REAL TIME SOFTWARE PLATFORM FOR DYNAMIC ON-DEMAND PUBLIC TRANSPORT

Harbil Arregui
Research Assistant, Vicomtech
harregui@vicomtech.org
Estíbaliz Loyo
Researcher, Vicomtech
eloyo@vicomtech.org
Oihana Otaegui
Head of Department, Vicomtech
ootaegui@vicomtech.org
Tel:+[34] 943 30 92 30, Fax:+[34] 943 30 93 93, www.vicomtech.org

ABSTRACT

A new integrated demand-responsive transport system is described in this paper, introducing an alternative to overcome the inefficiencies that traditional public transport schemes present in low-density housing areas in and around city centres. On-demand public transport satisfies many of the needs that fixed predetermined routes and schedules cannot fulfill. Both schemas are able to work together, as it is discussed in this paper, being on-demand services a good choice in areas and time intervals where low demand makes fixed routes unsustainable. The proposed solution consists of a flexible integrated management system for road public transport, integrating fixed services, on-demand fixed services and on-demand dynamic services. The architecture of the system and the features of the tailored two-stage algorithm for the dynamic routes are detailed in the paper. The main objective of this algorithm is to solve the multivehicle Dial-a-Ride Problem (DARP) minimizing the number of vehicles and the total distance of the routes. A heterogeneous capacitated fleet is used to serve the customers and the cost matrix is asymmetric, as real-life conditions are assumed. The desired quality of service is set by configurable parameters (e.g., waiting time window and maximum ride time). The performance shown by the algorithm during the tests made, permits offering real-time response to user requests.

INTRODUCTION

Fixed public transport services operate according to fixed schedules and routes even though the demand at each moment is low or inexistent, which means that there can be quite a lot of waste in operating the transport system. That waste cannot be reduced unless the general quality of the service is also reduced, offering, for example, fewer buses. On-demand public transport works following user requests, and therefore satisfies many of the needs that traditional public transport cannot fulfill. Both schemas are able to work together, being on-demand services a great alternative in the areas and time intervals where low demand makes fixed routes and schedules unsustainable.

Some on-demand transport trials have already been successfully done in several countries, showing multiple social, economical and environmental benefits (1). However, many of those implementations need requests to be made in advance. The routing and scheduling algorithm of the presented software system responds to user requests in real time, generating the necessary order of stops to be followed by the vehicles. A new algorithm has been developed and integrated in the system for the generation of routes and schedules needed by these latter services.

The on-demand dynamic transportation approach presented here is found in the literature inside the Dial-A-Ride Problem (DARP) variants. DARP is a particular case of the Pickup and Delivery Problems (PDP) for passenger transportation. In (2), authors provide a general framework for dynamic one-to-one PDPs, presenting the most relevant related literature (see also (3)). Unlike in static routing problems, in a dynamic routing problem, some of the input data (in most cases user requests) are revealed or updated during the period of time in which operations take place, so a solution cannot be a static output. Several constraints exist and are used to classify different problems (4): e.g. limits in the fleet and in the capacity of the vehicles, time windows, journey durations or distance limits, (5) (6).

Examples of this kind of applications put in practice in many countries are door-to-door services for elderly or disabled (7). However, most of them require previous arrangement of the pickups, so no real time services are provided. Some initiatives are being carried out, but currently there are no operating systems deployed in the province of Gipuzkoa (Spain).

CURRENT SITUATION OF THE PUBLIC TRANSPORT IN GIPUZKOA

Gipuzkoa (Spain) is a very dispersive province with vast highly populated areas and high mobility rates among different municipalities. It is divided in 7 smaller regions and they all display important levels of generation and attraction of journeys as the weight of the capital Donostia-San Sebastian with respect of the rest of the territory is relatively low in comparison to the other provinces.

