figure 3. Mapping mechanism

The third step of the process is to assign pairwise priority between the offered services based on those selection criteria. However, as pairwise priority assignment is a time-consuming task, and as offerings are decoupled from the application, the pairwise priority assignment of the offered services based on those selection criteria needs to automated.

This requires a mapping mechanism to map the measured/ calculated values of the offered services to the pairwise priority assignment scale which will be discussed in the next section.

The priority vector coming from the application side is then multiplied by the priority vector from the offering side. The result is then called the overall priority vector. The service with the highest priority value in the overall priority vector is the best service.

4.3.2. Automated priority assignment for the offerings

Different communication services can have different effects. The value (or attribute) of these effects can be assigned beforehand based on benchmarks or can be obtained dynamically by using sensing software. Whichever way the attributes are obtained, the offered effects need to be automatically prioritized as the offerings are decoupled/hidden from the application. Therefore, an automatic mapping mechanism from measured values to the priority scale (1, 9) is required.

The mapping should have certain properties. First, the mapping must be generic, i.e. not specific to effects or units of measured values. Second, the mapping must be monotonic.

An approach for mapping has been proposed which uses a monotonic interpolation/extrapolation scheme [9] as shown in figure 3. In this case, the application requirements provide value points for interpolation/ extrapolation (must be monotonic) of measured values to the priority scale. A monotonic interpolation/extrapolation of these points is used to define a mapping. In addition, the specific measured values of the offerings are then mapped to these priorities. Assuming that $f()$ is a function used to define a mapping. As an example, considering interpolation, the requirements must contain at least the following two points

- $x_0, where f(x_0) = 1$
- $x_n, where f(x_n) = 9$

If there are measurement values, y, not within the interval $[x_0, x_n]$, we can extrapolate

- $if y < x_0, then f(y) = 1$
- $if y > x_n, then f(y) = 9$

To use inter-/extrapolation, an application developer must specify two points but can have as many parameters as he wants to be more precise.

The aforementioned mapping mechanism is used to assign a priority of one service over another for every selection criteria (effect).

4.4. Network abstraction API

An application programming interface (API) is required to send the application requirements to the broker and to return a suitable or the best service to the application. Affiliated with the SIG FUNCOMP, a special interest group for functional composition of the German-Lab project, we created an interface titled *GAPI: A G-Lab Application-to-Network Interface* which can be used for this purpose [18].

5. IMPLEMENTATION

The aforementioned service selection process has been implemented using the Java programming language version 1.6. The requirements and offerings are assigned statically in variables. No database is used to store those values. A separate method has been implemented to map the offered values to priorities as shown in the figure 4.

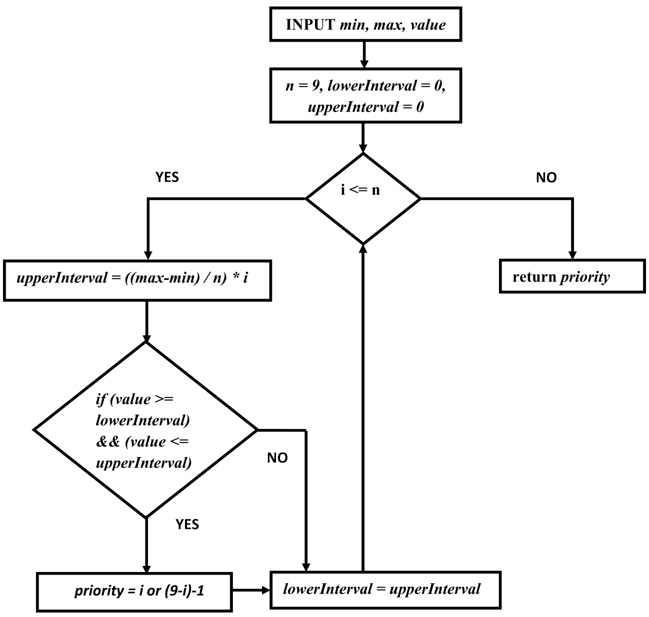

Figure 4. Priority assignment algorithm

5.1. Service selection time

To measure selection time, six selection criteria have been chosen and pairwise prioritized as shown in figure 5. The values of 0.11, 1 and 9 means that the lowest, equal and the highest priorities respectively. The measured values of the offerings is shown in the figure 6. For this experiment, CentOS is used on a Pentium(R) Dual-Core CPU E5300 with 2.6 GHz speed and 6 GB RAM.

