Service Availability Analysis of a Multimodal Travel Planner Using Stochastic Automata

Spyridon Evangelatos[†], Zannis Kalampoukis[*], Ioanna Fergadioti[†], Stelios Christofi[‡], Bill Karakostas[§] and Yannis Zorgios[*]
[†]Inlecom Systems
Phoenix House, 24 High Street, Wanstead, London, United Kingdom
E-mails: spyros@inlecom.com, ioanna@inlecom.com
[*]CLMS
4-6 University Way, London, United Kingdom
E-mails: z.kalampoukis@clmsuk.com, yz@clmsuk.com
[‡]eBOS Technologies
124 Strovolos Avenue, Strovolos 2042, Nicosia, Cyprus
E-mail: stelios@ebos.com.cy
[§]VLTN
Welvaartstraat 33B, Antwerpen 2000, Brussels, Belgium
E-mail: bill.karakostas@vltn.be

***Abstract*—In this paper, we address the problem of evaluating the aggregated services offered to the user through a multimodal travel planner. The effective design of a multimodal travel planner is considered a complex task since it requires the identification of the best possible route that combines segments of different transport modes. We describe a novel architecture and its respective connectivity infrastructure that incorporates all available travel-related services through the integration of several application programming interfaces (APIs) and we analyze its robustness regarding unplanned service interruptions using well-known mathematical tools such as the Stochastic Automata Networks. Additionally, we demonstrate the system's availability and effectiveness through realistic measurements obtained from the testing environments of various Global Distribution Systems (GDSs) and transport operators that result in the enhancement of the required Quality of Service (QoS) and Experience (QoE) for such intelligent systems.**

I. Introduction

Multimodal travel planning is an important element of Intelligent Transport Systems (ITS) deployment. It seamlessly integrates information for different transportation modes, allowing the travellers to make decisions about their planned journey. To achieve efficient travel planning, the information obtained from all the relevant providers must be integrated, so as to be rapidly and easily accessible on the spot. Additionally, in the case of service interruptions or delays, the travellers need to be alerted in real-time through a sophisticated mechanism that will allow them to re-plan their journey. The term of *multimodality* has been recently introduced in the travel industry in various forms, such as route planners for tourists [1, 2] or urban travel planners [3, 4] in the wider context of smart cities [5].

S. Evangelatos is also with the National & Kapodistrian University of Athens, Department of Physics, Athens, Greece

Multimodal travel planners are intelligent systems that search all the possible routes with combined transportation modes and return to the commuter personalized solutions characterized by accuracy and reliability. An agent-based simulator for multimodal travel mobility that simulates the movements of commuters on the different transport networks is presented in [6], focusing mainly in finding the right platform architecture and system's functionalities so as to take full advantage of the future Internet technologies.

Dynamic and on-demand service delivery is one of the main preconditions for successful business and thus service availability must not be compromized. Service interruption not only has negative effects in user experience in terms of QoS and QoE but can be also translated into revenue loss [7] by corrupting the business process. Even though understanding the properties of service availability is rather limited they are already widely used to build software architectures and enable complex business processes.

The paper is organized as follows: In Section II we briefly discuss the methodologies used for measuring service availability. Section III presents in detail the proposed system architecture and Section IV provides its availability analysis. In Section V the results from the availability analysis are presented. Related discussion in Section VI conclude this paper.

II. Related Work

Several quantitative, qualitative, and analytical methodologies have been used to assess service availability [8–10]. Quantitative assessment is based on real-time measurement and monitoring but it is difficult to be applied to services since they lack of adequate metrics. On the other hand, qualitative

