SYNTHETIC SPEECH FOR REAL TIME DIRECTION-GIVING

Christopher M. Schmandt and James Raymond Davis
The Media Laboratory
Massachusetts Institute of Technology

Abstract

The **Back Seat Driver** is a research prototype of a system to use speech synthesis as a navigational aid for an automobile equipped with localization equipment. We are evaluating the user interface by field trials. As this is work in progress, this paper will primarily give an overview of the system and describe its components. Included will be discussion of the map database, route finding algorithm, repair strategies, and the discourse generator.

With advances in navigation technology and automotive electronics[3,8] has come increasing interest in cars that know where they are and can help you figure out how to reach your destination. Most prototype projects have used various forms of display to present this information, and not all of them have included route finding ability[2,5,7,10,12,13,14,15,18,20,21] For safety reasons, a display may not be particularly suited to this task, moreover there is some evidence that drivers do better following spoken directions than reading maps [19]. Our project, the Back Seat Driver, uses synthetic speech to give driving directions in real time. It plans a route, talks the driver through the route, and not only warns the driver when she has made an error, but also plans an alternate, corrective route.

This paper is an overview describing work in progress. We hope to publish more detailed explanations of the various portions at a later date. At the time of this writing (June, 1989) we have a working system on the road and are simultaneously conducting field trials and improving the direction giving ability and database. Although we do not aspire to prove that voice is better than graphics for direction giving, we do aim to build an optimal system. Early results are very encouraging, suggesting that speech may prove to be a powerful technology in automobiles of the future.

Talking about Directions

There are many factors which contribute to good route description by people, some of which our system only touches on. The problem is complex and simple solutions are not likely to produce comfortable interfaces.

A good route is not simply the shortest, but is more likely to be a combination of the fastest and the easiest to follow. "Easiest to follow" will, however, differ between directions given in advance and directions given in real time by a fellow passenger. Directions given in advance (as e.g. by [4], or the system at Hertz rental counters) must be simple, because the driver alone has the burden of interpreting and following the directions, and there is no help if the driver gets lost. When the direction giver is in the car it is practical to use minor streets or short cuts.

Good directions take into account conceptual *portions* of a route, which make it easier of the driver to keep track of her location on a more global basis. These may include named neighborhoods, types of neighborhoods (business, residential, parks) and types of roads (expressways, parkways, "main" roads, twisty or narrow streets).

By way of example, one of the authors was recently given directions at a car rental counter in a city new to him. The agent at the counter said *"As you leave the airport, keep bearing to the right. You'll go around the end of the runway and see signs for the Interstate north."* The "computerized driving directions" printed at the counter described the same route as 5 separate segments, with mileages and names for each. Especially as it was night, the latter were almost impossible to follow, while the former had succinctly captured the salient aspects of the route.

When the directions are being given by a passenger, the real-time aspect becomes more important. Directions will be given just in time, taking into account vehicle speed, difficulty of the expected maneuver, driving styles, and road, weather, and traffic conditions. During long highway stretches with little need for description, the direction giver must maintain the driver's confidence. The passenger will also be watching for errors and trying to warn against them, again based on fine observations of the vehicle's speed and

Manuscript received June 9, 1989 0098 3063/89/0200 0649$01.00 © 1989 IEEE

direction. When a mistake is made, the passenger will tell the driver about it and together they will take corrective action (which is unlikely to be simply a sudden stop!).

Project Goals

The main goal of this project is to evaluate the utility of speech synthesis as the user interface to a real-time navigation system in an urban environment. Of particular concern is the discourse structure:

- how should driving acts be described?
- how can a description be generated from a route?
- how should timing considerations be applied?
- what kinds of feedback, both positive and negative, does the user require?
- what kinds of visual cues are most useful in describing an approaching location?

This information is gained from both laboratory simulations and field trials.

Our desire is to build the best possible real-time route describer. Although we believe a speech interface to the navigation unit is superior and safer than a visual interface, we do not plan to conduct direct comparison studies.

In the course of field trials to evaluate and improve our automatic direction giving, we hope to specify key components of the map database. We expect discourse behavior may need to vary with conditions (traffic, weather, day/night). It is likely that different visual cues may be useful in these situations. All must be represented in the database.

Geographic Database

Our database covers 41 square miles in the Boston area, including parts of Boston, Cambridge, Brookline, Somerville, and Watertown. It originated as a DIME (Dual Independent Map Encoding) file distributed by the United States Geological Survey[1]. A DIME file consists of a set of straight line segments, each with a name, a type, endpoints in longitude and latitude, and some additional information such as address numbers. Initially our database contained many errors. Correcting them required actually traveling most of the segments.

