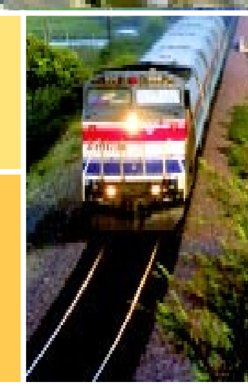
# Translating GIS Street Network Files
## for Use with Paramics


www.ncgia.ucsb.edu/vital




VITAL


Vehicle Intelligence and Transportation Analysis Laboratory
National Center for Geographic Information and Analysis
University of California, Santa Barbara

California Department of Transportation
Traffic Operations

# Translating GIS Street Network Files for Use with Paramics

Final Report

Contract 74A0067

Richard L Church, PI
Val Noronha, Project Director

2003 June 30

Vehicle Intelligence & Transportation Analysis Laboratory
NCGIA/Department of Geography, University of California
Santa Barbara CA 93106-4060 USA
www.ncgia.ucsb.edu

**Technical Report Documentation Page**

| | | |
|---|---|---|
| 1. Report No.<br>CALTRANS/CA/TO-03/02 | 2. Government Accession No. | 3. Recipient's Catalog No. |
| 4. Title and Subtitle<br>TRANSLATING GIS STREET NETWORK FILES FOR USE WITH PARAMICS | | 5. Report Date<br>June 2003 |
| | | 6. Performing Organization Code |
| 7. Author(s)<br>Richard L Church, Val Noronha | | 8. Performing Organization Report No.<br>51-356-936701-3OPIN |
| 9. Performing Organization Name and Address<br>University of California, Santa Barbara<br>Department of Geography, 3611 Ellison Hall<br>Santa Barbara CA 93106-4060 | | 10. Work Unit No. |
| | | 11. Contract or Grant No.<br>74A0067 |
| 12. Sponsoring Agency Name and Address<br>California Department of Transportation<br>Sacramento CA 95814 | | 13. Type of Report and Period Covered<br>Final |
| | | 14. Sponsoring Agency Code |
| 15. Supplementary Notes<br>In conjunction with U.S. DOT project DTRS-56-00-T-0002 (Research and Special Programs Administration). | | |

16. Abstract

One of the principal impediments in traffic microsimulation is the creation of detailed road infrastructure data. In Paramics, the data files are entitled "nodes," "links," "categories," etc. Much of the required data exist in DOT GIS files, but GIS employs a different geometric model and the data are not easily exchanged. This project addresses the conversion problem, performing methodological research on the semantic aspects of the problem, and creating a working tool, Shape to Paramics (S2P). S2P offers three different methods to translate the GIS polyline model to the Paramics line-arc model. The first is line generalization, which achieves impressive data reduction but can create abrupt angular turns. The second method, applicable only to some roads, is to fit a singular circular arc to the original polyline. The third method is to interpolate biarcs, a series of circular arcs and straight lines. GIS data vary considerably in quality and specifications, and there are situations where manual editing is required on either the input or output of S2P; but for the most part the tool converts GIS files easily and accurately. The project incorporates a test component that documents the performance of the tool.

| | | | |
|---|---|---|---|
| 17. Key Words<br>GIS, centerlines, shape files, Paramics | | 18. Distribution Statement | |
| 19. Security Classif.(of this report)<br>Unclassified | 20. Security Classif.(of this page)<br>Unclassified | 21. No. of Pages<br>41 | 22. Price |

Form **DOT F 1700.7** (8-72)

# Table of Contents

# *List of Illustrations*

# *List of Tables*

# *Glossary*

| | |
|---|---|
| CAD | Computer Assisted Drafting |
| Caltrans | California Department of Transportation |
| DMI | Distance Measuring Instrument |
| DOT | Department of Transportation |
| ECE | Electrical and Computer Engineering |
| ESRI | Environmental Systems Research Institute, Redlands CA |
| GDT | Geographic Data Technologies Inc |
| GIS | Geographic Information System |
| ITS | Intelligent Transportation Systems |
| NCGIA | National Center for Geographic Information and Analysis, UCSB |
| RS | Remote Sensing |
| UCB | University of California, Berkeley |
| UCSB | University of California, Santa Barbara |
| UTM | Universal Transverse Mercator |
| VITAL | Vehicle Intelligence and Transportation Analysis Laboratory, UCSB |

# *Disclaimers*

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the STATE OF CALIFORNIA or the RESEARCH AND SPECIAL PROGRAMS ADMINISTRATION. This report does not constitute a standard, specification or regulation.

# Acknowledgements

This research was supported by the California Department of Transportation (Caltrans), under Interagency Agreement: Transportation Master Research Agreement No 65A0108.