Among the seven regions of Gipuzkoa, it can be specially considered the region of Tolosaldea, which, with an area of 332 km^2, is the less populated region with 45000 inhabitants. However, it has the biggest amount of municipalities, 29. Bus services and mobility in this area were studied during the year 2008, in order to define the road map to strengthen and improve the offer. The municipalities mainly located along the 4 central axis of the region, all of them around the main city, Tolosa, showed quite good overall access to transportation. Otherwise, the rest were very limited and some small population nucleus did not even have any scheduled service. Many logistical problems were detected:

- Lots of services carry very few passengers
- Inefficient fleet management
- Many time-outs
- Areas of difficult access
- Fixed routes regardless of the number of users
- Some journeys are run with empty buses (out of service or in service without any passengers)
- Dead intervals between services
- Different areas are not equally covered

OBJECTIVES

The promotion of the use of public transport lies in improving the service offered, towards a flexible and fast public transport, adapted to the needs of users in small municipalities to connect to the big cities. The objectives can be stated as:

- Improve mobility rates in the regions of Gipuzkoa
- Increase the number of users
- Optimize the offered services
- Extend the service to currently non-served areas
- Provide the user with updated information about traffic flow and ways of mobility

PREVIOUS EXPERIENCES

A first stage towards the proposed integrated mobility system was made by Vicomtech in 2008, with LANDABUS project in Aia (Gipuzkoa, Spain), with the aim of covering the rural area of Aia-Zarautz-Orio with flexible bus services. Fixed and on-demand routes were designed to serve the local inhabitants. On-demand routes visit predetermined stops at previously set schedules, but each user willing to travel must make a request by telephone, SMS or Internet, at least 30 minutes in advance so as to activate the desired service.

The trial with an initial duration of a year started in June 2008 and by October, passengers tripled. After 3 years, the service still continues running with users showing high degrees of satisfaction. LANDABUS meant to be a basis to afterwards expand the concept to other regions and the new implementations of the model want to migrate to a totally on-demand service looking for a stronger resource optimization.

NEW PROPOSAL

A newer version with more flexible features is decided to include in the system in order to increase the number of passengers, optimize the journeys and extend the service to areas that are currently neglected. The proposal of the new public transport system combines 3 kinds of services:

- Fixed services (FS): Fixed timetables and routes that do not vary and are always active.
- On Demand Fixed Services (ODFS): Fixed timetables and routes, activated on demand (e.g. LANDABUS in Aia).
- On Demand Dynamic Services (ODDS): Vehicles pick riders up at the times and places specified by them. Hence, services must be generated or updated each time a request is received.

The area is divided in two line types: Main lines (with nodes in easily accessible areas, often with high population rates) and Secondary lines (they connect more remote nodes, accessible by secondary roads). In addition, the day is divided in different time intervals according to the demand: Rush hours (more than 5 passengers by vehicle on average) and Valley hours (less than 5 passengers by vehicle on average).

Fixed services are scheduled for rush hours. They travel along fast lines through the four main routes and make few stops, lasting no more than 30 minutes. Shuttle services may connect the main routes with the more scattered municipalities that are not included in the fast lines.

Demand responsive services are chosen for the valley hours, at the intervals where traditional fixed routes are more likely to travel empty or with very few passengers. The vehicles that are in idle state, waiting for their fixed routes to start, can be used to transport on-demand users. On the one hand, the main lines will be supplied by on-demand services that follow preestablished routes and schedules. On the other hand, the secondary areas will be traveled following dynamically generated routes. These dynamic journeys are created by the ODDS algorithm explained below.

SYSTEM ARCHITECTURE

The general schema of the platform is shown in Fig. 1.