A DIME file alone is not sufficient for finding routes. The DIME files indicate physical connectivity, but route finding requires *legal* connectivity, i.e., one can legally drive from one segment to the next (one way streets are a simple example). We extended the data base format to explicitly represent legal connectivity. Since some streets are better than others, the database must include a measures of *quality*. We take this to be a largely subjective measure of the ease of locating and following a street. This allows the route finder to prefer to use streets of higher quality.

The generation of easily followed natural descriptions requires more extensions. We added a number of new segment types to distinguish bridges, underpasses, tunnels, rotaries, and access ramps. All these extensions were done for an earlier route finding project[4].

We are presently adding landmarks to the database. Drivers need landmarks to know how far to drive and when to turn. If the Back Seat Driver had eyes, it could simply choose landmarks as needed by looking for them in the landscape. Being blind, it must rely on landmarks coded into the map database. We have added traffic lights,.stop signs, and some buildings to the portions of the landmark database. A main task now is to determine what else must be added.

In addition to landmarks, other information is useful for providing assistance following a route. We found it very useful to add lane information, both number of lanes as well as any turn restrictions on lanes (e.g. left turn only). On short street segments, it is important to give lane advice (*"After the turn you'll want to get into the left hand lane."*) or else the driver may be unable to make the following turn. Lane warnings (*"Stay out of the left turn lane."*) are also important driving cues.

An interesting problem arises at complex intersections, typically a maze of ramps between major arteries, possibly at different elevations (see figure 1). Such intersections are typically not accurately recorded in the map. Furthermore, limitations in the resolution of the position tracking equipment make it difficult to distinguish one segment from another, especially as they are likely to diverge at narrow angles. The combination of uncertainties in the map and uncertainties in position make it difficult to give a clear spoken directions. Fortunately these intersections are usually well signed, so the Back Seat Driver can give directions by referring to the signs, e.g. *"Follow the signs to the expressways and airport"*. The wording of these signs needs to be in the database. It is important that Back Seat Driver's understand what the sign says, not simply utter the words. There are two reasons for this. First, our internal representation for text is based on syntactic structure, not text strings. Second, the objects mentioned in the signs (cities and roads) should be entered into the discourse model. They should become salient for future reference. This means that the text of a sign must be parsed, so that e.g. the sign text "Cambridge, Somerville, and Storrow Drive" should become a conjunction of the two cities "Cambridge" and "Somerville" and the street named "Storrow Drive".

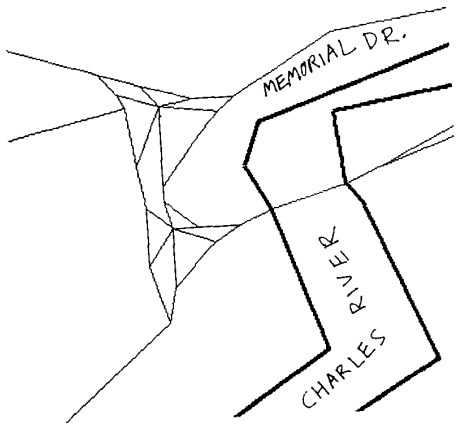

Figure 1: Access ramps at an interchange

System

Our vehicle is equipped with a localization unit built by NEC Home Electronics, Ltd., the project sponsor. It is a dead-reckoning position keeping system which uses speed and direction sensors. To compensate for error, it uses map matching on a map database stored on CD ROM. The system described more fully in [16,17]

As this is a research prototype, much of the computation is done in a base station computer laboratory (on a Symbolics Lisp Machine), rather than a computer on the vehicle. Two cellular telephones link the computer to the car. The on board navigational hardware transmits position and velocity via modem and cellular phone to the base station. The base station computer does all route planning and discourse generation. Speech synthesis is performed in a commercial text-to-speech synthesizer (Dectalk) cabled to the Lisp Machine. Synthesized instructions to the driver are relayed via the second cellular link and a speaker phone in the car. The keypad of the second phone also serves as the driver's control unit for the Back Seat Driver. Through this keypad a driver selects a destination, requests repeats of spoken information, and accesses other services of the Back Seat Driver.

A block diagram of the system appears in figure 2.