The research was related to and benefited from research conducted by the National Consortium on Remote Sensing in Transportation (Infrastructure Management), at the University of California, Santa Barbara, supported by the United States Department of Transportation (Research and Special Programs Administration) and the National Aeronautics and Space Administration (NASA), contract DTRS-00-T-0002.

In the conduct and execution of the research we are grateful for the contributions of Professors Michael F. Goodchild (Department of Geography) and Yuen Fang Wang (Department of Computer Science). We also acknowledge the assistance of Ting Lei, who developed the biarc interpolation routine. Francisco Iovino, Qin Zhang and Xiaohang Liu were employed as programmers under the project. These are all personnel of the University of California, Santa Barbara. In addition, Yonnel Gardes at the University of California, Berkeley, performed the evaluation and testing.

# 1 — Introduction

Traffic microsimulation modeling software such as Quadstone's Paramics® has considerable potential in improving transportation planning, operations and emergency management. However it requires extremely detailed data on road geometry, structure (number of lanes), speed limits, etc, as well as demand and trip generation. The potential of these tools is limited by the availability and quality of the input data, and this is currently a real impediment to the widespread use and benefit of microsimulation.

In many cases the requisite data do exist in various DOT offices, for example road geometry is archived in the design and construction department, lane information and speed are available from maintenance records. Partly for historical reasons, the data are stored differently, and the software systems related to them do not readily communicate with each other. For example, road geometry is traditionally created and stored in computer-assisted drafting (CAD) files with local coordinate systems; for planning purposes, road centerlines are usually re-surveyed and stored in geographic information system (GIS) files, typically tied to regional or global coordinate systems (e.g. latitude-longitude, state plane or Universal Transverse Mercator, UTM); and maintenance data reside in linearly referenced files in which location is captured as route ID and distance offset rather than coordinates.

For its part, Paramics treats road sections as straight line segments or circular arcs — these are amenable to modeling driver response to curvature, but the geometric representations are so different from those used in CAD and GIS files that they are incompatible. The result is that the road geometry files required for Paramics are created by laborious manual effort, digitizing lines and curves with reference to plans and aerial photographs. When modeling small sections of highway, this is not a major problem, because the manual process of re-creating road geometry is not prohibitively time consuming. However, when modeling extensive and complex networks, where there are tens or hundreds of neighboring arterial and collector streets that impact traffic on highways, hand coding of the street and highway system is a problem. Modeling off-highway traffic, such as emergency evacuation of neighborhoods, also requires the coding of large street network databases.

For microsimulation to take advantage of the wealth of data already resident in GIS and CAD systems, the incompatibility between data models and structures must be overcome. CAD and GIS software packages typically include import filters that allow interchange of data; a similar filter must allow the import of GIS data into Paramics.

ArcView® from Environmental Systems Research Institute (ESRI) is one of the most popular basic GIS packages available, and the ArcView Shape file is a widely used GIS format, that represents a variety of data: points, lines, polygons and others (street network files fall under the "line" category).

The objective of this project is to facilitate the translation of files from ArcView to Paramics. The work covered by the contract is the second phase of study. In Phase I, at no cost to Caltrans, UCSB created a rudimentary tool to accomplish a GIS-Paramics translation. This exploratory work identified areas in which the tool would have to be improved to be of practical value, for example, to handle ramps, and to take advantage of attributes in the DBF component of the shape file. Phase II, the current phase, set out to create a useful, working tool. At the outset it focused on three principal issues. Superficially, translation can be reduced to a matter of **syntax**, but as the body of the report describes, there are issues of computational performance and traffic throughput, that are **semantic** rather than syntactical. Accordingly a considerable portion of the research effort was devoted to finding appropriate semantics of translation, and a testing component of Phase II (performed by UC Berkeley) was designed to document this. A third issue in the project, the one most visible to the user, is the **user interface**.

The principal deliverable of this project is a software tool, nicknamed S2P (Shapefile to Paramics), which inputs a GIS shape file, processes it according to user preferences, and outputs a set of files for Paramics. Software documentation is part of the deliverable.

It should be noted that in at least two respects, the project is somewhat unbounded. The first is the user interface, which can be polished *ad infinitum* with elaborate help systems, etc, but this was not the scope of the project. The other is the issue of data adequacy. GIS data contain inaccuracies and errors, some as a consequence of the scale of the data (for example, 1:20,000 data are generally accurate to ± 10m and clearly do not stand scrutiny at the sub-meter level), and some that originate with poor data gathering and quality control. There are also differences in the way data are represented by vendors, differences in datums and coordinate systems, etc. While it is possible to design software to deal with discrepancies and errors, this functionality is best left to the GIS that generates the Shape file for translation. S2P is limited to providing functionality that is not contained in mainstream GIS, and that is essential to the translation.

