A Hybrid System For Route Planning

Sarajane Marques Peres, Josué Pereira de Castro
Alejandro Martins, Roberto C. S. Pacheco
Aran Bay Tcholakian, Rosina Weber Lee
Lia Caetano Bastos, Fernando Gauthier

Graduate Program in Production Engineering - PPGEP
Federal University of Santa Catarina, Brazil, UFSC
martins@eps.ufsc.br

Abstract

This paper presents a hybrid system for route planning. The system is composed of two principal modules: a case based reasoning (CBR) and a genetic module. The CBR module is responsible for the user interface. It retrieves the solutions founded by the genetic module, and adapts these solutions to the user preferences. The genetic module is responsible for generating new routes, based on the shortest path between the origin and destination points.

Introduction

Several research areas have addressed the Route Planning optimization problem, including Geographic Information Systems (Abdel-Aty, Abdallah and As-Saidi, 1997), Decision Theory, Genetic Algorithms (GA), Operational Research (Golden and Assad 1988), and Case-based Reasoning (CBR). We propose the combination of GA and CBR to improve route planning efficiency. An effective solution toward this end should deal with constraints such as rush hour delays, traffic patterns, road conditions, and so on. By integrating different artificial intelligence models we have designed a system that targets such objective.

The system is composed of two modules: the case-based reasoning and the genetic module. The first is responsible for the user interface. It evaluates the routes stored in a case base and executes the best retrieved solution. If none adequate route is retrieved, the genetic algorithm module is activated to create new routes. When the GA does not meet any feasible route, the system proceeds by adapting one of the similar solutions found in the case base.

Route Planning

Route planning is an optimization task with a variety of applications. Several authors have dealt with this subject. Lee and Fishwick (1995), for instance, developed simulation techniques to automate the decision making process in uncertain and complex environments. Christofides (1985) also describes an algorithm and several heuristics to solve vehicle routing problems.

The proposed system aims to help drivers to decide the route to be taken between two given points in urban zones. The route chosen must be both efficient (regarding minimal feasible length) and satisfactory (in terms of driver's preferences). Meeting both objectives has been neglected in route planning literature, making optimization techniques insufficient to balance optimality and user's satisfaction. The combination of CBR and GA made possible to find routes where both aspects are considered.

CBR and GA in Related Applications

Case-based reasoning and genetic algorithms have already been applied in traffic control and route planning, either as combined or stand-alone approaches.

Nakamiti and Gomide's work (1994) have applied CBR and GA to manage traffic flow more efficiently. Given a traffic condition, the system controls the light times by adapting successful solutions given for previous similar conditions. The system uses GA to adapt the retrieved cases from the case base. The retrieval is based on attribute similarity between the input and the stored cases.

Haigh and Veloso (1995) developed a case-based reasoning system for route planning method. The routes used in the past are stored and may be retrieved and reused to generate new routes. In this system, the planner can use several retrieved cases to generate a new route by merging these cases. An *efficiency value* is associated to each retrieved case, indicating its "quality". Depending on these values, the system either reuses the known routes or searches for alternative routes.

Choi and Woo (1997) proposed an evolutionary route planning algorithm to reveal the optimal route between origin and destination in road traffic networks. Their work also presents a simulation of a network with several constraints.
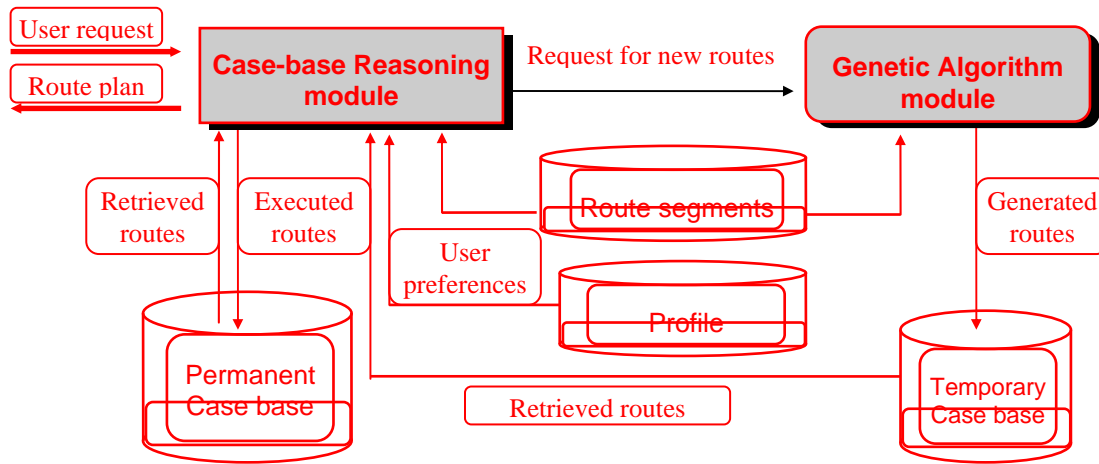
Figure 1: System Architecture.