Routes

The Back Seat Driver can find the shortest, fastest, or most easily followed route. Route finding uses an A* search algorithm[11]. Depending on the driver's preference, one of three cost metrics is used, The *distance* metric is simply the sum of the lengths of the segments which compose a route. The *speed* metric scales each distance by a factor dependent on street quality. In addition, penalties are incurred for turns (more for left turns than right), stop signs, and traffic lights (which may be red). The *simplicity* metric, following [6], seeks to minimize the number of turns by imposing a distance penalty for each turn.

Discourse Strategies

The instructions are detailed and natural, and include a rich taxonomy of driving verbs. The dialog system uses cues such as vehicle speed and difficulty of driving actions to attempt to deliver instructions at the proper pace and in a timely manner. In addition, the system can anticipate some of the driver's possible mistakes and give warnings to avoid them.

Describing a route requires going from a series of segments (typically city blocks) in the database to a series of travel segments which will be separated by decision points. For example, going straight down a main street for five blocks will not be thought of by the driver as five separate acts, but rather one street traversal. A key piece of this analysis is that the need to make a decision is based on knowledge of what is *obvious*. Drivers do not want to be nagged at each corner to continue straight, but when they come to a questionable fork in the road they do want to be told which way to proceed.

If the driver does make a wrong turn, or misses a turn, the Back Seat Driver describes the error and then incrementally calculates a new route, rather than simply backtracking to the point of the error. Route planning includes weighting for length of the trip, difficulty of driving maneuvers (such as left turns against traffic), street quality, and complexity of the spoken directions.

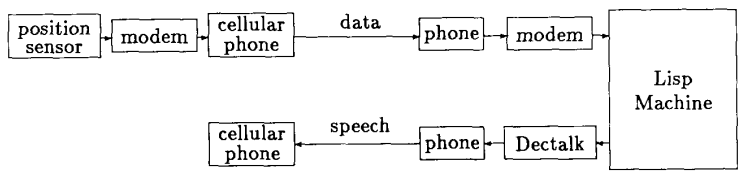

Figure 2: Communications block diagram

As opposed to much prior work in discourse generation, the Back Seat Driver is a real-time system which must factor in a number of temporal considerations. It needs to give each stage in the directions at just the right point, in terms of the time it takes to execute the driving maneuver as well as the speed of the vehicle approaching the intersection. For safety considerations, we would rather err on the side of giving the driver plenty of warning, but a cue given too far in advance may be miscued (e.g., a turn taken at an earlier intersection). Additionally, the software must consider the length of time it will take to recite an utterance. It is better to miss a turn and plan a new route than start describing the turn at a time when it may be unsafe to execute it (i.e., already well into an intersection).

There are several reasons to give instructions before the act, if time permits. One is to allow the driver to hear the instructions several times, and the other is to allow time to prepare for some acts, e.g., turns from a multi-lane street. These advance notices are lower priority than the description of the act itself, according to an internal set of system goals. Thus, they can be presented if there is adequate time, but will be ignored if the vehicle is approaching the next decision point too quickly.

Reassuring

While the driver is following a route, the system adopts a persistent goal of keeping the user reassured about her progress and the system's reliability. If Back Seat Driver were a human, this might be unnecessary, since the driver could see for herself whether the navigator was awake and attending to the road and driver. But the driver can not see the system, and so needs some periodic evidence that the system is still there.

One piece of evidence is the safety warnings the system gives (e.g. *"slow down"* before a turn), but if all is going well, there will not be any. The system gives two other kinds of evidence that things are going well. First, when the user completes an action, the system acknowledges the driver's correct action, saying something like *"nice work"* or *"good"*. This feature is very popular with most test drivers.

The second form of evidence is to make insignificant remarks about the roads nearby, the weather, and so on. If the driver assumes that the navigator is being cooperative, as set out in Grice's maxims of cooperative conversation [9], then the driver can infer that everything is going well, for otherwise the navigator would not speak of trivial matters. It isn't clear, however, that one really wants a chatty speech synthesizer. Certainly this feature could be useful in a rented car in a new city, where it might actually have some interesting things to say.

Summary

The following is one of the more complex utterances of the Back Seat Driver to date. It summarizes many key points mentioned in this paper, and indicates the current state of operability of the discourse generator:

> Get in the left lane because you're going to take a left at the next set of lights. It's a complicated intersection because there are two streets on the left. You want the sharper of the two. It's also the better of them. After the turn, get into the right lane.

The Back Seat Driver is already working in prototype form. Our present concerns are to determine what a spoken driving assistant should say, to understand how time and speed affect this decision, and to learn what features a map database must have to support generation of instructions.

Acknowledgments

The authors wish to gratefully acknowledge the support of NEC Home Electronics, Ltd.

