A Genetic Algorithm Based on Complex Networks Theory for the Management of Airline Route Networks

Xiao-Bing Hu and Ezequiel Di Paolo

Abstract Airline companies need to organize and manage their route networks in a more cost-efficient and reliable way, in order to cope with increasing customer demands and market changes. This paper attempts to apply complex network concepts and techniques to model airline route networks, and the focus is then put on how to develop an effective and efficient Genetic Algorithm (GA) to optimize airline route networks in terms of certain network properties which are identified to have crucial roles to play in making airline route networks cost-efficient and reliable. The chromosome structure in the proposed GA is based on complex network modelling, and as a result, effective evolutionary operators, particularly a highly efficient uniform crossover operator, are developed. The results demonstrate that the reported GA has a good potential to improve the topology of airline route networks in terms of network properties of interest such as operating costs and network robustness.

1 Introduction

Every airline company needs to organize its services to cover a set of cities of interest by providing either direct or indirect flight connections. An airline route network is a complete set of all direct flights provided by the company to cover its targeted cities. In the past two decades, the aviation traffic volume around the world has kept soaring up, and the competition between airline companies has become more and more fierce [1]. To survive and/or to make more profits, an action the

X.-B. Hu
Centre for Computational Neuroscience and Robotics, University of Sussex
xiaobing.hu@sussex.ac.uk

E.D. Paolo
Centre for Computational Neuroscience and Robotics, University of Sussex
ezequiel@sussex.ac.uk

X.-B. Hu and E.D. Paolo: *A Genetic Algorithm Based on Complex Networks Theory for the Management of Airline Route Networks*, Studies in Computational Intelligence (SCI) **129**, 495–505 (2008)
www.springerlink.com

airline companies have to take is to reorganize their route networks in a more cost-efficient and reliable way. Particularly, after the deregulation of the passenger aviation market at the end of last century, many "trunkline"-carriers took advantage of the possibilities of the liberalised market and reorganised their networks from "point-to-point" into "hub-and-spoke" topologies [2–4]. On the other hand, some new or recently started airline companies continued operating "point-to-point" networks on a low-cost, no-frill, low-price basis. Besides changing the fundamental structure from point-to-point to hub-and-spoke topologies, airline route networks can somehow carry growing traffic volumes by other means such as using larger airplanes, increasing frequencies on existing routes, and adding new routes [6].

Modelling airline route networks is a crucial step for managing such networks, and it has attracted attention from many researchers [3, 5, 7, 8]. This paper will shed some light on the modelling of airline route networks with complex network techniques. More than modelling, the focus of the paper is to develop an effective and efficient genetic algorithm (GA) to optimize the topology of airline route networks, which is a major concern of airline companies. Complex networks, i.e. networks whose structure is irregular, complex and dynamically evolving in time, are all around us in our daily life [9]. As a recently developed mathematical framework, complex network theory has a good potential for systematically studying airline route networks. Actually, one can consider that, the two main categories of airline route networks, point-to-point and hub-and-spoke networks, are engineering examples of small-world network and scale-free network, respectively. Therefore, the corresponding concepts, measures and algorithms developed in complex network theory are useful in the systematic study of airline route networks. The first objective of this paper is to introduce some complex network properties suitable for the modelling of airline route networks. Based on these ideas, the second objective is to develop an effective and efficient algorithm to optimize the topology of airline route networks. As large-scale parallel stochastic search and optimization algorithms, GAs, if properly designed, have the capability of producing high quality solutions to the optimization of network structures [10–12]. The design of evolutionary operators, i.e., mutation and crossover, is particularly crucial in the successful implementations of GAs for such problems.

2 Problem Formulation

2.1 Network Modelling and Properties

In the framework of graph theory, an airline route network can be represented as an undirected graph $G = (\mathscr{N}, \mathscr{L})$, which consists of two sets $\mathscr{N}$ and $\mathscr{L}$, such that $\mathscr{N} \neq \emptyset$ and $\mathscr{L}$ is a set of unordered pairs of elements of $\mathscr{N}$. The elements of $\mathscr{N} \equiv \{n_1, n_2, \ldots, n_N\}$ are the nodes of the graph G, i.e., the cities linked by the airline route network, while the elements of $\mathscr{L} \equiv \{l_1, l_2, \ldots, l_L\}$ are its links, i.e., if there is a link between two cities, it means the airline provides direct flight service between these two cities. The number of elements in $\mathscr{N}$ and $\mathscr{L}$ are denoted by N and L, respectively. A node is referred to by its order i in the set $\mathscr{N}$, and each link is defined by a couple of nodes i and j, and is denoted as (i, j) or $l_{i,j}$. A graph of airline route network $G(N,L) = (\mathscr{N}, \mathscr{L})$ can be completely described by the adjacency matrix $\mathscr{A}$, a $N \times N$ square matrix whose entry $a_{ij}, (i, j = 1, \ldots, N)$ is equal to 1 when the link $l_{i,j}$ exists, and zero otherwise.

