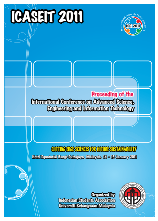
Proceeding of the International Conference on Advanced Science, Engineering and Information Technology 2011
Hotel Equatorial Bangi-Putrajaya, Malaysia, 14 - 15 January 2011
ISBN 978-983-42366-4-9



Multi-leg Searching by Adopting Graph-based Knowledge Representation

Siti Zarinah Mohd Yusof[#], Vijanth Sagayan Asirvadam*, Mohd Fadzil Hassan[+]

[#, +] *Computer and Information Sciences Department, *Electrical and Electronics Department*
Universiti Teknologi PETRONAS
Bandar Seri Iskandar, Tronoh, Perak, Malaysia
E-mail: [#]zarinah.yusof@gmail.com

***Abstract*— This research explores the development of multi-leg searching concept by adopting graph-based knowledge representation. The research is aimed at proposing a searching concept that is capable of providing advanced information through retrieving not only direct but continuous related information from a point. It applies maximal join concept to merge multiple information networks for supporting multi-leg searching process. Node and edge similarity concept are also applied to determine transit node and alternative edges of the same route. A working prototype of flight networks domain is developed to represent the overview of the research.**

***Keywords*— Graph-based Knowledge Representation, Graph Theory, Maximal Join, Node Similarity, Edge Similarity.**

I. Introduction

Internet is becoming a significant element in travel industry as it provides a vast communication network that links computer worldwide [1]. Airlines passengers could obtain information such as destination, flight schedules, price and discount, airlines bookings and other related services which are displayed in the Internet [1]. In an airline website, usually users could view destinations network or a routes map from a specific departure city (origin) as a pre-trip guidance to make flight ticket reservation.

In making reservation, users have few choices in selecting type of air journey that they intent to pursue. The choices available are one-way journey, round-trip or multi-city journey. One way journey is considered as a trip in a continuous direction from origin to the final destination without a return to the origin [1]. If users want to have a return, they could choose round trip journey. Both types of journey would only involve one-leg searching as the process only retrieves direct destination from a point to another point.

Multi-city or multi-leg journey is a trip from the origin to the outward destinations in an itinerary plan. Users are needed to select a number of destinations of this type. Multi-leg searching would ease the selection process. It would not only retrieve direct destination from an origin but a number of continuous destinations. The outcome would be very beneficial as people do not always know exactly where they could go. Before a number of continuos destinations could be retrieved, multiple flight connections from different airlines are needed to be unified in order to form broader, sufficient information.

Few literature surveys regarding existing flight connection systems and graph related areas are initiated. From the findings, most of the researches focus on producing optimal path for multi-city journey from given destinations instead of searching and proposing the destinations to users. This preliminary study brings a motivation to conduct a detailed research on multi-leg searching presented in this paper.

A multi-leg searching concept that is capable of providing advanced information through retrieving not only direct but continuous information from a point is discussed throughout the paper (i.e.: continuous destinations that users could go further from an origin for multi-city journey). The scope of information network is expanded to support multi-leg searching process. Common destinations and routes among multiple flight connections would be figured out to efficiently support the concept.

Further details will be explained throughout this paper. Related works are discussed in section II. The methodology used is discussed in section III. Explanation on the results is discussed in section IV. Within section IV, a sample prototype is shown to represent the overview of the searching concept. As the research is still progressing, future directions will be discussed in section V. Lastly, the paper will be ended by a conclusion in section VI.

II. Related Work

Related work has been done in flight connection systems and applied graph-based knowledge representation concept. Detailed explanations of both areas are as follows:

A. Flight Connection Systems

There exist many researches that are focusing on air flight schedule or itinerary planner. In [2], the authors used a mixed initiative system, a kind of dialog established between the user and the system. The system makes suggestion even if no one satisfies completely the user criteria (called partial or soft constraint) satisfaction. This is an alternative to most of existing online booking engines use a sequential model. In sequential model, users enter the search criteria, clicks on the search button and waits until the results are displayed in a list format, which the involved processes are very tedious.

In [3], the authors presented a budget travel planning that leverages state-of-the-art semantic web technology to aggregate related web resources and reason on them to provide automated travel planning service. This research is proposed to solve problem in searching for all related information to make a travel plan. In [4], the author proposed a development of a multi-agent system that will allow the creation of itineraries through the given tourist's travel preferences including travel period, number of people, zones and tourist service of interest. This work is presented to design a good itinerary as it is essential if it is expected to enjoy the travel.