There are two notable areas in which Phase II does not attempt to provide solutions. One is linear referencing, which is the system used for referencing a wealth of useful highway data (e.g. number of lanes, speed limits, surface type and quality). If this were incorporated into the tool in the future, it would extend the ability to utilize existing Caltrans “post-mile” data sets. A second area of potential future development is the incorporation of data such as elevation and visibility, that can be derived by GPS and remote sensing, and which with appropriate modeling can improve the capability and realism of microsimulation. Similarly, grade and cross-slope can be incorporated into the modeling. UCSB has considerable expertise in the application of GPS and remotely sensed data to transportation, and leads the National Consortium on Remote Sensing in Transportation — Infrastructure. It has developed a number of data sets that could be used for the development and testing. These are directions for a possible Phase III.

Even without these enhancements, the current product stands to improve the capability of Caltrans in significant ways, dramatically reducing the amount of time required to prepare microsimulation files. More important, it will enable microsimulation on large networks such as entire cities or counties, that have until now been impractical to code by hand. S2P outputs will have to be edited in some areas, where there are unusual street configurations, or where there are data errors or unusual coding practices in the GIS file. This is partly due to the inevitable disconnect between multipurpose GIS files, which are available on a continuum of scale and accuracy, and Paramics needs, which require a specific scale and accuracy level. Many GIS files contain insufficient detail; some of the high end products may contain too much detail.

This report is presented in four sections following this introduction. Section 2, Tool Design, describes the semantic translation problem in detail, and the approaches used to address it. It also presents the syntax and user interface issues that have been addressed in the software. Section 3 discusses some of the problems with input data, and how to select shape files appropriate for translation for microsimulation. Section 4 consists of Conclusions and Recommendations. It summarizes test results and presents a set of recommendations for future research. Section 5 briefly discusses implementation issues.

Appendix A contains an evaluation report on the software by Yonnel Gardes, at the University of California, Berkeley (UCB), who has considerable experience creating Paramics files manually. Due to the parallel schedules of evaluation and additional software development, the evaluation report is based on Version 2.0e of S2P, and many of the problems identified in the report have been corrected in Version 3. A section at the end of the evaluation report briefly reviews changes in Version 3.

A user's manual is presented as a separate volume.

Because the tool has obvious utility to a wide variety of potential users, UCSB has elected to make the executable software, user's manual, evaluation report and other files available at the project web site: *www.ncgia.ucsb.edu/ncrst/research/microsimulation*. In doing so, we extend the community of potential users, and gain feedback on future directions of development.

# 2 — Tool Design

At a technical level there are 3 components to the project:

- Semantics: the mathematical process of converting GIS shapes into Paramics-compatible line segments and circular arcs, interpreting road geometry appropriately;
- Syntax: expressing data in a format readable by Paramics, with special attention to ramps and directional streets
- User interface

## Semantics

In **GIS**, a road centerline is typically represented as a list of (*x*,*y*) points that approximates the path of the longitudinal axis of a road. The first and last points are usually intersections, and are called nodes, while the remaining points serve only to define the shape, and are called shape points. In Figure 1, the solid points A and B are the extremities of the polyline AB, while the open circles are the shape points that define the road geometrically. Road centerlines may have several hundred shape points, depending on the shape and the length of the segment. The road section AB corresponds to one record in a GIS database. In older GIS implementations, if any attribute changed, the geometry would be fragmented to correspond to the database. For example, if the number of lanes changed at the first shape point, that shape point would be considered a node, splitting AB into two polylines, associated with separate database records. Modern systems avoid this by using dynamic segmentation and linear referencing.

A "shape file," a GIS format introduced by Environmental Systems Research Institute (ESRI), actually consists of three related files with extensions SHP, SHX and DBF. The .SHP contains the geometric definition of geographic features, in this case polylines; the .SHX is an index file that facilitates retrieval of shape data, and the .DBF is a standard dBASE®-format database containing attributes of each shape.

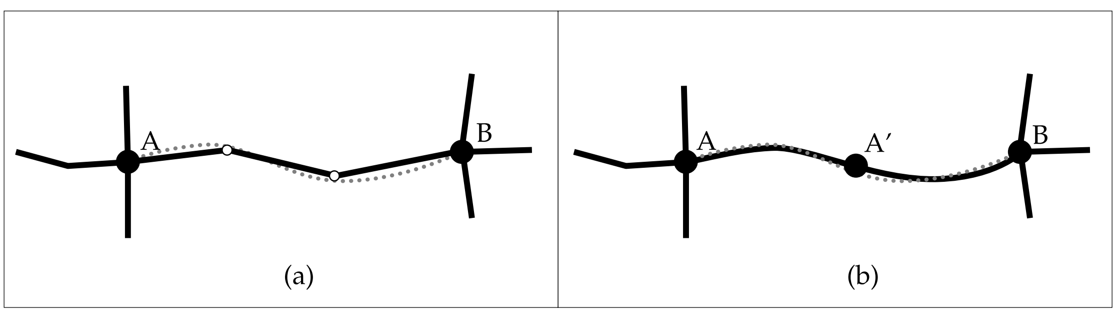



**Figure 1**. Polylines and curves. Road from A to B, represented by (a) a single polyline with two intermediate shape points, and (b) two circular arcs, AA′ and A′B. In neither case does the representation necessarily follow the exact path of the road (stippled line) — this depends upon the standard of exactness.