Degree and degree distribution. The degree $p_D(i)$ of a node i is the number of links incident with the node, and is defined in terms of the adjacency matrix $\mathscr{A}$ as

$$p_D(i) = \sum_{j=1}^{N} a_{i,j}. \tag{1}$$

In the airline route network, the degree $p_D(i)$ of a node i indicates that, starting from the city i, to how many other cities the airline provides direct flight services. The degree distribution $P_{DD}(k)$ is defined as the probability that a node chosen uniformly at random has degree k, i.e.,

$$P_{DD}(k) = \frac{1}{N} \sum_{i=1}^{N} s_i, \quad s_i = \begin{cases} 1, & p_D(i) = k \\ 0, & p_D(i) \neq k \end{cases} \tag{2}$$

The degree distribution $P_{DD}(k)$ can be used to identify whether a given airline route network is a point-to-point network or a hub-and-spoke network. One can reasonably expect that a hub city will have a largest node degree.

Shortest path in terms of flight distance and number of flight changes. Supposing $p_{SFD}(i, j)$ is the shortest flight distance from city i to city j, then the average shortest flight distance of the airline route network is

$$P_{ASFD} = \frac{1}{N(N-1)} \sum_{i,j \in [1,\ldots,n], i \neq j} p_{SFD}(i, j) \tag{3}$$

A well designed airline route network should have a relatively small average shortest path length in order to shorten flight times and therefore to cut operating costs. As is well known, the cost and time for changing flights at airports are not cheap. Therefore, if direct flight between two cities is out of the options, the least flight changes should be the first choice to both airlines and passengers. Let $p_{LFC}(i, j)$ denote the least flight changes required when travelling from city i to city j. Then the average least flight changes is

$$P_{ALFC} = \frac{1}{N(N-1)} \sum_{i,j \in [1,\ldots,n], i \neq j} p_{LFC}(i, j) \tag{4}$$

Robustness. Robustness is used to measure the importance of each individual city and route segment when the reliability/redundancy of a given airline route network is concerned. The closure of airport(s) and/or route segment(s) usually has a primary impact on the flight services which need to stop by the closed airport(s) or to go through the closed route segment(s). Each airline company needs to assess the robustness against closure of airport(s) and/or route segment(s) in its route network. That is, when certain airport(s) or route segment(s) is (are) closed, how easily can the affected flight service(s) be replaced by unaffected flight services provided by the same airlines? In this paper, we define two kinds of probabilities to assess the robustness of airline route networks, one in terms of nodes (airports), and the other in terms of links (route segments). In the first definition, $P_{NP}(n,m)$ is the probability that, after n random airports are closed, there exists a smallest sub-network which contains m airports. In the second definition, $P_{LP}(n,m)$ is the probability that, after n random route segments are closed, there exists a smallest sub-network which contains m airports. If, after a random closure of airports or route segments, the rest of the network still remains one connected graph, we say the smallest sub-network has 0 airports. Then $P_{NP}(n,0)$ and $P_{LP}(n,0)$ can be used to assess how robust are the flight services provided by the airlines against a random closure of n airports or route segments. In our study $P_{NP}(1,0)$ and $P_{LP}(1,0)$ are used to define node robustness and link robustness in order to measure the contribution of each node and link in the network reliability/redundancy. Suppose an original network exhibits network robustness $P_{NP}(1,0)$ and $P_{LP}(1,0)$. If a node or a link i is removed and the resulting new network has network robustness $\bar{P}_{NP,i}(1,0)$ and $\bar{P}_{LP,i}(1,0)$, then the robustness of node i and the robustness of link i are

$$P_{NP}(i) = P_{NP}(1,0) - \bar{P}_{NP,i}(1,0), \tag{5}$$

$$P_{LP}(i) = P_{LP}(1,0) - \bar{P}_{LP,i}(1,0), \tag{6}$$

respectively. Then the average node robustness and average link robustness are

$$P_{ANR} = \frac{1}{N}\sum P_{NP}(i), \quad P_{ALR} = \frac{1}{L}\sum P_{LP}(i). \tag{7}$$

2.2 Optimization of Airline Route Networks

There are two scenarios in the optimization of airline route networks, which represent different resource requirements and different extents to which the original airline route network will be modified. Consequently, we have two optimization problems to define as following.