In [5], the author presented a flight meta-search engine that is based on Metamorph framework. Metamorph provides mechanisms to model web forms together with the interactions which are needed to fulfil a request and can generate interaction sequences that pose queries using these web forms and collect the result. It shows how data can be extracted from web forms (rather than the data behind the web forms) to generate a graph of flight connections between cities. The authors implemented a system that uses a flight connection graph and a query planner to limit the number of requests by checking only the most promising paths.

From these surveys, it is understood that air flight passengers would have a number of preferences or criteria in making reservation. Besides, air flight passengers require extra information to ease them in making a travel plan. Although there are huge amount of information regarding flight connection in the internet, it is placed in a distributed manner. Centralizing the information would simplify the searching process.

B. Graph-based Knowledge Representation

As this research works with lots of information, graph-based knowledge representation would be one of the research areas that could be the best way to represent multi-level searching concept. Ryszard Raban emphasized the advantage of using graphical logic in his work [6]. He stated that the graphs are not used as just another graphical representation of information system requirements, but the full power of this graphical logic system has been employed to fully and precisely capture type definitions, referential integrity constraints and global constraints [6].

Maximal join has been seen as a potential concept in graph-based knowledge representation that would meet the objective of expanding the information network. Simon Polovina explained that the joining of graphs facilitates inference because more projections can be made into bigger graphs [7]. In addition, Simon Polovina mentioned in his work that maximal join, which extends the notion of join in SQL in database systems, defines the optimal method by which graphs are joined [7].

Michel Chein and Marie-Laure Mugneir described the effect of maximal join operation which is to maximally join, or merge, connected subgraphs of two graphs [8]. In addition, Simon Polovina mentioned that maximal join occurs when graphs are joined on the most common, or maximally extended, projection [7]. Dickson Lukose in his work explained that two graphs can be joined on maximally extended common projections. Multiple graphs could result if there is more than one maximally extended common projection [9].

In [9] and [10], maximal join has also been applied using tool as a conceptual graph manipulation functions or operator. Maximal join has been widely used in merging graph. This method is seen as a good method to be applied in the research.

III. Methodology

There are three main elements that would be used throughout the research- Maximal Join, Node Similarity and Edge Similarity. The methodology used to pursue this research are broadening information network using Maximal Join and finding node and edge similarity. A sample prototype has been developed to represent the overview of the searching concept.

A. Broadening Information Network Using Maximal Join

In order to provide multi-leg searching, the information network involved should be connected with sufficient information. An information network could be expanded by merging it with another network. Maximal join operation is applied to merge those networks.

As referred to [8], maximal join operation would maximally join, or merge, connected subgraphs of two graphs. According to Michel Chein and Marie-Laure Mugnier, the simplest way of joining two graphs is by using external join operation which consists of merging two concept nodes of two disjoint graphs [8]. Before maximal join operation is performed, condition for merging concept nodes and a condition for merging relation nodes have to be defined first as well as the strategies for exploring the graph.

B. Finding Node and Edge Similarity

Node and edge similarity are necessary to be figured out. Node similarity would determine point to merge disjoint graphs and as a point of route transit; transit from one graph to another graph. Meanwhile, edge similarity would determine number of alternatives of the same route. For instance, assume that node a in graph X is similar to node b in graph Y. As the nodes are similar, these nodes would be the merging point of graph X and Y. The nodes would also be the transit point from graph X to graph Y. Assume that edge m in graph I and edge n in graph J are similar, these edges represent two available alternatives of going through the same route. In order to determine node and edge similarity, labeled graphs concept plays a vital role.

C. Prototype Development

In parallel with the literature surveys and analysis, a sample prototype is being developed to represent the overview of proposed searching concept. A case study is selected and data sample used in the prototype are taken from prominent website within the area of the case study. The developed prototype is shown in section IV. Case study used for the prototype is destination networks of Low Cost Carriers (LCC) in Malaysia such as Air Asia and Fire Fly. The data sample are taken form their respective websites.

IV. RESULT AND DISCUSSIONS

Graph-based knowledge representation is used to explain the concept of maximal join, node similarity and edge similarity. A simple prototype is presented at the end of this section to represent the overview of the searching concept. The networks are represented using graph to show how the similarities are being measured and how the merging process is happened. Based on the case study, two flight networks of different LCC would be represented as two disjoint graphs. Fig. 1 shows graph *M* that represents destinations provided by Air Asia adapted from [12]. In the other hand, Fig. 2 shows graph *N* that represent destinations provided by Fire Fly adapted from [13]. Nodes and edges in both graphs are labeled with information that would be used in determining the similarities. Label of nodes is shown in Table 1 meanwhile edges are labeled with duration of journey.