The **Paramics** representation of a road more closely mimics the engineering design, where the centerline is a combination of straight line segments and circular arcs — in the computational geometry literature, such combinations are termed biarcs. There is no provision for shape points. Therefore where a road section cannot be geometrically defined by a single line or circular arc, it is broken into multiple entities creating intermediate nodes (e.g. A′), for strictly geometrical purposes. *In addition*, entities are fragmented when attributes change, as with older GIS polylines. This creates two degrees of redundancy and imposes undesirable computing overhead.

When converting GIS shapes into Paramics biarcs, each segment of each polyline could in theory be translated into a Paramics straight line link, but this would create a large Paramics file with hundreds of links defining each road. Two competing objectives must be traded off:

- To minimize run-time performance penalty in both computing time and simulated traffic throughput, by minimizing the number of line/arc segments that represent a polyline;
- To maximize positional fidelity between the polyline and biarc geometry.

There are several potential geometric solutions to the polyline-biarc conversion problem, detailed below, and it is up to the user to control the methods and parameters, depending on the type of output desired. By providing an appropriate user interface, S2P offers a conversion process that is accurate, efficient and flexible.

## Generalized Polylines

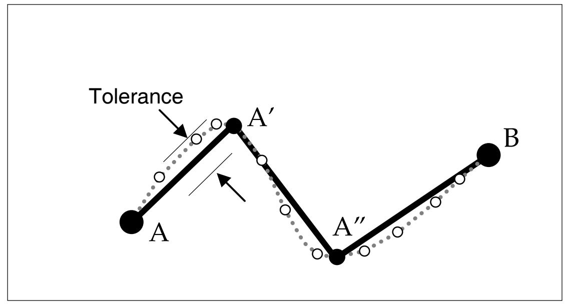



**Figure 2**. Generalization. Original (stippled) and generalized (heavy) polyline in which a subset of the original shape points are selected

A simplification of the problem, that was employed in Phase 1, was simply to reduce the number of shape points. This is illustrated in Figure 2. A polyline with considerable geometric detail (stippled), 12 shape points + 2 nodes A and B in this example, is reduced to a minimal set of points that describe its basic shape (solid line: 2 shape points + 2 nodes A′ and A″). The generalization algorithm by Douglas and Poiker (1973) is an effective way to accomplish this, selecting points that most effectively define the shape within a given lateral tolerance. Shape points that fall within the tolerance are considered redundant, and deleted. Paramics software performance improves because there are far fewer points to be processed, but some positional accuracy is sacrificed. There are several intermediate degrees of generalization between the two forms shown in Figure 2, controlled by setting the lateral tolerance appropriately. This is true whether the polyline is reduced to a coarser polyline (as shown) or biarcs as in Figure 1(b).

An additional problem, particularly serious for microsimulation, is the loss of tangent continuity between sections of a generalized polyline. The trade-off between geometric accuracy and Paramics performance depends on the application and context, and is ultimately to be decided by the user.

There are two motivations for preserving the original polyline shape as faithfully as possible, i.e. to setting a low lateral tolerance and minimizing the amount of generalization. One is purely esthetic: to the extent that microsimulation assists at public presentations to various stakeholder constituencies, the effect of highly generalized, angular polylines can be prejudicial to the impression of quality. The second issue is that geometry, particularly continuity, impacts simulated traffic throughput. Straight lines are shorter than curves, and easier to negotiate. Moreover, at every node, Paramics establishes start-line and stop-line vectors. Where there is a significant change in orientation of these vectors, Paramics recognizes the intersection as a turn, and vehicles are made to slow down to negotiate the turn. To avoid artificially impacting traffic throughput, tangent continuity between adjacent biarc segments must be enforced. Using polylines, this usually means preserving a higher density of shape points around curves — the Douglas Poiker algorithm usually does this. Alternately the problem can be addressed by (a) beveling sharp corners by appropriate positioning of kerb points, or (b) filleting corners using circular arcs.