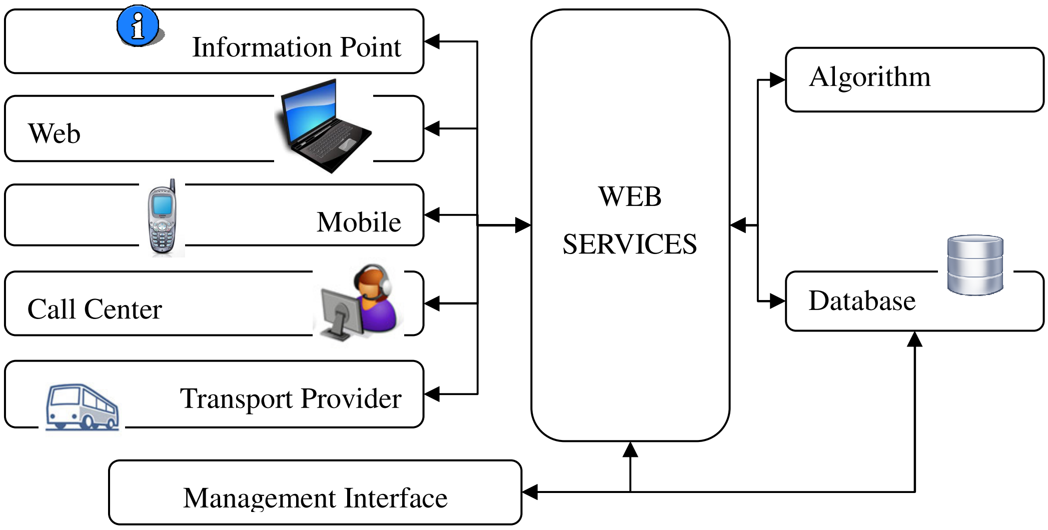

Figure 1. General overview of the system architecture

The system includes: a database designed for the management of the entities involved, a routing and scheduling algorithm to calculate ODDS, web services, and management, transport service provider and user interfaces. Multimodal interfaces can access the system through the web service platform allowing users make/update requests and consult services on line, via SMS, using the Call-Center service or information points. Transport service provider is informed about the routes that the vehicles must follow thanks to the interface created for that purpose. Onboard equipment in vehicles is able to connect to the system to update their location and the status of the services.

ODDS ALGORITHM

On-Demand Dynamic Services (ODDS) need users to specify the origin and destination stops and the desired time of pickup (or drop-off). An insertion heuristic algorithm is used, according to the requests received, to dynamically generate the order of the stops to pickup and to deliver all the users at the specified time, assigning a suitable vehicle to ensure that capacity constraints are not violated. The problem set for the algorithm follows the approach of the Dial-a-Ride Problem. The solution presented in this paper relies on two operating modes (insertion and optimization) and it has been designed to minimize the number of vehicles and the total distance run by the whole set of services generated for the considered time slot. It successfully administers a heterogeneous fleet with seats for the general public and for those users with limited mobility. The quality of service is guaranteed by a maximum waiting time threshold at the pickup stops and a maximum ride time threshold for each user, defined as a linear function of the direct travel time between pickup and delivery.

PARAMETERS AND CONSTRAINTS

Stops and distances

The management interface allows the administrator to add and update the location and information of stops in the database. For the implementation of the system it was decided that the list of possible pickup and drop-off stops would be previously predetermined. With the driving distances between all the pairs of stops, a matrix is built. The algorithm uses the distance measured in time, that is, the duration of the journeys.

So, d(A,B) is the direct (shortest) driving distance in minutes between stop A and stop B, being $d(A,B) \leq d(A,C) + d(C,B)$. In real life conditions distance matrix is asymmetric, that is, $d(A,B) \neq d(B,A)$, due to possible one-way roads.

Requests

The parameters retrieved from each request *r* are:

- O*r*: Origin stop.
- D*r*: Destination stop.
- TO*r*: Pick-up time at origin.
- p*r*: Number of general passengers.
- pRM*r*: Number of passengers with special mobility needs.

Passengers

For each stop *i* in a service, passengers are classified as:

- pin(*i*): Number of general passengers willing to get on the vehicle.
- pout(*i*): Number of general passengers that get off the vehicle.
- pinRM(*i*): Number of passengers with limited mobility willing to get on the vehicle.
- poutRM(*i*): Number of passengers with limited mobility that get off the vehicle.

Finite heterogeneous capacitated fleet

Vehicle-related information must be stored in the database previously using the management interface. It is possible to specify the time slot according to each type of day, when a vehicle is available for ODDS services. For a vehicle *v*, its capacity can be defined with the three following parameters:

- C*v*: Number of seats for general passengers.
- CRM*v*: Number of seats for mobility reduced passengers.

- eq$v$: Equivalence between general and mobility reduced seats, thus, number of general passengers that can travel in the place of a mobility reduced passenger. This value has to be eq$v \geq 0$.

Being n the number of stops to visit by vehicle v, capacity must ensure that for all s = 0, 1,... n, if $C_v \geq \sum_{i=0}^{s}(pin(i) - pout(i))$, then, $C_{RMv} \geq \sum_{i=0}^{s}(pin_{RM}(i) - pout_{RM}(i))$.

Else, eq$v$ must be non-zero and,

$$C_{RMv} - \sum_{i=0}^{s}(pin_{RM}(i) - pout_{RM}(i)) - \frac{1}{eqv}\left(\sum_{i=0}^{s}(pin(i) - pout(i)) - C_v\right) \geq 0.$$

If none of the conditions is satisfied, passengers cannot be served by vehicle v.