A. Node and Edges Similarity

The similarity of node and edge could be determined through side information assigned to each node and edge or in specific term, it is called label. According to [11], two nodes will be similar if they have similar in/out neighbors. Based on Fig. 1 and 2, it is shown that many similar nodes are available as these nodes have the same neighbours and airport codes. Node m_6 and n_6 also could be considered as similar as both nodes have the most common neighbours. In addition, label of nodes in Table 1 shows that both nodes have the same city name, Kuala Lumpur. Therefore, these nodes would be one of the transit points available between graph *M* and graph *N*. These two nodes best at presenting transit node concept as it enables passengers go to n_7 from graph *M*.

It is also shown that similar nodes found in the graphs are connected by edges. These connecting edges are considered as similar thus provide alternatives of going through the same route. The best example that could be extracted from Fig. 1 and Fig. 2 is edge a_1 and edge b_1. The labels of journey duration are the same and these edges are connecting similar node between m_1/n_1 and m_4/n_4. From the graph, it is best to describe that there are two available service provider alternatives if users want to take a route from LGK to PEN, either by Air Asia or Fire Fly.

B. Maximal Join

In order to broaden main information network, disjoint networks need to be merged. However, the merging point needs to be figured out first. Maximal join occurs when graphs are joined on the most common, or maximally extended, projection. Based on the previous discussion, node m_6 and node n_6 have the most common neighbours. These nodes are also considered to have the most common or maximally extended, projection. Therefore, node m_6 and node n_6 would be best merging point that unified graph *M* and graph *N*. Fig. 3 shows the merged graphs of graph *M* and *N*. The green nodes represent similar nodes and the red dotted lines represent similar edges.