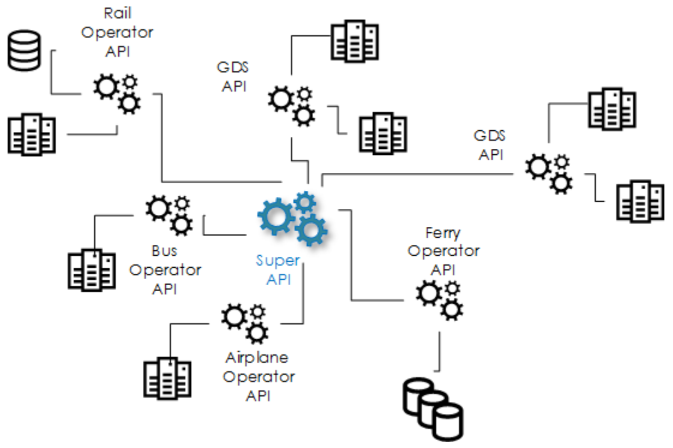

Fig. 1. The simplified view of the Super Travel API connectivity infrastructure

availability assessment is performed informally through questionnaires and interviews but as it can be easily understood, these results are easy to misinterpret, and sometimes difficult to compare. Analytical methods are used to model services and their fault behavior thus determining service availability. Up to now they have exhibited relatively low industry penetration due to scalability, complexity, and evolution problems.

Most of the approaches that address service availability employ mainly middleware concepts which can improve significantly the execution of existing services. As such, they are closer to the implementation rather than the design phase of fault-tolerant systems. In the vast majority of these approaches, services are neither explicitly modeled nor their availability is quantified [11].

Reliability block diagrams (RBD) [12] are a useful visualization framework for analyzing the service availability risks and behavior. The main drawbacks of RBDs are their inability to express services with varying states, dependent events, and non-series-parallel topologies. Fault trees analysis (FTA) [13] provides a compact, graphical, intuitive method to analyze system reliability/availability. Although this method has several advantages such as the provision of a systematic basis for quantitative analysis and the highlight of critical aspects of failure, it can not be adopted for large complex systems since the state space becomes quite large and difficult to handle.

Stochastic models are increasingly used to predict service performance and reliability. In [14] a continuous-time Markov chain (CTMC) formulation of composite services with failures was developed for the booking process of a travel agent. Even though this approach provided a closed-form expression for the overall reliability of travel services, the authors made several assumptions such as the lack of resource contention and the choice of specific distribution functions which are not always applicable to real-life situations.

III. The Super Travel API architecture

There are many solutions including journey planners across Europe, covering one or more transport modes and one or more countries. However, today's market availability is still a long way from providing travellers with the information needed to plan a door-to-door trip within Europe, regardless of the number of countries or modes of transport involved. The standardisation landscape is quite fragmented and continuously evolving, following technology advancements. The challenges that we try to address through this architecture are the establishment of common specifications for data formats, data exchange protocols and interconnection of existing solutions, without hampering technological innovation and the provision mechanisms that will tap into existing systems and promote interconnection of services.

Furthermore, to facilitate the deployment of multimodal travel planners with high availability services, we analyse a new service-oriented architecture, that together with the itinerary optimisation engine and the front-end constitute the proposed multimodal travel planner. Services inter-operate based on a formal definition that is independent of the underlying platform and programming language and their descriptions are used to advertise the service capabilities, interface, behaviour, and quality.

Our proposed connectivity infrastructure is depicted in Fig. 1 and is comprised by a semantics-based "Super Travel API" which provides the capability to bridge both internal IT as well as external services with existing protocols and standards, regardless of the underlying APIs of the various data sources. At the moment, in the travel industry there are some key APIs for the major services underpinning systems such as GDSs. However, the main problem hampering interoperability in the industry is that access to such APIs is hard-wired to the applications of travel agents and other system users constituting more difficult for APIs to provide new advanced services.