The System Modeling

Balancing optimality and user satisfaction requires a model that keeps tracking of both targets while seeking for a solution. This can be understood as the need for *general knowledge*, that is, the model has to balance requirements coming from different frameworks. In CBR theory, a system that employs an external method of general knowledge is classified as a horizontal integrating system (Aamodt, 1993). In order to provide general knowledge we combine CBR with GA (improving the search for intended results). The system is then classified as a *hybrid system*, since it combines two AI techniques on a single horizontal integrating system (Medsker, 1995).

The route site (street traffic network) is represented by road segments and their intersections. Each route is composed by a subset of adjacent and ordered segment. The route begins by the initial point of the first segment (origin) and ends with the final point of the last segment (destination).

The system uses the following parameters: (a) *the user's profile*, consisting of an archive with restrictions (*e.g.,* site restrictions, street conditions preferred, etc.) for each user category (*i.e.,* ambulance service, taxi drivers, etc.); *the user's request* (origin and destination and traffic time). The user's request is converted into a case, firing the CBR module. This module searches for similar cases (eventually adapting the most similar routes to the input case), presenting the solutions to the user. If no similar case is found, the genetic module is activated.

When requested, the genetic module creates new routes converted into cases and returns them to the CBR module. The CBR, then, proceeds the adaptation to the input case. Once again, the user checks the solution. If, after this process, a satisfactory route was not found, the system displays the restrictions prohibiting the route generation. The system architecture is briefly described in Figure 1.

The Case-based Reasoning Module

The first requirement in the proposed model is to meet the user profile. Humans tend to keep or adapt previous successful approaches when facing new similar situations. This was the main motivation for building a Case-based Reasoning module to pursue a satisfactory route plan, applying analogical reasoning. This module is responsible for presenting route suggestions to the user based on his or her request. Figure 2 depicts the route representation used in the CBR module. Some attributes are used as indexes and others as route descriptors. The case indexes are the following: origin and destination points, segments, case suitability degree, and user's profile. The other attributes are only descriptive and are presented to the user as additional information.

ID	Origin	Dest	Adequate Degree	Descriptive Attributes		ID	Segments

Figure 2:   Route representation in the case base.

Initially, the module searches the case base for the most similar route to the one desired by the user. Eventually, there is need for performing a route adaptation (*e.g.,* traffic accident). In this case, the adaptable route has segments that can be replaced by another set of *satisfactory* segments. The substitutive set of segments is obtained from a partial case retrieval. Figure 3 illustrates an example of adaptation process. The interval ABC can be replaced by the interval ADEC, observed in a different case.
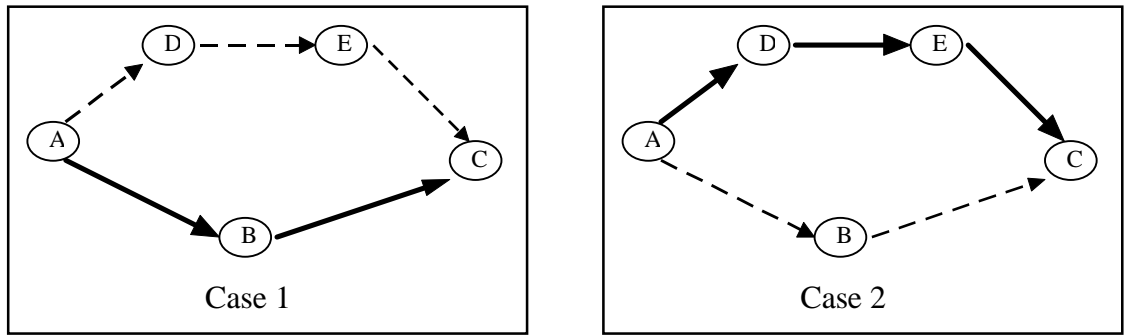
Figure 3: Route adapting process.