Delay	Throughput	Jitter	Loss Rate	Energy Consumption	Data Length
1	5	9	5	9	5
0.2	1	1	2	1	1
0.11	1	1	1	1	1
0.2	0.5	1	1	2	5
0.11	1	1	0.5	1	1
0.2	1	1	0.2	1	1

Figure 5. Pairwise priorities of the six selection criteria

Effects	S1	S2	S3	S4	S5	S6
Delay	10	50	250	200	220	260
Throughput	1	2	10	12	15	32
Jitter	1	2	10	15	17	20
Loss Rate	2	4	5	8	10	15
Energy Consumption	10	50	40	70	120	100
Data Length	1500	500	600	1400	700	1000

Figure 6. Measured values of the offered services

Beginning with the two selection criteria and two services, both selection criteria and the offered services have been incremented by 1 until 6 and the service selection and mapping times have been measured. We found that the mapping time is linearly increased with 23 micro seconds is required for mapping the 6 services using the six selection criteria as shown in the figure 7. Selection time is exponentially increased and requires 0.48 ms to select the best service among the six offered services using the 6 selection criteria as shown in figure 8.

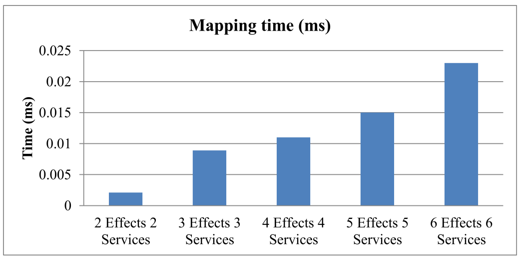

Figure 7. Mapping time

Mapping can be done during runtime or beforehand, when the measured values are already available. In that case, only selection time is considered.

5.2. Benefits and future work

This selection approach has several advantages; first, pairwise prioritization of requirements as an input, second, con-

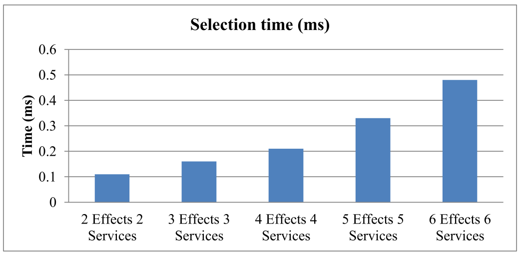

Figure 8. Selection time

sistency checking, third, benefits of relative prioritization over linear prioritization.

It is easy for people to compare two objects by using their properties. For example, a recruitment manager needs to select the best candidate for the job. One candidate has an excellent education but no working experience and another person has a good education but has 2 years of experience. The manager will take these two selection criteria of the candidate (education, working experience) and can easily identify which is more important to him. If working experience is more important to him, he will select the second candidate and otherwise he will select the first one.

When the number of selection criteria increases, the consistency of the pairwise priority assignment needs to be checked. As discussed earlier, the analytic hierarchy process provides a way to check consistency.

AHP uses relative prioritization rather than linear prioritization which is used in MAUT. In linear prioritization, the priority value of the requirement is assigned linearly like $delay > throughput > loss$ which means that a service with the lowest delay should be selected at first. If two services have the same delay, then the service with the highest throughput is selected. In relative prioritization, the selection criteria is pairwise prioritized. That means, a service is selected based on all of the considered criteria not only a single criteria like in a linear prioritization technique.

Currently, the selection time is calculated by considering at most six effects and services, as today the number of networking services and selection criteria is limited. However, in the future, evaluation can be done by increasing the number of services and criteria.

6. RELATED WORK

The work related to layerless future network architectures is presented at first. Then, the work for service selection is presented.