Waiting time window

The maximum time window W is used as an adjustable parameter of the system, and its value is applied for all the requests at first. It defines the maximum time a user should be awaiting at the pickup stop. Every time new requested stops are introduced into the stop sequence of a service l, time windows Wl are readjusted (tightened) by the Insertion Mode of the ODDS algorithm. The width of these time windows is the same for all the stops in a service at the end of the procedure, and the widest the window, the more flexibility the service has to accept new requests. The vehicle assigned should preferably visit stops at the beginning of the window.

Maximum ride time M*r*

The maximum ride time is defined for each request r as a linear function of the direct travel time between pickup and delivery. It guarantees that each user does not spend more than a certain time on the vehicle. The value of Mr is defined as: $Mr = d(Or, Dr)[1 + R]$, where R is the maximum percentage of time that the system is allowed to exceed the direct traveling time between two points. In addition, absolute maximum and minimum values are established (i.e. Tmax and Tmin), so that if M$r \geq$ Tmax or Mr $\leq$ Tmin, they are respectively considered as threshold. Finally, notice that the time T(Or, Dr) spent by a service to travel between Or and Dr must fulfill $d(Or, Dr) \leq T(Or, Dr) \leq Mr$.

INSERTION MODE

The basis of the algorithm is to insert new requests into existing services thanks to the flexibility given by the nonnegative time window parameter. The origin and destination stops of each request are chronologically ordered in a list and the feasibility of the sequence is analyzed. Time windows are adjusted taking into account the distances between each pair of sequential stops in the new order.

- TO1*r*: time window beginning at origin
- TO2*r*: time window end at origin
- TD1*r*: time window beginning at destination
- TD2*r*: time window end at destination

A service *l* dedicated uniquely to serve a request *r* is graphically represented in Fig. 2(a). If a new request *q* is received, Fig. 2(b), the system tries to insert it in the same service and update the time windows, Fig. 2(c). Chronologically ordered the beginning of the windows, the new temporal sequence of the stops is (Or, Oq, Dq, Dr). Once inserted, stops lose their pickup/drop-off meaning, and a service becomes a sequence of n generic stops (A, B, C, …, n). A stop sequence is feasible if a vehicle can travel from the first to the last stop visiting each stop at a time inside its window. Taking into account that the traveling time between two consecutive stops A and B is d(A, B), the beginning and end of the time windows must be adjusted ensuring W*l* > 0. If it is not feasible, while the maximum ride time constraint is not violated, the drop-off time of the new request can be shifted forward changing the sequence and checking the feasibility again.

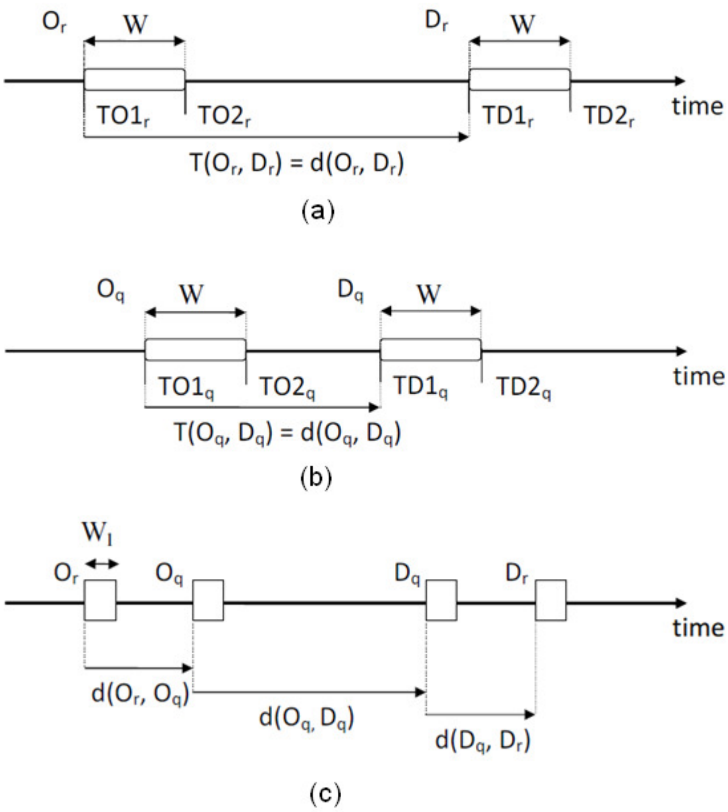

Figure 2. Request *r* (a) and Request *q* (b) are inserted into the same service obtaining the new stops sequence (c).