## Circular Arcs

A single circular arc can sometimes be an appropriate generalization of a polyline, particularly given that roads are designed in straight and curved sections. This is particularly true of entrance and exit ramps. Circular arcs have the advantage of being less data intensive than polylines; a well-placed arc can replace a dozen straight line segments. An arc approximation method was developed for S2P, using a simple geometric routine to fit a circular arc to three points, checking that every point in the polyline falls no further than a given tolerance from the arc. This method seems more robust than the alternative of computing a least squares circular arc. Polylines that fail the tolerance test are processed by the Douglas Poiker algorithm.

The caveat is that when a polyline is represented by a single circular arc, the orientation of the road segment at the intersection may change significantly, impacting the accuracy of traffic simulation. A low tolerance is required to keep this from happening, and this results in few polylines being reduced to a circular arcs, most being processed by the Douglas Poiker algorithm instead.

### Biarcs

A comprehensive geometric solution clearly involves an approximation as a combination of geometric objects. The main objective is to represent road centerline data as a combination of straight line segments and circular arc segments, where tangencies must coincide at boundary or end points of each segment (circular arc or straight line segment). The tangency requirement is perhaps the most difficult issue in translating a polyline into a set of simple geometric objects. The research found, however, that this problem is not unique to the problem of translating road centerline data, but is a general problem in computer aided design and control. A literature research determined that, for example, computer controlled lathes and milling machines require the decomposition of a smooth curve into a set of connected circular arcs and straight line segments, that resemble the tangency conditions required in representing road centerline data for Paramics. This type of problem is called a biarc approximation of polygonal curves. The biarc representation problem can be found in the computer aided design research literature (Parkinson and Moreton 1991; Rosin and West 1980; Meek and Watson 1995; and Yang 2002). Although there are a number of algorithms that have been developed for this problem, it is not clear which one might be the best overall. We selected for development the technique developed by Yang (2002). This technique utilizes a local biarc interpolation process which is called recursively and sequentially until a fit is produced within a pre-specified tolerance. This algorithm was implemented in the C++ programming language and incorporated into S2P. Alternate techniques for biarc interpolation do exist; however, the scope of this project limited their development and testing.

## *Syntax*

### Files

S2P creates six Paramics files: *nodes*, *links*, *categories*, *centers*, *linktypes*, *vehicles*. Paramics creates additional files (*version*, etc) when it is executed and fed the S2P outputs.

*Nodes, links* and *centers* contain the geometrical descriptions of the road centerlines. The *categories* file defines road categories based on speed limit, width, number of lanes and road type (rural, urban, highway). The variables are read from the shape DBF file if available, or default values are provided by the software if the database does not contain the required values. Databases label these data fields differently, therefore the user interface allows appropriate variables to be selected from the DBF file.

### Special Syntax

*One-ways.* By default, each street in the GIS file is coded as a 2-way street in the Paramics output file, e.g. a link A→B is written again in the direction B→A. To treat surface one-way streets and directional carriageways differently, the DBF file must provide information on traffic direction. Not all databases contain this, and S2P can therefore erroneously code ramps and other entities in both directions, when it is visually obvious that they are one-way. This cannot be avoided; while it is possible to make some inferences and characterize freeways based on speed limit, in cases where such a field exists in the attribute database, it would be extremely difficult to write code to detect traffic direction entirely by geometric analysis.

*Ramps.* Paramics uses special coding for ramps when they merge with freeways *and* there is no increase in the number of freeway lanes. Normally the operator places a node off the freeway, and invokes a command to merge it with the freeway. Paramics then creates the merge, and an acceleration lane. S2P detects whether there is a change in the number of lanes when a ramp meets a freeway, and codes the ramp appropriately, eliminating all manual steps. To do this, it requires that the DBF file identify freeways with an appropriate feature class code. Any non-freeway link that joins a freeway is merged downstream of the connecting node.

### *Interface*

S2P provides a map display with zoom and re-center abilities, to examine any part of the map and the visual effect of different generalization parameters. Input and output layers and nodes can be turned on and off.

#### Information Required and Produced by S2P

The following information is sought from the user:

- Input file name
- Type of polyline-biarc conversion, and generalization tolerance
- Units of measure
- Database field selection for road characteristics: speed limit, one way, number of lanes, width, type.

On the output side, the software creates Paramics files in a user-specified folder. It reports conversion statistics, in particular the number of nodes and links in the input and output files (which depends on the level of generalization employed) and the percentage gain or reduction in these numbers.

#### Attribute Data

Caltrans indicated in March 2003 that much of its data are from Geographic Data Technologies Inc (GDT), and that it would be particularly useful if S2P were "hardwired" for GDT. The GDT format is documented in the Dynamap user manual (GDT 2002). In street and highway files, the variable "ONE_WAY" indicates directional permission on a road, "FCC" is a feature class code that flags freeways and other recognizable road types, etc. However there are problems associated with hardwiring:

- Files could be mistaken for being GDT when they are not, by virtue of their names, or vice versa;
- Should the field values change in future editions of GDT data, or should Caltrans change its vendor, the hardwiring becomes ineffective.