Free-optimization. In this scenario, it is assumed that, for a given set of cities (N is therefore fixed), the airline company has enough resources, e.g., aircraft and staff, to run any network whose L is subject to an upper bound $L_{UB} \leq N \times (N-1)/2$, and it is willing to discard its old network completely and switch to a brand new one. L_{UB} roughly defines the maximum resources the airline company can deploy. Therefore, the objective of optimization is to find a route network which connects all given cities at the lowest cost in terms of a specific objective function. In this paper, the basic objective function is constructed as following based on average least flight changes, shortest flight distance, node robustness, link robustness, and/or degree distribution

$$J_1 = \alpha_1 P_{ALFC} + \alpha_2 P_{ASFD} + \alpha_3 P_{ANR} + \alpha_4 P_{ALR} + \alpha_5 E_{DD} \quad (8)$$

$$E_{DD} = \sum_{i=1}^{N-1} (\max(\tau(i), \mid P_{DD}(i) - D_{DD}(i) \mid) - \tau(i))^2 \quad (9)$$

where $\alpha_i, i = 1, \ldots, 5$, are weights which determine the contribution of each network property to the objective function, D_{DD} is a desirable curve of degree distribution, and $\tau(i)$ defines a tolerable zone for the gap between P_{DD} and D_{DD}. If the deviation of P_{DD} from D_{DD} at point i is beyond the tolerable zone, i.e., $\mid P_{DD}(i) - D_{DD}(i) \mid > \tau(i)$, it will be penalized.

Basically, more links can reduce the absolute values of P_{ALFC}, P_{ASFD}, P_{ANR}, P_{ALR} and E_{DD}, but the operation costs may soar up sharply if too many direct flights between medium and small airports are provided. Therefore, although the airline company has enough resources to run a network with up to L_{UB} links, it may prefer an L as small as possible. Here we introduce a threshold value L_S, and any $L > L_S$ will be punished accordingly as following

$$J_2 = J_1 + \beta \max(0, L - L_S) \quad (10)$$

where β is a penalty weight. The free-optimization problem can then be formulated mathematically as

$$\min_{L,\{l_1,\ldots,l_L\}} J_2, \quad N - 1 \leq L \leq L_{UB}. \quad (11)$$

Free-optimization is useful for newly established airline companies to organize their networks and services.

Constrained-optimization. For existing airline companies, a radical change to their route networks is rarely the case, but a gradual modification is more preferable. In this scenario, most resources are occupied, and there is limited flexibility in redeploying them. Without losing the generality, suppose we can remove L_R old links from the original network, add in L_N new links, and L_R and L_N must satisfy the following constraints

$$\underline{L}_R \leq L_R \leq \overline{L}_R, \quad (12)$$

$$\underline{L}_N \leq L_N \leq \overline{L}_N, \quad (13)$$

where $\underline{L}_R$ and $\overline{L}_R$ are lower bound and upper bound for L_R, and $\underline{L}_N$ and $\overline{L}_N$ for L_N. If $\underline{L}_R = \underline{L}_N = 0$, $\overline{L}_R = L_O$ (the number of links in the original network) and $\overline{L}_N = L_{UB}$, then the constrained optimization problem is relaxed into the free-optimization problem as discussed before.

Suppose the operation costs are not sensitive to an L_N within the range $[\underline{L}_N, \overline{L}_N]$. Then we do not need to penalize L, which equals $(L_O + L_N - L_R)$ and J_2 is not necessary. Therefore, the constrained optimization problem can be formulated as

$$\min_{L_R, L_N, \{l_{R,1},\ldots,l_{R,L_R}\}, \{l_{N,1},\ldots,l_{N,L_N}\}} J_1 \tag{14}$$

subject to Constraints (12) and (13), where $l_{R,i}$, $i = 1, \ldots, L_R$ are old links that need to be removed, and $l_{N,i}$, $i = 1, \ldots, L_N$ are new links that need to be added.