In the early 1990s, a small group of network researchers concentrated on dynamic micro-protocol composition, meaning that they decomposed the functionality of existing protocol stacks into a set of micro-protocols, and then composed those micro-protocols dynamically based on incoming requests from an application. Some of those works are Dynamic Configuration of Protocols (DaCaPo) [19] and Function Based Communication Subsystem (FCSS) [20]. In [21] the authors point to a drawback of the above approaches and ask for a generic description so that new deployments can be facilitated and implementation customization can be kept to a minimum. [22] focused on networking protocols rather than the functionality, services or roles provided by those protocols which were focused on by [23] and [24].

Some recently completed and ongoing projects are working on Network Functional Composition. Those projects are Automatic Network Architecture (ANA) [1], NetServ [25], Recursive InterNetwork Architecture (RINA) [5], eXpressive Internet Architecture (XIA) [3], Forwarding on Gates (FoG) [26], Net-Silo [24], 4WARD [27], Self-Net (Self-Management if Cognitive Future Internet Elements) [28] and the Recursive Network Architecture (RNA) [29]. Descriptions of some of the aforementioned projects has been comprised in a state-of-the-art paper [30].

A template-based approach is similar in concept to the NENA approach. In the NENA approach, netlets (i.e., a network stack) for each domain are composed during design time by network engineers assisted by software. Selection of an appropriate netlet is done during runtime by using MAUT [31]. However, selection of appropriate mechanisms (i.e., building blocks) is not done in the NENA approach. In the template-based approach, not only are appropriate templates selected at runtime but also appropriate mechanisms are selected.

As selecting communication services to make a protocol stack automatically is a new field, few related works have been found. The mentionable one is a MCDA approach, MAUT which is used in NENA to select the best composed protocol stack during runtime. However, MAUT has no integrated mechanism to check consistency of the given priorities. That is why, an external mechanism is required for doing this task which is not available.

Most of the approaches right now use static selection of functionality during design time. Some of those approaches are ANA, RINA, XIA and FoG.

7. CONCLUSION

Driven by Future Internet projects like GENI and FIND, worldwide research of future network architectures results in several architectural approaches like NENA, XIA, SONATE, RINA, and ANA, to name a few. Even though the same service with different qualities of attributes can be offered by the same architecture, the probability of having such a case can be even higher when there are many architectural approaches and virtualization techniques.

Therefore, selecting a suitable, or the best service, based on application requirements is essential. A suitable service can be selected just by matching the description of the offered services with the requirements. Selection of the best service is required.

Selecting the best service using a single criterion is trivial. For example, considering a single selection criterion delay,

Table 1. The requirements matrix (CR = 6.23%)

Effects	Delay	Throughput	Jitter	Priority
Delay	1	5	9	0.7651
Throughput	0.2	1	1	0.1288
Jitter	0.11	1	1	0.1062

the best service is the one with the lowest delay. However, communication services have multiple selection criteria. That is why, selecting the best service is a multi-criteria decision making problem.

For solving such a problem, different multi-criteria decision analysis methods exist in management science. For example, MAUT, AHP, Evamix, Regime, ELECTRE III, NAIADE and MOP/GP. We chose the Analytic Hierarchy Process (AHP) for communication service selection as it supports relative prioritization and checks consistency.

However, the process is required to be adapted for communication service selection. In a service oriented network architecture, offerings are decoupled from the application. That is why the measured or estimated values of the offered services need to be mapped based on the hints coming from the application. This is done by the proposed mapping mechanism.

We implemented the process of selecting the best service in the Java programming language and evaluated using at most six selection criteria (effects) and offered services. The result shows that 0.503 milliseconds (selection time (.48 ms) + mapping time (.023 ms)) is required to select the best service between six offered services using six selection criteria.

To conclude, applications use networks differently, and therefore have different network requirements. At the same time, networking capabilities and protocols make advances. This paper shows how applications can make use of advancing network capabilities by specifying requirements and using a selection process to choose the best available communication service.

Describing application requirements and communication services supports the parallel development of both applications and communication services, which leads to the evolution of the Internet. As soon as new protocols or networks emerge that fulfill the application requirements, they can be automatically selected by using the service selection process.