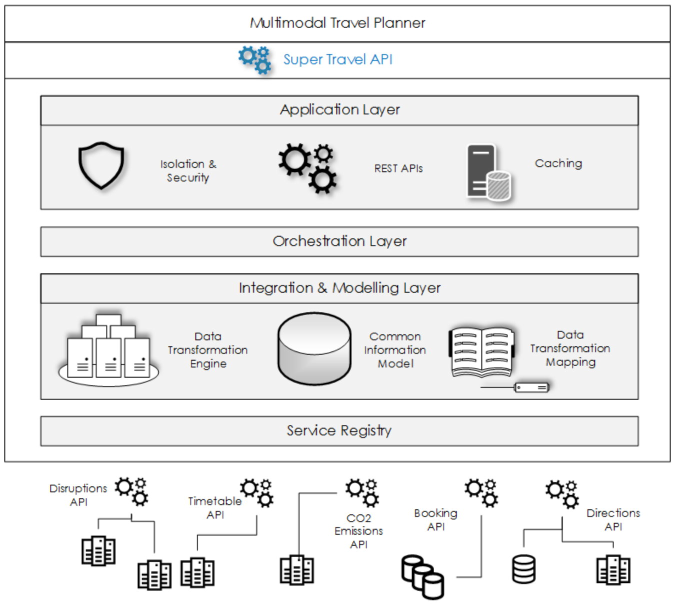

Fig. 2. The Super Travel API high-level architectural view

The Super Travel API architecture aims to address significant interoperability issues of the travel domain, in a way that it generates value for both data owners as well as service providers. Key architecture decisions include modularity, service composability and minimization of adoption costs for new entrants to the travel services ecosystem. As such, we split the system in four basic discrete modules, the Service Registry, the Integration and Modelling layer, the Orchestration layer and the Applications layer. Fig. 2 displays a high-level architectural view, depicting all modules and their respective positions.

The Service Registry is a web application serving as a semantically enriched index of travel related services and APIs in a multimodal context. This implies that both open and proprietary hierarchical data structures are annotated with formal semantic concepts and actions that facilitate automatic machine communication and interaction with minimal human intervention. Such an approach employs normalization by semantic annotation to facilitate service discoverability and therefore business expansion opportunities for participating entities. Naturally, this requires a common reference model, or language that enables mapping between heterogeneous data formats.

The latter resides in the Integration and Modelling layer, a module that can be considered a system by itself, in the sense that it receives schemata and data from the Service Registry and uses a Data Transformation interface to produce structures ready to be used and consumed throughout the ecosystem. The enabling technology for this component is the Common Information Model (CIM), a formal encoding of various aspects of multimodal travel, that enables expressive representation of travel-related services in an open, exportable format. The CIM is used as a basis model, therefore, by performing bidirectional transformations from all known schemata to it and by employing advanced software tooling, we can safely state that we can obtain implicit transformations between proprietary data structures, an achievement that can potentially have immense impact in the ever-changing world of on-line transport services which is in a state of fragmentation at this stage. The Integration and Modelling layer can be expanded upon request, to facilitate introduction of new participants in the ecosystem and employs machine learning to gradually automate the process by harnessing user validation and lexicographical ontologies such as WordNet [15] and ConceptNet [16].

The Orchestration layer, is a web application that delivers work flows by choreographing existing services and business logic. Using the abstraction that deals with commonly understood concepts instead of proprietary software implementations is vital in order to deliver a complete framework for integration and exposure of multimodal travel services. The Orchestration layer assists this effort by delivering tools that enable service composability and enrichment by dynamically discovering available data sources in the Service Registry, using programming techniques to establish work flows capable of achieving and the desired results.

The Applications layer is the collective output of the Super Travel API, by essentially being a generator of multiple,

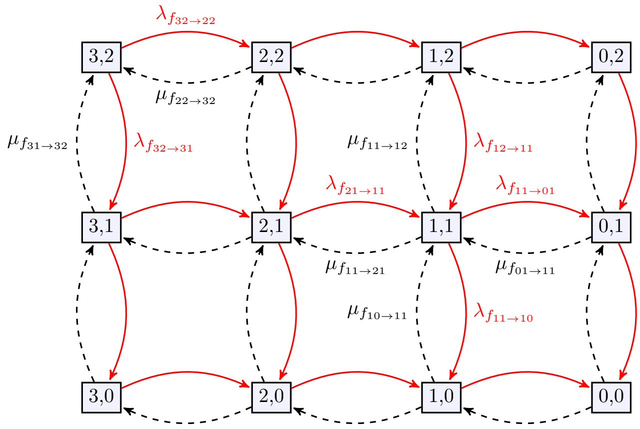

Fig. 3. The transition diagram for a continuous time Markov chain corresponding to the Super Travel API.

different REST APIs, suitable for consumption by authorized clients. The generated APIs are programmed on a suitable development environment, deployed and isolated, to enforce security, follow enterprise-grade specifications and minimize dependencies.