3 A GA for Airline Route Networks

Here we use the adjacency matrix to design our chromosomes. For an airline route network with N nodes and L links, the associated adjacency matrix M_A is an $N \times N$ 0-1-valued symmetric matrix, where the entry $M_A(i, j) = 1$ means a link exists between node i and node j. To make our GA more computer-memory-efficient, we take advantage of the symmetry of M_A, and construct our chromosome as a 0-1-valued vector composing of $N \times (N-1)/2$ elements, which we call genes. Gene 1 to gene $(N-1)$ come from $M_A(1,2)$ to $M_A(1,N)$ in order, and gene $((N-1)+(N-i+1)) \times i/2+1$ to gene $((N-1)+(N-i+1)) \times i/2+(N-i)$ from $M_A(i,i+1)$ to $M_A(i,N)$ for $i = 1, \ldots, N-1$. In this way, all connectivity information in a given network is encoded in the associated chromosome, as illustrated in Fig. 1.

However, the above 0-1-valued basic chromosome structure is only suitable for the free-optimization problem. In the constrained-optimization scenario, existing links in the original network need to be distinguished from newly added links, and the removal of existing links should also be separated from the removal of newly added links, such that any modification to the original network can be traced and Constraints (12) and (13) can be checked against. To this end, we allow the genes in the basic chromosome structure to take values between 0 and 3. In the chromosome related to the original network, which is called seed chromosome in this paper, only 0 and 3 are used to set up genes: gene with value of 3 is related to an existing link, while 0 means no link. By randomly introducing changes to its genes, the seed chromosome is used to set up the first generation of chromosomes. If a new link is added, the associated gene changes from 0 to 1, while if an existing link is removed, the associated gene changes from 3 to 2. After a random initialization based on the seed chromosome, the resulting new chromosome may have all values from 0 to 3, as illustrated in Fig. 1(d). No matter what operation is carried out on a gene in a new chromosome, its value only changes between 0 and 1, or between 2 and 3, depending on its initialized value based on the seed chromosome.

A feasible solution in airline route networks should satisfy certain constraints such as (12) and (13). Consequently, the associated feasible chromosome must satisfy the following constraints on the genotypic level:

$$\sum g(i) \leq L_{UB} \tag{15}$$

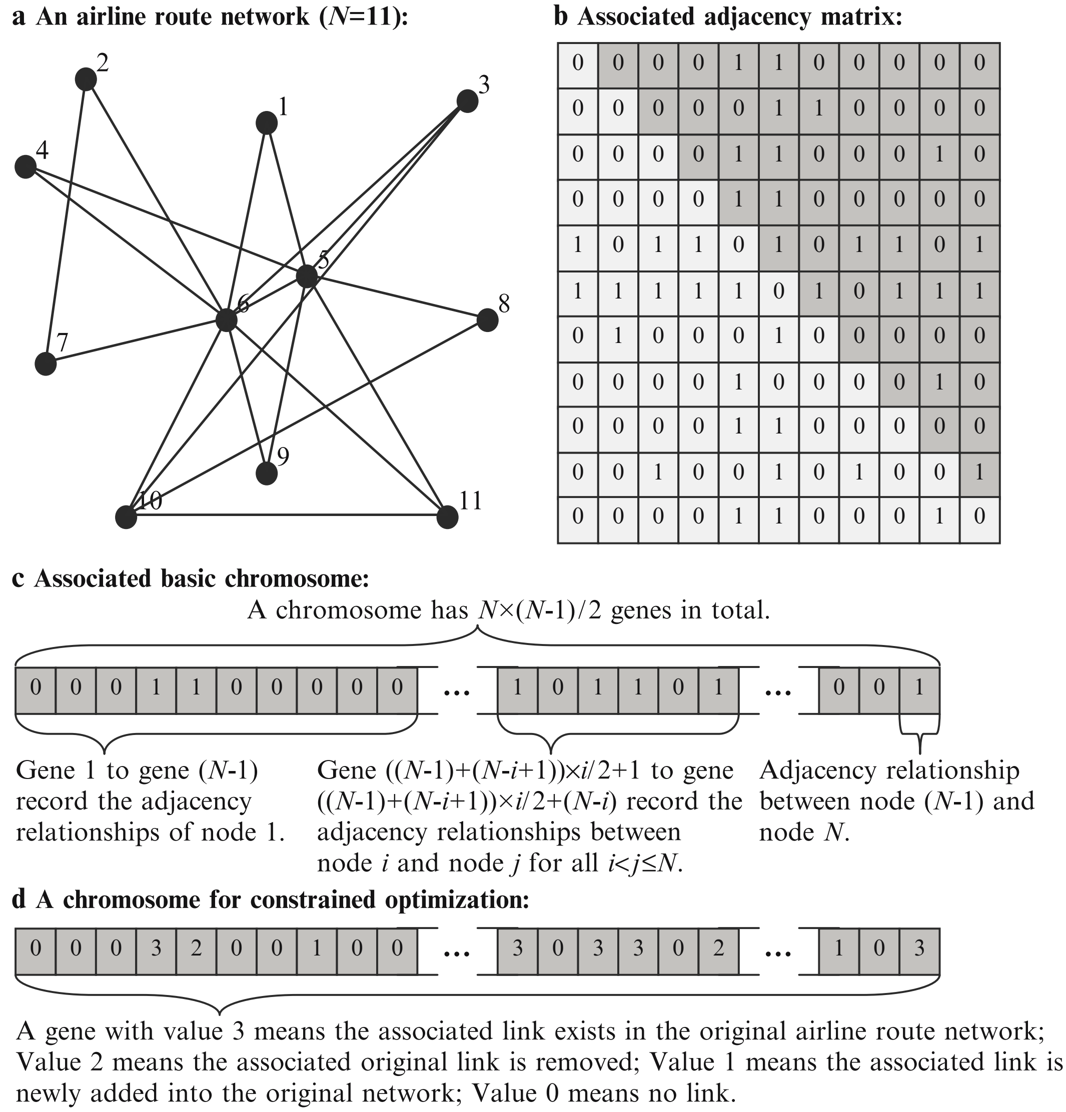

Fig. 1 Chromosome structure

$$\underline{L}_N \leq \sum (g(i) == 1) \leq \overline{L}_N \tag{16}$$

$$\underline{L}_R \leq \sum (g(i) == 2) \leq \overline{L}_R \tag{17}$$

where "==" is a logical function returning 1 when two variables are equal; otherwise, returns 0. Constraints (16) and (17) are for the constrained-optimization problem, while Constraint (15) is for both optimization problems.

During the initialization of the first generation, in the free-optimization scenario, we randomly distribute 1 into a 0-valued chromosome, subject to Constraint (15); while in the constrained-optimization scenario, we reverse the values of some randomly chosen genes in the seed chromosome, subject to Constraints (16) and (17). The reverse operation can be described as

$$g_N(i) = \begin{Bmatrix} 1 - g_O(i), \ g_O(i) < 2 \\ 5 - g_O(i), \ g_O(i) \geq 2 \end{Bmatrix}, \quad i = 1, \ldots, N \times (N-1)/2. \tag{18}$$