Therefore we coded the properties of GDT in an initialization file, S2P.INI. By modifying this file it is easy to change the fields and values associated with various road characteristics. For example, the method by which the database encodes freeway sections is specified in S2P.INI as follows:

```
FIELD.FREEWAY=FCC,A15,A25
```

This means that the freeway flag is encoded in the variable "FCC" as A15 or A25. If a database from another vendor uses the variable name "ROADTYPE" and codes freeways as "F", the statement would read:

```
FIELD.FREEWAY=ROADTYPE,F
```

S2P.INI is a text file that can be modified with any editor. This method of field specification lends itself to any database vendor, while minimizing repetitive steps in the user interface. The INI file also contains user preferences such as generalization parameters and display colors.

# 3 — Input Data Quality

The output of S2P depends largely on the quality of input shape file data. Sample GIS files initially supplied by Caltrans for testing were of scale ~1:10,000 to 1:25,000, i.e. positional accuracy 5–10 m. This is typical of GIS files in use in most DOTs, and of those available from some commercial vendors. By contrast, Paramics files used for microsimulation tend to be in the 1–2 m accuracy range. This points to an important difference between input and output expectations. In this section we discuss some of the potential issues with input data, by way of guidance in choosing appropriate GIS databases. Noronha (1999), and Noronha & Goodchild (2000) describe the kinds and magnitude of error found in centerline databases.

## *Positional Accuracy and Scale*

Positional accuracy of a centerline is of limited importance in some GIS transportation applications, because road length (measured by a Distance Measuring Instrument or DMI) is stored explicitly in the attribute database, not calculated from coordinates. Measurement from coordinates can under-represent distance by as much as 15% (VITAL 2000).

In Paramics, distances are derived from the coordinates, and the accuracy requirement is consequently more stringent. The need for geometric accuracy may depend on the size of the study area. Over short sections of road, small differences matter; over larger areas the same absolute errors may be less significant. Curve geometry influences the speed at which a link is traversed, hence positional errors could impact microsimulation results.

## *Topological Accuracy*

Databases are known to contain errors of omission and even commission, and there may be topological errors in connectivity between roads. In some regards this is related to scale: short roads such as ramps and driveways may appear on detailed maps, but not on smaller scale maps. Clearly these database errors propagate into serious inaccuracies in microsimulation results. Figure 3 shows some examples of positional and topological error.

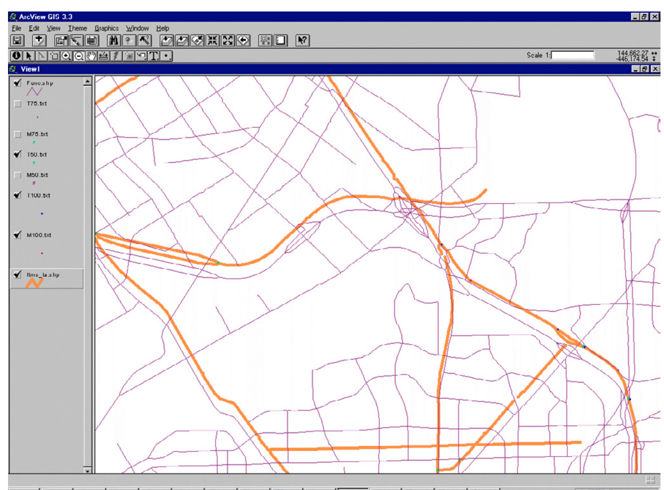


**Figure 3**. Los Angeles files files used in Phase 1 tests. Freeway layer (orange) and street layer (blue, also showing freeways) are in different positions. More important, note missing segments and lack of ramps in freeway layer, in intersection at northeast.

On reviewing the Los Angeles files (Figure 3) used in Phase 1, we find significant accuracy problems. In addition to positional errors, there are gross generalizations of highway features (i.e. a single line representing both carriageways, rather than separate east-bound and west-bound lines) and highway intersections (e.g. no ramps), missing segments, and coarse geometrical representations. Caltrans is in the process of developing higher accuracy centerline files statewide, and from discussions with the Office of Photogrammetry, it appears that the new databases are likely to be of better quality and more suitable to processing for microsimulation.

## *Carriageways and Lanes*

A carriageway can be defined as a physically distinct road track. A freeway typically has two carriageways, one in each direction. Paramics allows for directional carriageways to be coded as two-way roads, with a median distance specifying the separation between the carriageways.