Overall, we consider the patterns and practices that the Super Travel API architecture uses to be crucial in the advancement of the state-of-the-art with regards to multimodal travel options, accessibility to everyone as well as business sustainability around travel and tourism in general. With the proposed architecture, we facilitate interoperability not by performing point to point integration but by exposing points in the internal processes and resources of each operation to expedite the integration from multiple third parties APIs resulting in a more flexible and cost effective integration. The implementation of the Super Travel API architecture has been realized using the model-driven software engineering platform *zAppDev* [17].

IV. Availability Analysis

In practice, every GDS has some information concerning the services provided by each transport operator such as booking and ticketing but most of the time it lacks of information regarding static data like schedules and infrastructure. This kind of data is being updated by the operator usually once or twice a year. In order to have a meaningful and realistic scenario for our analysis, we assume that in order for a travel planner to be fully functional per transport mode the functionality of at least one GDS or the respective's operator's API is required similar with a redundancy system.

One of the main practices to analyze these complex systems based on their services is through the Continuous Time Markov Chains (CTMCs) [18], which are stochastic processes evolving through a discrete set of states, some of them corresponding to functional configurations and the remaining of them to non-functional configurations. CTMCs provide a useful modelling formalism for evaluating the availability of

$$
\mathbf{Q} = \begin{bmatrix}
q_{32\to32} & q_{32\to31} & 0 & q_{32\to22} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
q_{31\to32} & q_{31\to31} & q_{31\to30} & 0 & q_{31\to21} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & q_{30\to31} & q_{30\to30} & 0 & q_{30\to20} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
q_{22\to32} & 0 & 0 & q_{22\to22} & q_{22\to21} & q_{22\to12} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & q_{21\to31} & 0 & q_{21\to22} & q_{21\to21} & q_{21\to20} & 0 & q_{22\to11} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & q_{20\to21} & q_{20\to20} & 0 & 0 & q_{20\to10} & 0 & 0 & 0 \\
0 & 0 & 0 & q_{12\to22} & 0 & 0 & q_{12\to12} & q_{12\to11} & 0 & q_{12\to02} & 0 & 0 \\
0 & 0 & 0 & 0 & q_{11\to21} & 0 & q_{11\to12} & q_{11\to11} & q_{11\to10} & 0 & q_{11\to01} & 0 \\
0 & 0 & 0 & 0 & 0 & q_{10\to20} & 0 & q_{10\to11} & q_{10\to10} & 0 & 0 & q_{10\to00} \\
0 & 0 & 0 & 0 & 0 & 0 & q_{02\to12} & 0 & 0 & q_{02\to02} & q_{02\to01} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & q_{01\to11} & 0 & q_{01\to02} & q_{01\to01} & q_{01\to00} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & q_{00\to10} & 0 & q_{00\to01} & q_{00\to00}
\end{bmatrix} \quad (1)
$$

complex systems since they can handle many of the interdependencies and dynamic relationships among the relative subsystems. These state-space models have also the capacity of handling different failure and repair behaviours that are more realistic in modern systems such as correlated failures and repair dependencies. A popular assumption made for the analysis of complex systems is that the time that the system remains in each state follows an exponential distribution.

In our case, let us consider that the chain evolves in a discrete state space $\Omega = \{1, 2, \dots, k\}$ where the l states correspond to functional configurations and the rest of them, i.e., $k - l$ to non-functional. A failure is described by a transition from a functional to a non-functional state, whereas a repair is represented by a transition from non-functional to functional state. Since these events can occur at any time, CTMCs are suitable for modelling the behaviour of the Super Travel API system. We assume that each travel service accessed through an API has its own exponentially distributed time to failure and time to repair with their respective rates λ_f and μ_f.

We denote $\mathrm{X}(t)$ the state of the Super Travel API at time $t,\ t > 0$ and $\mathbf{Q} = [q_{ij}]$ the infinitesimal generator (or transition rate) matrix with q_{ij} representing the transition rates from state i to j, for $i \neq j$ and $q_{ij} = -\sum_{j \neq i} q_{ij}$ for all the diagonal elements. Every state is described by the set of numbers of the APIs corresponding to the operators and the GDSs. In our scenario, in order to book a multimodal journey we combine three transport operators and two GDSs. The behaviour of the Super Travel API can be modelled as the irreducible CTMC, with its transition diagram shown in Fig. 3 where, according to our assumption, only the last state with the all-zero element (i.e., state $(0, 0)$) is classified as non-functional.