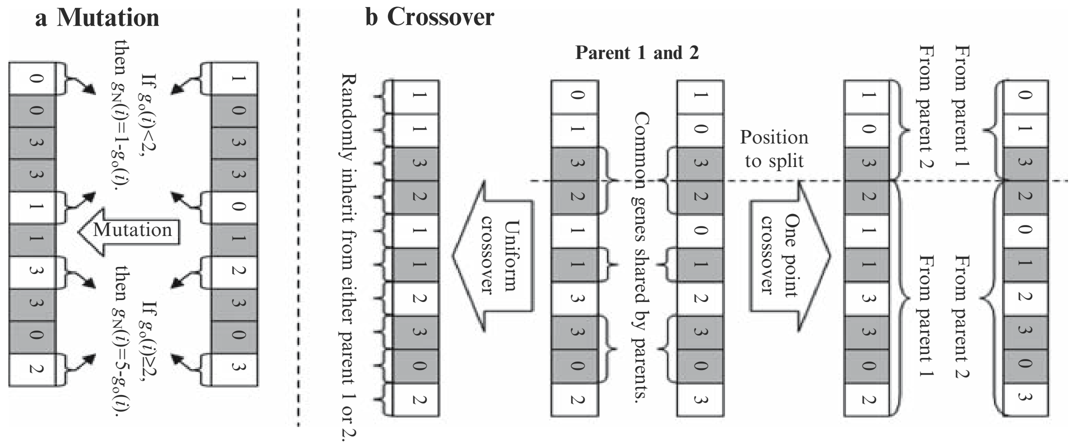

Fig. 2 Evolutionary operators

Mutation aims to increase the diversity of chromosomes, so that the GA can exploit the solution space as widely as possible in order to stand a better chance to hit a global optimum. Basically, the mutation operator chooses randomly some genes in a given chromosome at a certain probability, and reverses their values. To avoid generating unfeasible chromosomes, in the free-optimization scenario, L_{UB} must be observed, i.e., Constraint (15) must be satisfied when a gene is reversed from 0 to 1; while in the constrained-optimization scenario, each reverse operation should be subject to Constraints (16) and (17). Sometimes, a single reverse operation is unfeasible, for instance, when $\underline{L}_R = \overline{L}_R$. In this case, the reverse operation will be carried out in pairs, i.e., by randomly choosing a gene, then randomly choosing another gene with a reversed value, and reversing their values simultaneously. Figure 2(a) gives an illustration of the above mutation operation.

Crossover is used to identify, inherit and protect common genes shared by fit chromosomes, and at the same time, to re-combine non-common genes searching for new solutions. Crossover is crucial for GAs to converge quickly to optimums or sub-optima. In the airline route networks, common genes represent the common connection information (adjacent or not adjacent) shared by different network topologies for the same set of cities. If some good networks include certain same connection information, then it is reasonable to believe this connection information is useful to construct the optimal network. Based on the chromosome structure defined above, one point crossover and uniform crossover can both work well with common genes, as illustrated in Fig. 2(b). In one point crossover, a split point is chosen randomly, each of the two parents split at the chosen point into two pieces, and piece 1 (or piece 2) from parent 1 is combined with piece 2 (or piece 1) from parent 2 to generate offspring. In uniform crossover, each gene in the offspring inherits the same gene from either parent 1 or parent 2 at a half-to-half chance.

In this paper, we choose uniform crossover because (i) it is more powerful in terms of exploiting all possibilities of re-combining non-common genes [13], and (ii) the feasibility issue can easily be addressed. In uniform crossover chromosome feasibility requires we operate on each gene rather than on gene sections/pieces.

For one point crossover, a separate feasibility-checking process is required after the crossover operation, while for uniform crossover, the feasibility issue can be addressed within the operation itself as follows.

Step 1: Inherit common genes, i.e., for $i = 1, \ldots, N \times (N-1)/2$, $g_3(i) = g_1(i)$ if $g_2(i) = g_1(i)$; otherwise $g_3(i) = -1$.

Step 2: Re-combine non-common genes subject to Constraints (15) to (17), i.e., for $i = 1, \ldots, N \times (N-1)/2$, when $g_3 = -1$, do random inheriting if this does not violate any relevant constraint; otherwise, inherit the gene which satisfies all relevant constraints.

4 Preliminary Experiments

In this section we only give the results of some preliminary experiments to demonstrate the potential of the proposed model and the associated GA in the management of airline route networks. Figure 3(a) gives an example of using a combination of P_{ALFC} and PASFD as objective function to reorganize the airline route network, while Fig. 3(b) uses a combination of P_{ALR} and P_{ANR}. In Fig. 3(a), the original airline route network has a $P_{ALFC} = 3.68$. After adding three extra flight routes, the new airline route network has a P_{ALFC} of 2.38. At the same time, P_{ASFD}, is reduced from the original 772.58km to 598.59km. Figure 3(a) illustrates that, by carefully choosing where to add just a few new services, it is possible to significantly reduce the operating costs of an airline company. The airline route network in Fig. 3(b) has poor network robustness in terms of links, i.e., $P_{ALR} = 0.4412$. Therefore, we use the proposed GA to re-organized the flight services: replace 10 old services with the same number of new services. As a result, P_{ALR} has a 66.7% increase to 0.7353,