8. APPENDIX 1: BEST SERVICE SELECTION: AN EXAMPLE

The goal is to select the best service among the three services: S1, S2 and S3. For achieving this goal using our approach, we choose three selection criteria: Delay, Throughput and Jitter and pairwise-prioritized them as shown in Table 1. As it is seen in the table, delay is given strongly more important than (5) throughput and absolutely more important (9) than Jitter. To make the matrix consistent, throughput and jitter are assigned strongly less important than (0.2) and absolutely less important than Delay (0.11) respectively.

Table 2. Measured/Estimated values of Services

Services	Delay (ms)	Throughput (Mbps)	Jitter (ms)
S1	10	1	1
S2	50	2	2
S3	250	10	10

Table 3. Overall priority vector computation

Req. vector	0.7651	0.1288	0.1062	Priority
	Delay	Throughput	Jitter	
S1	0.5869	0.0740	0.5790	0.52
S2	0.3583	0.1176	0.3685	0.33
S3	0.0549	0.8084	0.0524	0.15

Assuming that the services S1, S2 and S3 offer the values of Delay, Throughput and Jitter according to table 2.

These values are mapped to the scale of (1, 9) using the the mapping algorithm depicted in figure 4. The requirement matrix is consistent as its consistency ratio is less than 10%. The overall priority is then obtained by multiplying the priority vector of the requirement matrix with the offered matrix. The service with the highest value in the overall priority vector is chosen as the best service, which is S1, as shown in table 3.

9. ACKNOWLEDGMENTS

We acknowledge our colleagues for their valuable suggestions. Thanks to Prof.Thomas L. Saaty for providing helpful feedback regarding AHP.

REFERENCES

[1] “ANA,” http://www.ana-project.org/, Online; accessed 25-April-2011.

[2] L. Voelker, D. Martin, I. E. Khayat, C. Werle, and M. Zitterbart, “A node architecture for 1000 future networks,” in *International Workshop on the Network of the Future 2009, Dresden, Germany*, 2009.

[3] “XIA,” http://www.cs.cmu.edu/ xia/technical/technical-approach.html, Online; accessed 3-Feb-2012.

[4] Paul Müller and Bernd Reuther, “Future internet architecture - a service oriented approach (future internet architecture - ein serviceorientierter ansatz),” *it - Information Technology*, vol. 50, no. 6, pp. 383–389, 2008.

[5] “RINA,” http://csr.bu.edu/rina/, Online; accessed 03-February-2012.

[6] Andrea de Montis, Pasquale De Toro, Bert Droste-Franke, Ines Omann, and Sigrid Stagl, “Assessing the quality of different mcda methods,” 2000.

[7] Rahamatullah Khondoker, "Implementation and evaluation of regular channel access for cooperative spectrum sharing between coexisting wlan systems," *Master Thesis*, 2009.

[8] Rahamatullah Khondoker, Eric MSP Veith, and Paul Mueller, "A description language for communication services of future network architectures," *In Proceedings of the 2011 International Conference on the Network of the Future*, pp. 69 – 76, 2011.

[9] Rahamatullah Khondoker, Bernd Reuther, Dennis Schwerdel, Abbas Siddiqui, and Paul Mueller, "Describing and selecting communication services in a service oriented network architecture," in *the proceedings of the 2011 ITU-T Kleidoscope event, Beyond the Internet? Innovations for future networks and services, Pune, India*, December 2010.

[10] T. L. Saaty, "The analytic hierarchy process," *McGraw-Hill, New York*, 1980.

[11] B. Roy, "Multiple criteria methodology for decision aiding," *Parigi: Economica*, 1985.

[12] H. Voogd, "Multi-criteria analysis with mixed qualitative-quantitative data," *Delft University of Technology, Department of Urban and Regional Planning*, 1981.

[13] R. L. Kenny and H. Raiffa, "Decisions with mulitple objectives: Preferences and value trade-offs," *John Wiley and Sons, New York*, 1976.