References

[1] *Geographic Base File GBDF/DIME: 1980 Technical Documentation.* U.S. Department of Commerce, Data Users Services Division, 1980.

[2] Peter Braegas. Function, Equipment, and Field Testing of a Route Guidance and Information System for Drivers (ALI). *IEEE Transactions on Vehicular Technology*, VT-29(2):216–225, May 1980.

[3] Donald F. Cooke. Vehicle Navigation Appliances. In *AUTO CARTO 7 International Symposium on Automation in Cartography*, pages 108–115, March 1985.

[4] James R. Davis and Thomas F. Trobaugh. *Direction Assistance.* Technical Report 1, MIT Media Laboratory Speech Group, Dec 1987.

[5] Ronald A. Dork. Satellite Navigation Systems for Land Vehicles. In *IEEE Position and Location Symposium*, pages 414–418, 1986. IEEE 86CH2365-5.

[6] R. J. Elliot and M. E. Lesk. Route finding in street maps by computers and people. In *Proceedings of the National Conference on Artificial Intelligence*, pages 258–261, 1982.

[7] Edward J. Krakiwsky et al. Research into electronic maps and automatic vehicle location. In *AUTO CARTO 8 International Symposium on Automation in Cartography*, 1987.

[8] Robert L. French. Automobile Navigation: Where is it going? In *IEEE Position and Location Symposium*, pages 406–413, 1986. IEEE 86CH2365-5.

[9] H. P. Grice. Logic and conversation. In Cole and Morgan, editors, *Syntax and Semantics: Speech Acts*, pages 41–58, Academic Press, 1975.

[10] Peter Haeussermann. *On Board Computer System for Navigation, Orientation, and Route Optimization.* Technical Paper Series 840483, Society of Automotive Engineers, 1984.

[11] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on SSC*, 4:100–107, 1968.

[12] Stanley K. Honey, Marvin S. White Jr., and Walter B. Zavoli. Extending Low Cost Land Navigation Into Systems Information Distribution and Control. In *IEEE Position and Location Symposium*, pages 439–444, 1986. IEEE 86CH2365-5.

[13] Toshiyuki Itoh, Yasuhiko Okada, Akira Endoh, and Kenji Suzuki. *Navigation Systems Using GPS for Vehicles.* Technical Paper Series 861360, Society of Automotive Engineers, 1986.

[14] M. D. Kotzin and A. P. van den Heuvel. Dead Reckoning Vehicle Location using a solid state rate gyro. In *Proceedings of the 31rst IEEE Vehicular Technology Conference*, pages 169–172, April 1981. IEEE publication 81CH1638-6.

MR. SCHMANDT received his B.S. in Computer Science from MIT and an M.S. in computer graphics from MIT's Architecture Machine Group. He is currently a Principal Research Scientist and director of the Speech Research Group of the Media Laboratory at M.I.T.

His research interests are focused on interactive computer systems and human-interface issues of synchronous and asynchronous communication. His work emphasizes voice interaction for telecommunication based applications, with a goal of describing and then emulating human conversational behavior.

[15] Ernst Peter Neukirchner and Wolf Zechnall. Digital Map Data Bases for Autonomous Vehicle Navigation Systems. In *IEEE Position and Location Symposium*, pages 320–324, 1986. IEEE 86CH2365-5.

[16] Osamu Ono, Hidemi Ooe, and Masahiro Sakamoto. CD-ROM Assisted Navigation System. In *Digest of Technical Papers*, pages 118–119, IEEE ICCE, 1988.

[17] Osamu Ono, Hidemi Ooe, and Masahiro Sakamoto. Navigation and communication system. In *Digest of Technical Papers*, IEEE ICCE, 1989.

[18] Otmar Pilsak. *EVA: An electronic Traffic Pilot for Motorists.* Technical Paper Series 860346, Society of Automotive Engineers, 1986.

[19] Lynn A. Streeter, Diane Vitello, and Susan A. Wonsiewicz. How to tell people where to go: comparing navigational aids. *International Journal of Man/Machine Systems*, 22(5):549–562, May 1985.

[20] Katsutoshi Tagami et al. *Electro Gyro-Cator: New Intertial Navigation System for Use in Automobiles.* Technical Paper Series 830659, Society of Automotive Engineers, 1983.

[21] Walter B. Zavoli and Stanley K. Honey. Map Matching Augmented Dead Reckoning. In *Proceedings of the 35th IEEE Vehicular Technology Conference*, pages 359–362, 1986. IEEE CH2308-5.