GIS files do not follow this protocol. Older GIS files do represent freeways as single lines, but this is only for want of detail. Better quality files represent directional carriageways as separate centerlines. Other physically separate carriageways (e.g. HOV carriageways) may similarly have their own geometry. There may be variation in interpretation of an object, depending on the target market of the vendor and the coding practices of its principal customers. For example, to allow HOV lanes higher average speeds, a vendor may choose to represent them as geometrically separate centerlines, even though there may be no more than a yellow line separating them from SOV lanes. Other vendors may be more tuned to physical access, e.g. for emergency and service vehicles, and may separate centerlines only when there is a physical impediment to navigation, such as a Jersey barrier or median. As the technologies for gathering centerline data improve (e.g. with GPS and remote sensing), it is likely that databases will have more rather than less data, and the focus will shift towards carriageways or even lanes rather than roadways. The net effect is that a variable "median distance" in Paramics parlance is not a valid concept in most GIS files, however, this by itself does not affect the translation or outcome of simulation in any way.

Figure 4. Turnpike at Cathedral Oaks, Santa Barbara (intersection A in Figure 5), looking north. No physical separation of lanes.

A serious problem arises when the carriageway representation is extended to regular urban roads, as in the case of GDT data. GDT represents many roads with double-yellow-line dividers (e.g. Cathedral Oaks, Figure 4) as dual lines, perhaps to prohibit U-turns at random points along the road. Turning paths in intersections must then be explicitly represented as links (Figure 5), some of which can be as short as 2–3 m. Paramics cannot handle such short links and generates error warnings on input. Furthermore the short links may not be perfectly aligned, with the result that start and stop lines are misaligned in Paramics.

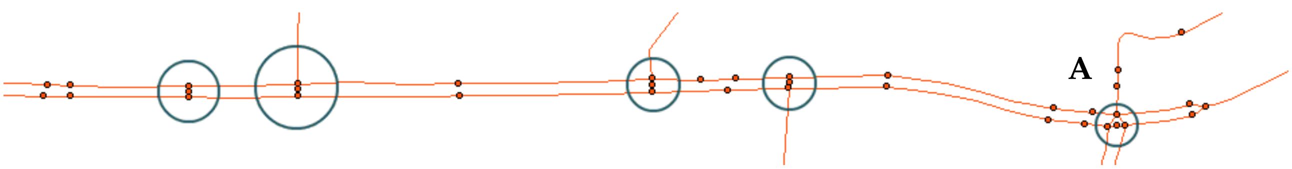



Figure 5. Cathedral Oaks, Santa Barbara, represented as separate carriageways in GDT. Short links (circled) connecting the carriageways are necessary to enable turns, but links of this length cause problems in Paramics.

This reflects an inevitable evolutionary point of conflict between GIS and microsimulation. GIS centerline data are vectors, whereas Paramics models roads as polygonal spaces. The translation of vectors to polygons at this level of spatial detail can result in degeneracy; this is why Paramics disallows short links. Automatic detection and topological reduction of dual-line representations to single-line representations is exceedingly complex and outside the scope of this project. The practical solution for the present is to edit either the GIS file or the Paramics file manually, to reduce the level of detail.

## *Segmentation*

For reasons that have to do with legacy GIS practice, some street databases (including GDT) segment roads at non-grade intersections. Figure 6 shows the centerline for the inferior road (solid yellow) with nodes at two overpass points. The superior roads (overpasses, stippled yellow) are broken at the same nodes. In GIS, these nodes are introduced as a consequence of "planar enforcement", used for coding polygons such as block groups. They are superfluous for transportation; the danger is that software that enforces planar geometry treats these nodes as at-grade 4-way intersections.

It is possible to address this problem by reconstituting such fragmented segments into continuous links, however the amount of effort required to write generic code to do this places it outside the scope of the current project. Spurious nodes have to be manually deleted, either in the GIS prior to processing by S2P, or in the Paramics file after S2P processing.

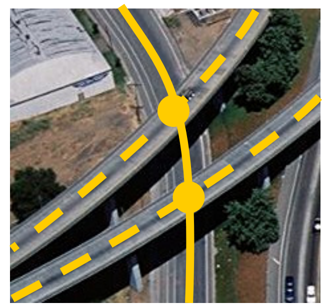


Figure 6. Link broken at overpasses

S2P does however use elevation information (if available in the shape file) to distinguish between inferior and superior nodes. In GDT data, superior nodes are coded with pseudo-elevations of 1, 2, etc, positive or negative, effectively an ordinal reduction of metric elevation. S2P multiplies these ordinal values by a user-selectable factor (e.g. 6 meters), (a) separating nodes so that there are no longer connected for traffic flow purposes, and (b) producing elevation values, completing the $(x,y,z)$ triplet for nodes in overpasses. Note that these elevations are relative, not absolute, and apply only at multi-level intersections. This means that an overpass that runs at a constant 10 m elevation over a 2 m freeway *incorrectly* shows a *gradient* up to the overpass point. The gradient could produce undesired effects on traffic behavior, and Paramics may reduce the gradient to a more realistic value on input.