[14] Matthias Ehrgott and Xavier Gandibleux, "Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys," *Kluwer Academic Publishers*, 2003.

[15] G. Munda, "Multi Criteria Evaluation in a Fuzzy Environment - Theory and Applications in Ecological Economics," *Hidelberg: Physika Verlag*, 1995.

[16] E. Hinloopen, "De regime methode, ma thesis," *Interfaculty Actuariat and Econometrics, Free University Amsterdam*, 1985.

[17] Thomas L. Saaty, "Decision making with the analytic hierarchy process," *Int. J. Services Sciences*, vol. 1, no. 1, pp. 83–98, 2008.

[18] Florian Liers, Thomas Volkert, Denis Martin, Helge Backhaus, Hans Wippel, Eric MSP Veith, Abbas Ali Siddiqui, and Rahamatullah Khondoker, "GAPI: A G-Lab Application-to-Network Interface," *In Proceedings of the 11th Würzburg Workshop on IP: Joint ITG and Euro-NF Workshop on "Visions of Future Generation Networks" (EuroView 2011)*, 2011.

[19] M. Vogt, Th. Plagemann, B. Plattner, and Th. Walter, "Eine laufzeitumgebung fuer da capo," *GI/ITG-Arbeitstreffen Verteilte Multimedia-Systeme*, 1993.

[20] B. Stiller, "FuKSS: Ein funktionsbasiertes kommunikationssubsystem zur flexiblen konfiguration von kommunikationsprotokollen," *GI/ITG-Fachgruppe Kommunikation und Verteilte Systeme*, 1994.

[21] Birgit Geppert and Frank Roessler, "Generic Engineering of Communication Protocols - Current Experience and Future Issues," in *ICFEM '97: Proceedings of the 1st International Conference on Formal Engineering Methods*, Washington, DC, USA, 1997, p. 70, IEEE Computer Society.

[22] M. Vogt, Th. Plagemann, B. Plattner, and Th. Walter, "A Run-time Environment for Da CaPo," in *Proceedings of INET93 International Networking Conference of the Internet Society*, 1993.

[23] Robert Braden, Ted Faber, and Mark Handley, "From Protocol Stack to Protocol Heap: Role-Based Architecture," *SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 17–22, 2003.

[24] R. Dutta, G.N. Rouskas, I. Baldine, A. Bragg, and D. Stevenson, "The SILO Architecture for Services Integration, controL, and Optimization for the Future Internet," in *Communications, 2007. ICC '07. IEEE International Conference on*, June 2007, pp. 1899–1904.

[25] Suman Ramkumar Srinivasan, Jae Woo Lee, Eric Liu, Michael Kester, Henning Schulzrinne, Volker Hilt, Srini Seetharaman, and Ashiq Khan, "NetServ: Dynamically Deploying In-Network Services," in *ReArch '09: Proceedings of the 2009 workshop on Re-architecting the internet, Rome, Italy*, 2009.

[26] Florian Liers, Thomas Volkert, and Andreas Mitschele-Thiel, "Forwarding on Gates: A clean-slate Future Internet Approach within the G-Lab project," in *EuroView 2009*, Würzburg, Germany, 2009.

[27] "4WARD EU Project," http://www.4ward-project.eu/, Online; accessed 25-April-2011.

[28] "Selfnet project," http://www.ict-selfnet.eu/, Online; accessed 10-July-2007.

[29] Joseph D. Touch, Yu-Shun Wang, and Venkata Pingali, "A Recursive Network Architecture," http://www.isi.edu/touch/pubs/isi-tr-2006-626/, 2006, Online, accessed 16-Nov-2012.

[30] Christian Henke, Abbas Siddiqui, and Rahamatullah Khondoker, "Network functional composition: State of the art," *2010 Australasian Telecommunication Networks and Application Conference, Auckland, New Zealand*, pp. 43 – 48, 2010.

[31] Lars Voelker, Denis Martin, Christoph Werle, Martina Zitterbart, and Ibtissam El Khayat, "Selecting concurrent network architectures at runtime," in *IEEE International Conference on Communications (ICC 2009)*, June 2009.