If no satisfactory solution is found, the genetic module is activated, a new temporary case base is created, restarting the retrieving and adapting processes.

If the route selection was successful, the system runs the solution. Afterwards the user chooses whether or not include the route in the case base.

One of the most significant aspects in route planning is the need for rapid adaptation due to traffic network updating. In this system, the CBR module makes adaptive learning possible. In the beginning, the case base is empty. With its utilization, the routes are created, run, and stored in the case base, characterizing a continuous learning process.

The Genetic Module

The GA role is to find feasible routes between the user's origin and destination points. The strategy consists in searching and evaluating alternatives, seeking for the shortest route path.

The first modeling issue is how to represent routes as binary codes (*i.e.,* GA chromosomes). Previous work (*e.g.,* Whitley, Starkweather and Shaner, 1991) have addressed this matter by allowing every two-point combinations. This is valid only for networks fully connected. Actual routes require that every segment is a feasible path. This makes the usual mutation and cross-over operations inadequate, since they can lead to non-existent segments. In this work, we have developed an order independent scheme that makes the GA operations applicable in non-fully connected networks. This scheme allows the modeling of oriented graph and also guarantees route feasibility.

The GA module begins by establishing a random set of routes. Each route is a set of oriented segments, created by gradually adding paths. The addition of segments is guided by the shortest Euclidean distance between the candidate nodes and the destination point. The route assembling continues until the algorithm reaches either a final segment (hitting the destination) or the maximum number of generations. GA proceeds attempting to establish more

routes, until reaching a fixed number of runs (population size).

In this module, each route is represented as a chromosome (see Figure 4), where:

- a street traffic network segment is modeled by an allele;

- each allele receives one of the values in {0,1,*,#}. Value '0' means that the segment, at the current iteration, does not belong to the route; '1' indicates that the segment is a path candidate; '*' indicates a route segment; and '#' marks the segment as unfeasible for the current route;

1	2	3	4	5	6	7	8	9	10	...	N
0	*	0	#	*	1	*	#	1	0		0

Figure 4: Route represented in a chromosome form.

The *fitness operator* is applied to every allele valued with '1'. It is defined by three operations: (a) calculus of the Euclidean distances between all candidates and the destination; (b) *feasibility test*, which verifies each segment feasibility; and (c) *evaluation*, which checks for either change the allele value (to '*' or '#') or keep it as a candidate, according to the following criteria:

- If the candidate cannot be connected to the current segment set (*i.e.,* it is an unfeasible segment), take its Euclidean distance to the destination (d_c). Take also the shortest Euclidean distance between the current node set and the destination (s_d) (*i.e.,* the shortest Euclidean distance among the '*' alleles). If $d_c > s_d$, mark the candidate allele with '#'. This means that this segment will definitely not belong to the current route.

- Among all feasible candidates, take the one with minimum Euclidean distance to the destination and convert its allele value to '*'.

The *mutation* operator works on each allele valued with

either '0' or '1', randomly deciding whether to change it or not to its opposite value. The alleles valued with '*' or '#' remain unchangeable, fixing the schemata in order to assure segment feasibility and the algorithm efficiency in further operations.

The algorithm in Table 1 describes the genetic module operations:

Table 1: Algorithm used in the GA module.