Additionally, the proposed CTMC has not any absorbing states since every state is possible to be left and it is irreducible, i.e., it is possible to get from any state to any other state at some finite time, even with multiple transitions. We have assumed that two or more APIs can not fail (or repaired) simultaneously.

The state probability at time t which is the probability that the system is in state j at time t is denoted as

$$P_j(t) = \mathbb{P}\left\{\mathrm{X}(t) = j\right\} \quad (2)$$

Then, the row vector $\mathbf{P}(t) = [P_1(t), \dots, P_n(t)]$ represents the transient state probability vector of the CTMC and its behaviour can be described by the following Kolmogorov differential equation:

$$\frac{\mathrm{d}\mathbf{P}(t)}{\mathrm{d}t} = \mathbf{P}(t)\mathbf{Q} \quad (3)$$

given the initial probability vector at ime $t = 0$. Let $\boldsymbol{\pi}$ be the steady-state (time invariant) probability vector with $\boldsymbol{\pi} = \lim_{t\to\infty} \mathbf{P}(t)$. Then, from the global balance conditions, the steady-state probabilities of all states can be found from the rate matrix $\mathbf{Q}$ in 1 by solving the equations system

$$\boldsymbol{\pi}\mathbf{Q} = \mathbf{0} \quad (4)$$

The equations in (4) are linearly dependent and in order to uniquely determine the steady-state solution we impose the following normalization condition

$$\sum_{i\in\Omega} \pi_i = 1 \quad (5)$$

According to our assumption, the Super Travel API is available as long as it is not in the state $(0, 0)$. Hence the steady-state availability, $\mathcal{A}_{\text{ss}}$, is given by

$$\mathcal{A}_{\text{ss}} = 1 - \pi_{0,0} \quad (6)$$

The elements depicted in the $\mathbf{Q}$-matrix correspond to states can be easily expressed from the transition diagram according to the corresponfing rates λ_f and μ_f but for reasons of completeness we provide the diagonal elements which as already stated before, can be calculated as $q_{ij} = -\sum\limits_{j\neq i} q_{ij}$.

$$
\begin{aligned}
q_{32\to32} &= -\left(\lambda_{f_{32\to22}} + \lambda_{f_{32\to31}}\right) \\
q_{31\to31} &= -\left(\mu_{f_{31\to32}} + \lambda_{f_{32\to30}} + \mu_{f_{31\to21}}\right) \\
q_{30\to30} &= -\left(\lambda_{f_{30\to20}} + \mu_{f_{30\to31}}\right) \\
q_{22\to22} &= -\left(\mu_{f_{22\to32}} + \lambda_{f_{22\to21}} + \lambda_{f_{22\to12}}\right) \\
q_{21\to21} &= -\left(\mu_{f_{21\to31}} + \mu_{f_{21\to22}} + \lambda_{f_{21\to11}} + \lambda_{f_{21\to22}}\right) \\
q_{20\to20} &= -\left(\mu_{f_{20\to30}} + \mu_{f_{20\to21}} + \lambda_{f_{20\to10}}\right) \\
q_{12\to12} &= -\left(\mu_{f_{12\to22}} + \lambda_{f_{12\to11}} + \lambda_{f_{12\to02}}\right) \\
q_{11\to11} &= -\left(\mu_{f_{11\to21}} + \mu_{f_{11\to12}} + \lambda_{f_{11\to01}} + \lambda_{f_{11\to10}}\right)
\end{aligned}
$$

$$q_{10\to10} = -\left(\mu_{f_{10\to20}} + \mu_{f_{10\to11}} + \lambda_{f_{10\to00}}\right)$$
$$q_{02\to02} = -\left(\lambda_{f_{02\to01}} + \mu_{f_{02\to12}}\right)$$
$$q_{01\to01} = -\left(\mu_{f_{01\to02}} + \mu_{f_{01\to11}} + \lambda_{f_{01\to00}}\right)$$
$$q_{00\to00} = -\left(\mu_{f_{00\to10}} + \mu_{f_{00\to01}}\right)$$

In case of further APIs addition, which can be translated as several state additions in the CTMC, the state space of the system grows exponentially. In spite of the symmetry of the resulting CTMC, a closed-form solution for this case is a complex and error prone task if not impossible [19] and it can only be solved either by approximation techniques [20] or numerically through simulations. Intuitively, the aforementioned case can be seen as a case of redundancy. For example, adding more than one API of an operator, increases significantly the possibility of the Super Travel API to find the required information about the specific transport mean by at least one of the available APIs.