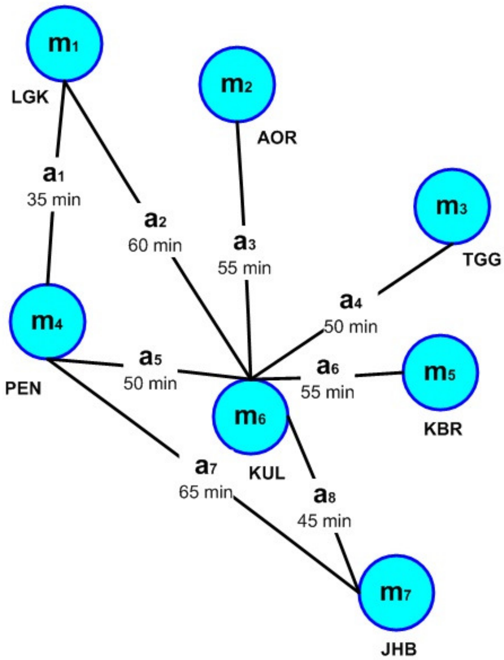


Fig. 1 Graph *M*: Air Asia Flight Network

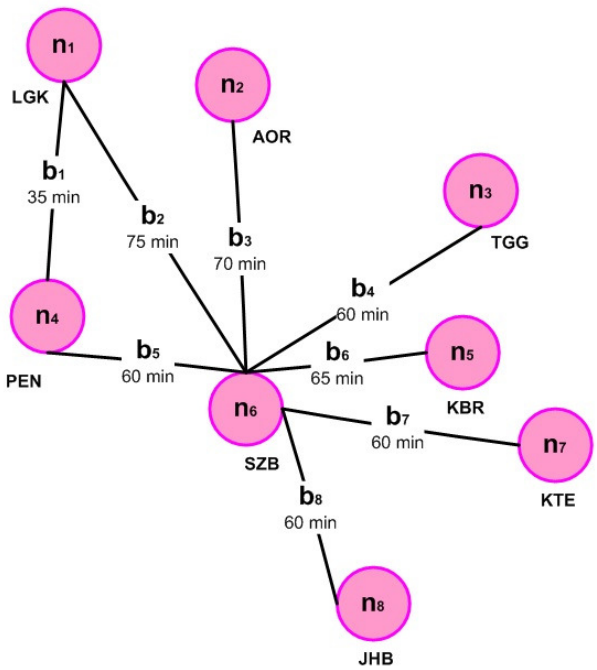


Fig. 2 Graph *N*: Fire Fly Flight Network

TABLE 1

LABELS OF NODES

	Labels of Graph M(m,a)		Label of Graph N(n,b)	
	airport_codes	city_name	airport_codes	city_name
1	LGK	Langkawi	LGK	Langkawi
2	AOR	Alor Setar	AOR	Alor Setar
3	TGG	Kuala Terangganu	TGG	Kuala Terangganu
4	PEN	Penang	PEN	Penang
5	KBR	Kota Bharu	KBR	Kota Bharu
6	KUL	Kuala Lumpur	SZB	Kuala Lumpur
7	JHB	Johor Bahru	KTE	Kerteh
8			JHB	Johor Bahru

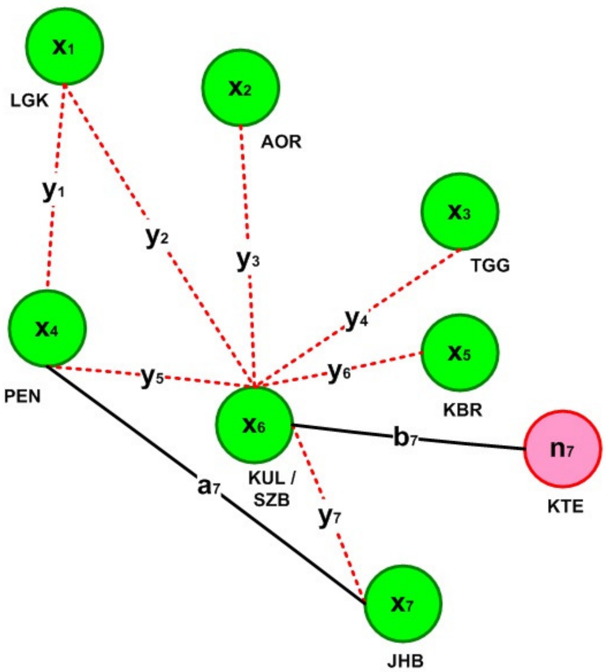


Fig. 3 Merged Graphs

C. Prototype

In delivering the overview of the research concept, a simple prototype has been developed. PHP: Hypertext Preprocessor has been chosen as the best server-side scripting language in developing it. As it supports many databases including mySQL, the prototype could be easily connected to PHPmyAdmin which is being installed together with PHP in XAMPP software. Data sample used in this prototype is taken from two LCC website - Air Asia and Fire Fly. These data is then combined into a database which is presented as merged graphs. Fig. 4 shows the search page snapshot of the prototype. Current development only includes from first leg till third leg searching presented as Number of Leg in Fig. 4. First destination would be one of the constraints for the searching process. From the first destination, this prototype would suggest further continuous destinations that users could travel. The searching result would be filtered according to assigned constraints which would not be discussed in this paper as the paper is only focusing on the graph-representation part.

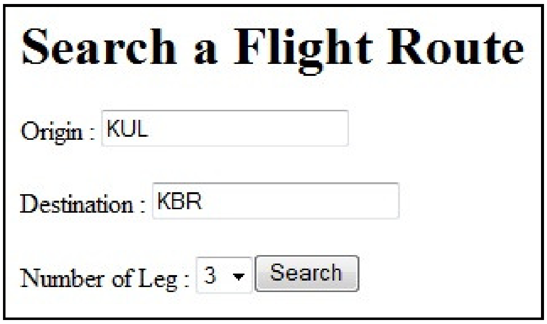


Fig. 4 Search Page Snapshot

Table 2 shown below is the result of multi-leg searching process which would be produced when users click on the Search button in Fig. 4. Multi-leg searching concept is represented by the produced result for first, second and third leg searching. From first destination, Kota Bharu (KBR), user would identify next destination and continuous destinations that they could travel further. As shown in the table, there are columns named Cycle to First Destination and Cycle to Origin. These columns are another two assigned constraints to mark the destinations which have a return to first destination or origin. As described before, SZB and KUL are considered as similar nodes which are also considered as the merging and transit point between Air Asia and Fire Fly graphs. If users from KBR return to SZB, it means that the users are going back to Kuala Lumpur. However, users could have a transit from SZB to travel to KTE using Fire Fly which the route is not provided by Air Asia. If users travel to PEN from SZB before going to LGK, users have the alternatives to choose whether they want to fly to LGK using Air Asia or Fire Fly as both LCC provide the same route.