The Insertion Mode is of great interest in the case of users asking for immediate transportation to accept them. It ensures that the scheduled stops will be traveled as expected without delay even if a new stop is included in the journey.

OPTIMIZATION MODE

Periodically, for services still not started, the set of results that has been accepted by the Insertion Mode is reconsidered by the Optimization Mode. This amount of requests is taken and the order they are introduced into the stops sequence is changed in each run, choosing the best result at the end of the computing time. Note that requests previously accepted and confirmed to the users cannot be rejected now. This optimization is multi-objective, the best result can be chosen among different priorities or a combination of them:

- Reduce total route duration
- Reduce total vehicles
- Reduce rejected requests (only those not confirmed)

The objective selected for the implementation of the final management system is the one with the minimum number of vehicles, with the shortest total traveled route.

RESULTS

Several tests have been made by simulating the expected demand according to the mobility study described above. Requests were created on purpose using real stop locations in the studied geographical area and the general configuration parameters used are W: 10 minutes and R: 0.4.

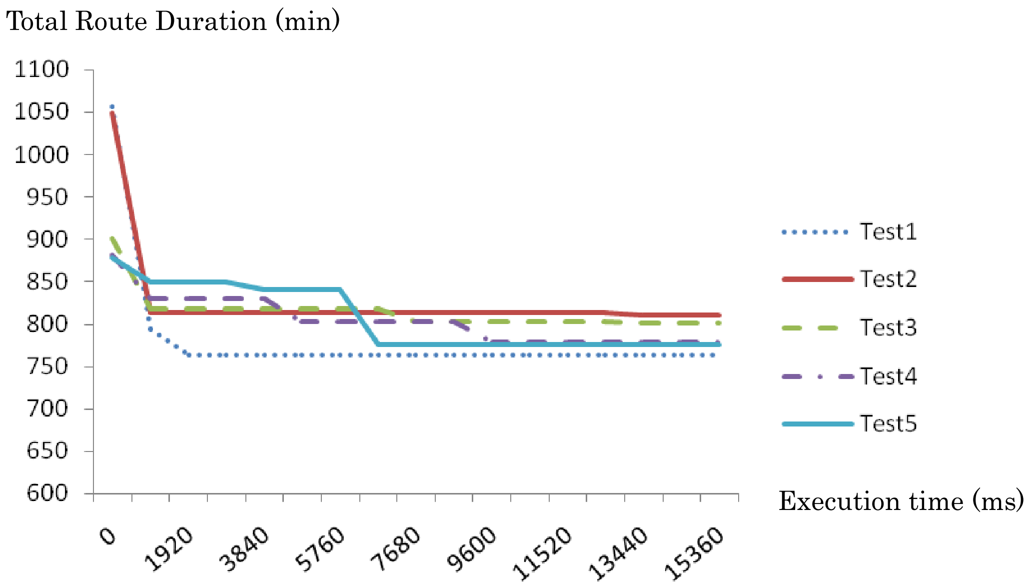

Figure 3. Total duration of the journeys, for 5 tests performing Optimization Mode.