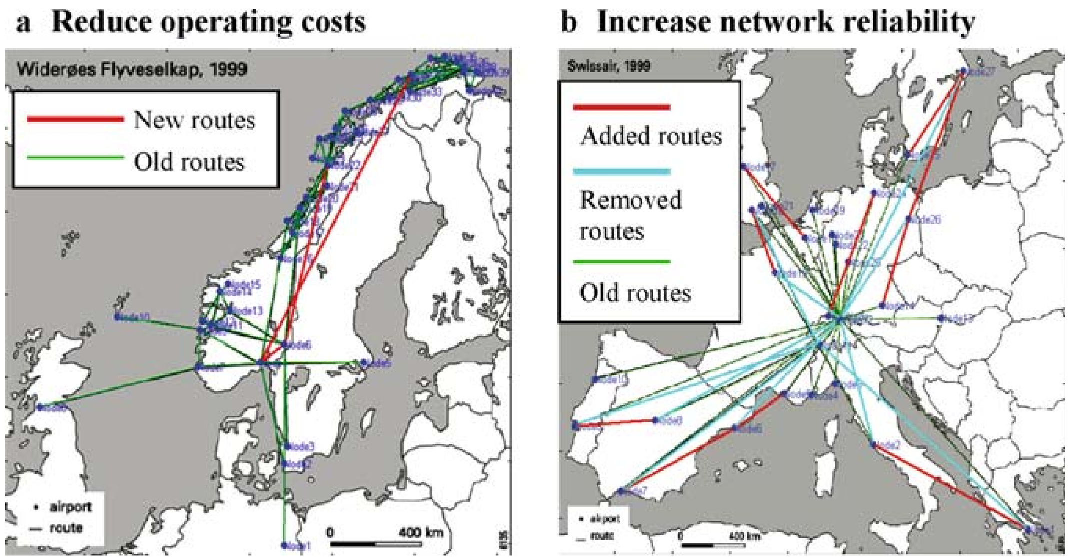

Fig. 3 Results of preliminary experiments

while the network robustness in terms of nodes, P_{ANR}, remains no change, i.e., is still 0.9643. This means the overall network robustness is significantly improved without change the scale of network. However, more effort, improvements, statistical analyses and comparative studies are necessary in order to complete the picture of the proposed modelling as well as the GA.

5 Conclusions

This paper attempts to develop an effective and efficient genetic algorithm for optimizing the topology of airline route networks in terms of cost-efficiency and reliability. To this end, complex network techniques are used to model airline route networks and some key network properties are identified and analyzed. Based on the complex network modelling, a novel GA is developed with emphasis on chromosome structure, mutation, uniform crossover and heuristic rules. The results of some preliminary experiments illustrate the potential of the reported GA for improving the management of airline route networks. Further research is in progress to complete the reported work.

Acknowledgments This work is supported by EPSRC Grant EP/C51632X/1.

References

1. Dennis, N.P.S., (2002) Long-term route traffic forecasts and flight schedule pattern for a medium-sized European airport. Journal of Air Transport Management 8:313–324
2. Reynolds-Feighan, A., (2001) Traffic distribution in low-cost and full-service carrier networks in the US air transport market. Journal of Air Transport Management 7:265–275
3. Burghouwt, G., Hakfoort, J., Van Eck, J.R., (2003) The spatial configuration of airline networks in Europe. Journal of Air Transport Management 9:309–323
4. Brueckner, J.K. and Spiller, P.T., (1994) Economics of traffic density in the deregulated airline industry. Journal of Law and Economics 37:379–415
5. Williams, G., (2001) Will Europes charter carriers be replaced by no-frills scheduled airlines? Journal of Air Transport Management 7:277–286
6. Swan, W.M., (2002) Airline route developments: a review of history. Journal of Air Transport Management 8:349–353
7. Guimera, R. and Amaral, L.A.N., (2004) Eur. Phys. J. B 38:381
8. Barrat, A., Barthelemy, M., Pastor-Satorras, R., Vespignani, A., (2004) Proc. Natl. Acad. Sci. USA 101:3747
9. Boccaletti, S., Latora, V., Moreno, Y, Chaves, M., Hwang, D.U., (2006) Complex networks: Structure and dynamics. Physics Reports 424:175–308
10. Eiben, A.E. and Schoenauer, M., (2002) Evolutionary computing, Information Processing Letters 82:1–6
11. Nagata, Y. and S. Kobayashi, S., (1997) Edge assembly crossover: A high-power genetic algorithm for the travelling salesman problem. In, Proceedings of the 7th International Conference on Genetic Algorithms, 450–457

12. Eiben, A.E. and Smith, J.E., (2003) Introduction to Evolutionary Computing, Berlin: Springer-Verlag
13. Page, J., Poli, P., Langdon, W.B., (1999) Smooth uniform crossover with smooth point mutation in genetic programming: A preliminary study. In Genetic Programming, Proceedings of EuroGP99, Sweden