V. FUTURE DIRECTIONS

For future works, detailed graph-based knowledge representation with a combination of constraints will be figured out. In parallel with the research progress, the prototype development also needs to be continued and the searching process needs to be enhanced by adopting constraints in order to produce the most relevant and efficient results to users especially in suggesting destinations that follow optimal path concept.

VI. CONCLUSIONS

In supporting multi-leg searching concept, a broader information network is needed. Graph merging is applied in this research to expand the information scope. Maximal join provide a way of doing it so through finding similar node from disjoint graphs which later the node will be the merging point. In addition, node and edge similarity is also being applied to extract node transit and alternative of the same route from the graphs. This information would be very beneficial for airlines or travel industry if it is presented well using graph-based knowledge representation.

TABLE 2

MULTI-LEG SEARCHING OUTPUT

Chosen Route

Origin	Destination	Service Provider
KUL	KBR	Air Asia

First Leg Searching

Origin	Destination	Service Provider	Cycle to Origin
KBR	SZB	Fire Fly	0
KBR	*KUL*	*Air Asia*	*1*

Second Leg Searching

Origin	Destination	Service Provider	Cycle to First Destination	Cycle to Origin
SZB	JHB	Fire Fly	0	0
SZB	KTE	Fire Fly	0	0
SZB	TGG	Fire Fly	0	0
SZB	*KBR*	*Fire Fly*	*1*	*0*
SZB	LGK	Fire Fly	0	0
SZB	AOR	Fire Fly	0	0
SZB	PEN	Fire Fly	0	0

Third Leg Searching

Origin	Destination	Service Provider	Cycle to First Destination	Cycle to Origin
JHB	SZB	Fire Fly	0	0
JHB	PEN	Air Asia	0	0
JHB	*KUL*	*Air Asia*	*0*	*1*
KTE	SZB	Fire Fly	0	0
TGG	SZB	Fire Fly	0	0
TGG	*KUL*	*Air Asia*	*0*	*1*
LGK	PEN	Fire Fly	0	0
LGK	SZB	Fire Fly	0	0
LGK	PEN	Air Asia	0	0
LGK	*KUL*	*Air Asia*	*0*	*1*
AOR	SZB	Fire Fly	0	0
AOR	*KUL*	*Air Asia*	*0*	*1*
PEN	SZB	Fire Fly	0	0
PEN	LGK	Fire Fly	0	0
PEN	LGK	Air Asia	0	0
PEN	*KUL*	*Air Asia*	*0*	*1*
PEN	JHB	Air Asia	0	0

REFERENCES

[1] *J. Semer-Purzycki, A Practical Guide to Fares and Ticketing, USA: Delmar, 2001.*

[2] *M. Torrens, P. Hertzog, L. Samson and B. Faltings, "Reality: A Scalable Intelligent Travel Planner", ACM, 2003.*

[3] *Y. Yang, S. Tang, Y. Xu, and Y. Xu, "A Budget Travel Planning System Using Ontologies for Web Information Aggregation", IEEE, 2006.*

[4] *M. Valdes and C. Cubillos, "A Multiagent System for Touristic Travel Planning", IEEE, 2009.*

[5] *B. Krupl and R. Baumgartner, "A Flight Meta-Search Engine with Metamorph", ACM, 2009.*

[6] *R. Raban, "Information Systems Modeling with CGs Logic", unpublished.*

[7] *S. Polovina, "Combining Graph", An Introduction to Conceptual Graphs, Berlin Heidelberg: Springer-Verlag, 2007.*

[8] *M. Chein and M.L Mugnier, Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs, London: Springer-Verlag, 2009, ch. 8.*

[9] *Lukose, "CGKEE: Conceptual Graph Knowledge Engineering Environment", unpublished.*

[10] *W. Mineau, "The Extensional Semantics of the Conceptual Graph Formalism", unpublished*

[11] *L. Ninove, "Graph Similarity Algorithms", Mathematical Engineering Department, Universit ́e catholique de Louvain, 2005, unpublished.*

[12] *Air Asia Flight Schedule, Air Asia Sdn. Bhd., retrieved on 18 November 2010, Avalable: www.airasia.com/iwov-resources/my/common/pdf/AirAsia/flightschedule/fullFlightScheduleRepot.pdf*

[13] *FireFly Schedule, FlyFirefly Sdn. Bhd., retrieved on 18 November 2010, Available: www.fireflyz.com.my/flights/schedule*