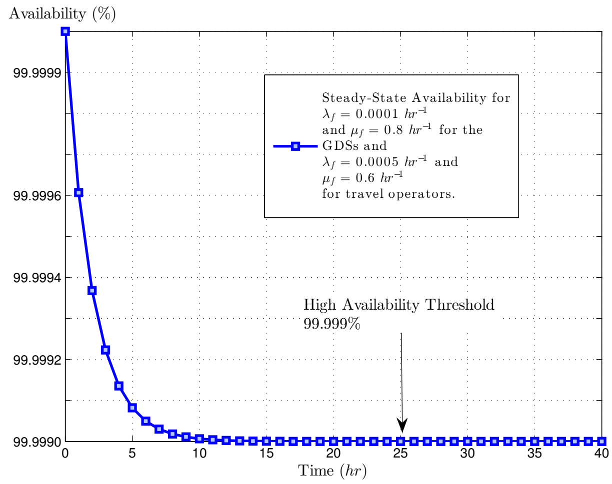

Fig. 4. Steady-state availability for the Super Travel API.

V. Validation Results

Travel APIs are expected to have a high level of robustness since there are payment and booking services involved that require reliable and secure transactions [21]. Some indicative values for the failure and repair rates of the APIs implemented in a test environment and measured directly from the Super API, for a proof of concept (PoC) travel planner, are $\lambda_f = 0.0001\ hr^{-1}$ and $\mu_f = 0.8\ hr^{-1}$ for the GDSs and $\lambda_f = 0.0005\ hr^{-1}$ and $\mu_f = 0.6\ hr^{-1}$ for travel operators. All of the commercial travel APIs are characterized by high availability reaching 100% at least regarding unplanned outages which are stressful events caused by human error, software problems or hardware failures.

The steady-state availability $\mathcal{A}_{\rm ss}$ has been calculated by solving Eq. (4) using the well-known Gaussian elimination algorithm which takes advantage of elementary matrix operations that preserve the rank of the matrix. In Fig. 4 the total availability of the Super Travel API is depicted according to the failure and repair rates of the connected APIs. As it can be readily seen, the steady-state availability exceeds 99.999% which is the threshold for many systems' high availability, or in other words, the Super Travel API's service downtime is less than 5.26 minutes per year.

The proposed architecture eliminates all single points of failure since the APIs involved in each service constitute a redundancy mechanism that, together with a failure component embedded in the architecture, achieves high availability assuring that a group of services will have a high level of operational performance.

VI. Conclusions

In this paper, we analysed the availability of travel services combined under a single aggregator namely, the Super API and we described an architecture for a multimodal travel planner which assist travellers in planning, booking and ticketing their journey and analyzed the total availability of the underlying infrastructure. The proposed travel planner which is the upper level of the Super API, is highly scalable since it can be implemented with the integration of several APIs with critical services, maintaining its high availability. As a future work, we will extend this analysis to incorporate planned and scheduled outages that are essential for routine system's maintenance or inspection. With the incorporation of additional travel APIs, even for the same geographical areas, the correlation degree between the Super Travel API components can change significantly leading to a more complex but intriguing connectivity structure and thus to a different Markov Chain model whose behavior we are planning to investigate.

Acknowledgments

This paper reflects only the authors' views; the European Union is not liable for any use that may be made of the information contained therein. The work reported is carried out by the EuTravel project, which has received funding from the European Union, under the Horizon 2020 programme (grant agreement No. 636148). For more details, please see http://www.eutravelproject.eu/.

References

[1] D. Gavalas, V. Kasapakis, C. Konstantopoulos, G. Pantziou, N. Vathis, C. Zaroliagis, "The eCOMPASS multimodal tourist tour planner," *Expert Systems with Applications*, Vol. 42(21), pp. 7303-7316, Elsevier, 2015.