MaxPop = *max number of desired individuals (constant)*
MaxGen = *max number of desired generations (constant)*
Gen = 0 {generation counter}
CompleteRoute = boolean {does the route hits the destination?}
For I := 1 to MaxPop do
    Repeat
      Fitness {try to change all '1' to '' or '#' alleles}*
      Mutation {only in '0' and '1' alleles}
      Until (CompleteRoute) or (++Gen = MaxGen)
    End

The GA output is an ordered set of optimized routes. The single criteria for discovering these routes was the shortest path. The other user's criteria are still missing. In order to balance length and satisfactory aspects, the GA responses have to be migrated to the CBR module. This is performed by transforming the GA outputs into a case base format (temporary case base). Depending upon the user decision, the temporary cases are incorporated into the permanent case base by the CBR module.

Case Study

As an example of the system application, we choose a ambulance service, whose principal restriction is the execution speed of the route. In time critical problems, it is important to avoid paths where traffic is influenced by daytime. For this reason, we chose the route execution hour and day as the attributes in the profile archive. Figure 5 depicts an actual street traffic network for the ambulance.
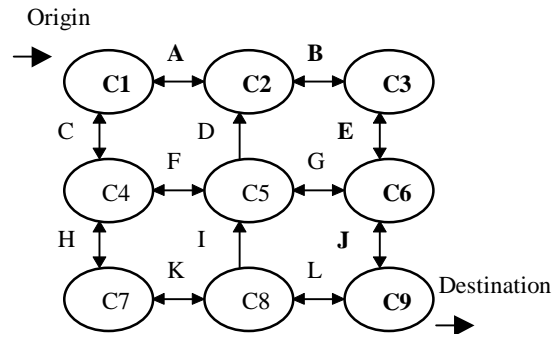

Figure 5: Example of street traffic network.

First, the user (ambulance driver) asks for a route plan. The request is evaluated by the CBR module, which seeks for similar cases. In this case, the suggested route was 'A-D-I-L'. The user was unsatisfied by this route plan. He knows that segment 'L' is currently blocked. The CBR module calls the GA module, asking for alternative solutions, and marking the segment L as unfeasible in the street traffic network profile.

A	B	C	D	E	F	G	H	I	J	K	L
*	1	1	0	0	0	1	0	1	0	0	#

Figure 6: First individual of the population.

As shown in Figure 6, the GA module begins by establishing an first individual of the population (*i.e.,* a set with the first route segment and other candidates).

The next step is to evaluate this individual, searching for the segment feasibility (Figure 7). In this case, segment B is the only feasible path. The algorithm also analyzes whether the unfeasible segments can eventually belong to the current route. In this case, segment C will never belong to any route containing segment A. Thus, this segment is made permanently unfeasible (*i.e.,* it is marked with '#').

A	B	C	D	E	F	G	H	I	J	K	L
*	*	#	0	1	1	0	0	0	1	0	#

Figure 7: Individual after fitness followed by mutation

The route design process remains until a final segment is found or until the algorithm hits the maximum number of generations. Figure 8 shows the result after a sequence of GA iterations over the same route plan.

A	B	C	D	E	F	G	H	I	J	K	L
*	*	#	0	*	0	1	1	1	*	0	#

1 2   3   4

Figure 8: Complete and ordered route obtained after some fitness and mutation operations.

The next GA step is to convert the suggested route plan into a case and send it to the temporary case base. The result is shown in Figure 9.

ID	Source	Destination	Adeq. Degree	Hour	Day	weather Conditions
1	C1	C9	90%	12:00	Mon	Good

(a)

ID	Segment
1	A
1	B
1	E
1	J

(b)

Figure 9: (a) Temporary case base; (b) Route adjusted in a form of a case.

Finally, the CBR module takes place and presents the new route to the user. If he decides to take the suggestion, the CBR integrates the new route into the permanent case base for future use in similar situations.

Conclusions and Future Work

In this work, we have presented a new model for route planning problems. In this approach, the suggested route plans are both efficient and satisfactory. They represent a balance between optimality (minimal feasible length) and user requirements. This was made possible by means of a hybrid intelligent system, including a module for each purpose. The user's requirements are pursued by the Case-base Reasoning module, while optimality is seek by the Genetic Algorithm module.