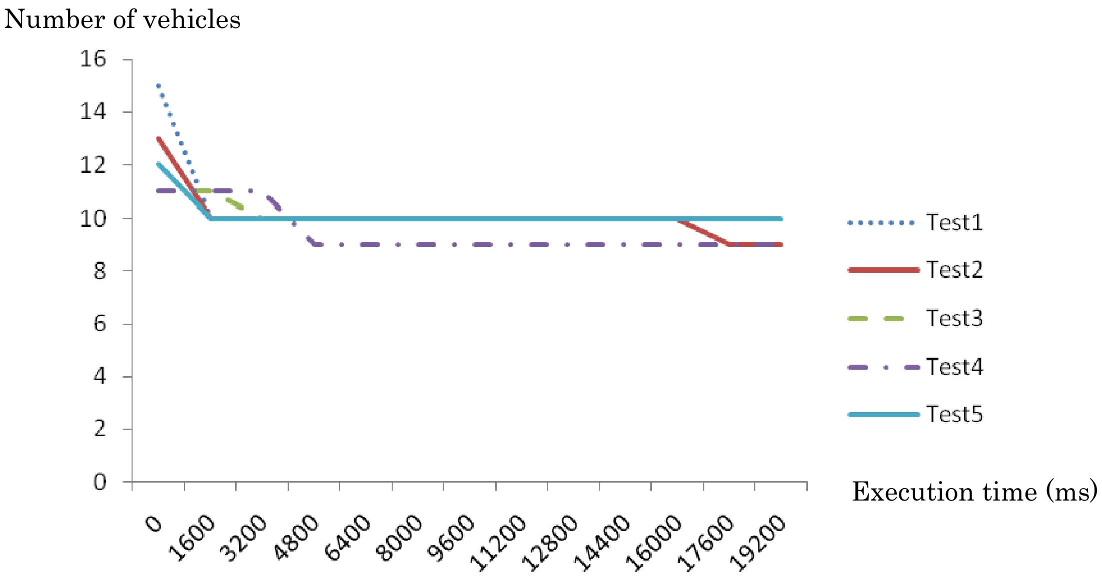

Figure 4. Total number of the vehicles needed, for 5 steps performing Optimization Mode.

The charts in Fig. 3 and Fig. 4 show the performance for a test case of 40 user requests distributed in 20 different stop locations, in a three hours time slot. The optimization algorithm has been tested 5 times to measure the response. It is shown that an immediate response can be achieved even with the optimization mode for the data used. In all 5 tests it takes less than a second to improve the route duration obtained by the first insertion execution reducing values to the 75% and they soon get to an almost stable solution. For instance, after one minute of execution total journey duration values for Test1 and Test4 remained the same as the ones obtained at 1.7 and 8.8 seconds respectively.

CONCLUSIONS

This paper presents a functional software platform to make road public transport more flexible, not only allowing users to have better access but also reducing the cost of the services in areas where traditional fixed routes are economically unsustainable. ODDS are proposed for specific areas and time slots and a new algorithm has been developed to manage requests and generate routes. Insertion Mode can easily study the acceptance of users even in vehicles that are running without making major changes in the routes. Optimization Mode can be configured to be performed during the desired computation time and, for the demand load in the studied geographical areas, it can even be valid to answer user requests online.

In conclusion, the system is of great help for an innovative understanding of efficient collective transportation and it is expected to be implanted in a real scenario by the end of the year.

ACKNOWLEDGEMENT

The authors would like to thank the Gipuzkoa Provincial Council, which has funded on demand transport project in Aia, constituting a starting point for this work.

REFERENCES

(1) "Evaluation Study of Demand Responsive Transport Services in Wiltshire". Available at: www.wiltshire.gov.uk/demand-responsive-study-200607.pdf

(2) G. Berbeglia, J.-F. Cordeau, and G. Laporte, "Dynamic pickup and delivery problems," *European Journal of Operational Research*, vol. 202, no. 1, pp. 8–15, 2010.

(3) M. Savelsbergh and M. Sol, "The General Pickup and Delivery Problem," Transportation Science, vol. 29, pp. 17–29, 1995.

(4) A. Fabri and P. Recht, "Online dial-a-ride problem with time windows: an exact algorithm using status vectors," Operations Research Proceedings, 2007, vol. 2006, no. XV, pp. 445–450, 2007.

(5) Jaw, J.J., Odoni, A.R., Psaraftis H.N., and Wilson, N.H.M. "A Heuristic Algorithm for the Multivehicle Advance Request Dial-a-Ride Problem with Time Windows." *Transportation Research Part-B Methodological 20, no. 3*, 1986, pp. 243-257.

(6) Cordeau, J.F. and Laporte, G., "The Dial-a-Ride Problem (DARP): Variants, modeling issues and algorithms", *4OR: A Quarterly Journal of Operations Research*, Springer, Berlin / Heidelberg, 2003, pp. 89-101.

(7) C. L. Schweiger, "Deployment of technology for paratransit: Examples of innovation," Vehicle Navigation and Information Systems Conference, 1996. VNIS'96, pp. 197–204, 1996.