[2] D. Gavalas et al., "Scenic Athens: A personalized scenic route planner for tourists," *IEEE Symposium on Computers and Communication*, pp. 1151–1156, Messina, 2016.

[3] A. Nuzzolo, C. Antonio and L. Rosati, "New Methods of Trip Planning for Travelers of Advanced Public Transport," *IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 13–18, Las Palmas, 2015.

[4] K. Rehrl, S. Bruntsch and H. J. Mentz, "Assisting Multimodal Travelers: Design and Prototypical Implementation of a Personal Travel Companion," in *IEEE*

Transactions on Intelligent Transportation Systems, Vol. 8(1), pp. 31–42, March 2007.
[5] L. Yu, D. Shao and H. Wu, "Next Generation of Journey Planner in a Smart City," *IEEE International Conference on Data Mining Workshop*, pp. 422–429, Atlantic City, NJ, 2015.
[6] M. Zargayouna, B. Zeddini, G. Scemama and A. Othman, "Simulating the impact of future Internet on multimodal mobility," *IEEE/ACS 11th International Conference on Computer Systems and Applications*, Doha, 2014, pp. 230-237.
[7] U. Franke, "Optimal IT Service Availability: Shorter Outages, or Fewer?," *IEEE Transactions on Network and Service Management*, Vol 9(1), pp. 22–33, March 2012.
[8] S. Distefano, C. Ghezzi, S. Guinea and R. Mirandola, "Dependability Assessment of Web Service Orchestrations," *IEEE Transactions on Reliability*, Vol. 63(3), pp. 689–705, Sept. 2014.
[9] H. Zheng, W. Zhao, J. Yang and A. Bouguettaya, "QoS Analysis for Web Service Compositions with Complex Structures," *IEEE Transactions on Services Computing*, Vol. 6(3), pp. 373–386, July-Sept. 2013.
[10] G. Zou, Q. Lu, Y. Chen, R. Huang, Y. Xu and Y. Xiang,"QoS-Aware Dynamic Composition of Web Services Using Numerical Temporal Planning," *IEEE Transactions on Services Computing*, Vol. 7(1), pp. 18–31, Jan.-March 2014.
[11] N. Milanovic and B. Milic, "Automatic Generation of Service Availability Models," in *IEEE Transactions on Services Computing*, Vol.4(1), pp. 56–69, Jan. 2011.
[12] R. Sahner, K. Trivedi, and A. Puliafito, *Performance and Reliability Analysis of Computer Systems,* Kluwer Academic Publishers, 2002.
[13] M. L. Shooman, *Probabilistic Reliability: an Engineering Approach*, McGraw-Hill, 1968.
[14] N. Sato and K. S. Trivedi, "Stochastic Modeling of Composite Web Services for Closed-form Analysis of their Performance and Reliability Bottlenecks," *Proceedings of the 5th International Conference on Service-Oriented Computing*, pp. 107–118, Springer-Verlag, 2007.
[15] G. Miller *WordNet: A Lexical Database for English*, Communications of the ACM, Vol. 38(11), pp. 39–41, 1995.
[16] H. Liu and P. Singh, *ConceptNet - A Practical Commonsense Reasoning Tool-Kit*, BT Technology Journal Vol. 22(4) 211-226, 2004.
[17] Powerful Automation Avoiding Platform Lock-in for Enterprise-grade Applications, http://www.zappdev.com/
[18] K. Trivedi, *Probability and Statistics with Reliability, Queuing and Computer Science Applications*, John Wiley and Sons, 2001.
[19] J. P. Katoen and I. S. Zapreev, "Simulation-Based CTMC Model Checking: An Empirical Evaluation," *6th IEEE International Conference on the Quantitative Evaluation of Systems*, pp. 31–40, Budapest, 2009.
[20] C. Baier, B. Haverkort, H. Hermanns and J. P. Katoen, "Model-checking Algorithms for Continuous-time Markov Chains," in *IEEE Transactions on Software Engineering*, Vol. 29(6), pp. 524–541, June 2003.
[21] M. Toeroe and F. Tam, *Service Availability: Principles and Practice*, John Wiley and Sons, 2012.