Analogical Reasoning and adaptive learning were the main motivation for building a CBR. This module not only takes similar past experiences to propose route plans, but also uses the new information to increase its memory for further iterations. When a solution is not available (insufficiency of knowledge or user dissatisfaction), CBR calls the Genetic Algorithm module.

The GA purpose is to establish route plans meeting the new conditions and keeping optimality as the primer concern. In addition, the method had to keep feasibility constraints and boost optimality search. We achieved these objectives developing a new chromosome representation scheme, suitable for route plan problems in oriented graph networks. The found route plans are evaluated by the CBR and presented to the user. The new route plan is eventually incorporated in the permanent case base.

The interaction between the CBR and GA aims to balance satisfaction and optimality constraints. By keeping the objectives separate and integrated we avoid increasing complexity in each module (*e.g.,* by including user profile in the GA, one would increase iteration time, since the fitness would be a more complex operator). We have also made adaptive learning possible, since the case base is dynamic with the time.

As future work, the group is evaluating different alternatives. First, we intend to implement an interface with a Geographic Information System (GIS). This optimizes the addition of new street traffic network configurations. We are also studying the possibility of including a fuzzy expert system to deal with season variables (*e.g.,* tourist season, concerning school, etc.) and temporal data (*e.g.,* daytime), related to traffic flow conditions. An example of combining GIS and fuzzy modeling can be found in Lee and Lee's work (1996). Finally, we are also studying other approaches for the optimality search, such as the use of GA* (Logan and Poli, 1997).

References

Abdel-Aty, M.A., Abdallah, M.N.and As-Saidi, A. H. 1997. A Methodology for Route Selection and Guides Using GIS and Computer Network Models. In *Proceedings of 76th Annual Meeting of The Transportation Research Board*, 20-21. Washington D.C.: Transportation Research Board.

Aamodt, Agnar. Explanation-Driven Case-Based Reasoning. *Topics in Case-Based Reasoning, (First European Workshop, EWCBR-93).* Wess, Stefan, Althoff, Klaus-Dieter & Ricthter, Michael (editors) Springer-Velarg, 274-288, 1993.

Choi, G.S. and Woo, K.D. 1997. The development of Optimal Route Algorithm Based On Evolution Program. *The Transactions of The Korean Institute of Electrical Engineers* 46(2):294-297

Christofides, N. 1985. Vehicle Routing In The Traveling Salesman Problem - In *A Guided Tour of Combinatorial Optimization,* 431-448. Great Britain.: John Wiley & Sons Ltd.

Golden, B. L., and Assad, A., 1988. *Vehicle Routing: Methods and Studies.* Amsterdam, Netherlands: Elsevier Science Publishers B.V.

Haig, K. Z. & Veloso, M.,. 1995 Route planning by analogy. *Proceedings of Case Based Reasoning research and development, First International Conference: ICCBR-95*, Springer-Verlag, 171-180.

Lee, J. and Fishwick, P.A., 1995. Simulation Based Real Time Decision Making for Rout Planning. *1995 Winter Simulation Conference*, 1087-1095. Crystal City, VA.

Lee, H.C., and Lee, C.C. 1996. Inexact strategy and Planning in Taipei City. *In Proceedings of Asian Fuzzy Systems Symposium, 1996, Kenting, Taiwan, Soft Computing in Intelligent Systems and Information processing*, 308-313. Piscataway, NJ.: IEEE Service Center.

Logan, B and Poli, R. 1997. Route Planning with GA*, Technical Report, CSRP-97-17, School of Computer

Science, University of Birmingham, England

Medsker, L. R. , *Hybrid Intelligent Systems*, Kluwer Academic Publishers: Boston, 1995.

Nakamiti, G. & Gomide, F., An evolutive fuzzy mechanism based on past experiences,. *Second European Congress on Intelligent Techniques in Soft Computing*, Aachen, Germany, sept, 1994.

Whitley, D., Starkweather, T. and Shaner, D. 1991. The Traveling Salesman and the Sequence Scheduling: Quality Solutions Using Genetic Edge Recombination. In *Handbook of Genetic Algorithms*, 350-372. New York, NY.: Van Nostrand Reinhold.