### Coordinate System, Datum and Projection

Paramics uses a local coordinate system and is unaware of projection and datum. This can be a problem when data sets are merged, because it means that two files for the same area on different projections do not overlay visually. The solution is to transform one or both files using GIS software, so that there is agreement on datum and projection. S2P performs basic transformations of coordinate units, and projects latitude-longitude shape files to UTM.

### Conclusion

In summary, as with any such software, S2P outputs are strongly dependent on the GIS input data. There are several ways in which a transportation network can be interpreted, using few or many nodes, and single or multiple carriageways[1]. Street network files are evolving towards more complex representations. S2P can re-interpret the geometry of a given shape using generalized polylines and biarcs, but it does not carry this to the point of joining, merging and dissolving shape records because the algorithms to do this are non-trivial. To achieve a smooth translation from a complex shape file such as GDT, either the shape file must be edited prior to S2P processing, to remove unnecessary detail, or the Paramics files must be edited after processing. An example from the Fairview freeway overpass and intersection in Santa Barbara illustrates the problem and solution.

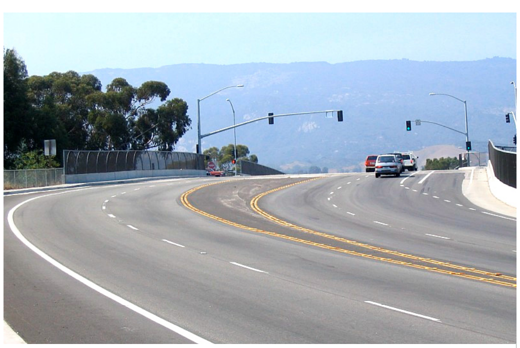


Figure 7. Fairview overpass: opposing traffic separated by double yellow line only.

The Fairview overpass (Figure 7) is represented in GDT as a double line, with numerous fragmented shapes (Figure 8), demonstrating issues described in two sections above ("Carriageways and Lanes" and "Segmentation"). Feeding this shape file directly into S2P, the output is obviously not acceptable, and requires extensive editing in Paramics.

[1] Object oriented data models such as UNETRANS are now capable of resolving these multiple interpretations, but this leading edge of GIS design is not reflected in generic shape file formats.

However, if the file is first edited in a GIS, reducing the double line overpass to a single line, reducing the number of nodes, and inserting as a DBF field the number of lanes on each shape, the output is perfectly acceptable and immediately usable in Paramics (Figure 9). Note that in the shape file used in Figure 9, extraneous nodes at non-grade intersections (discussed under "Segmentation" above) have not been removed in pre-editing; S2P separates them on output.

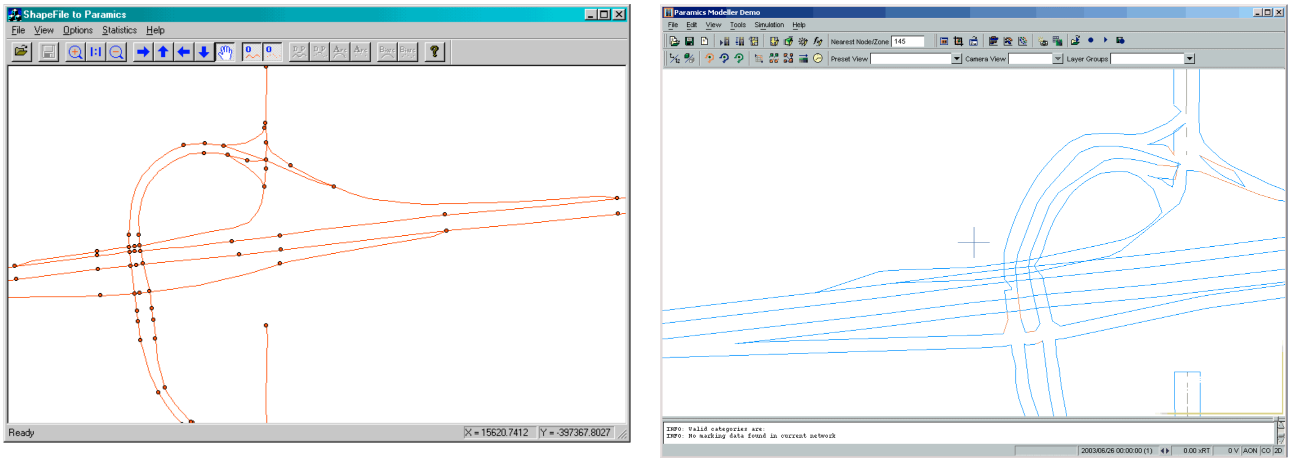



Figure 8. Fairview intersection: original GDT data and S2P output in Paramics. Extensive editing required in Paramics.

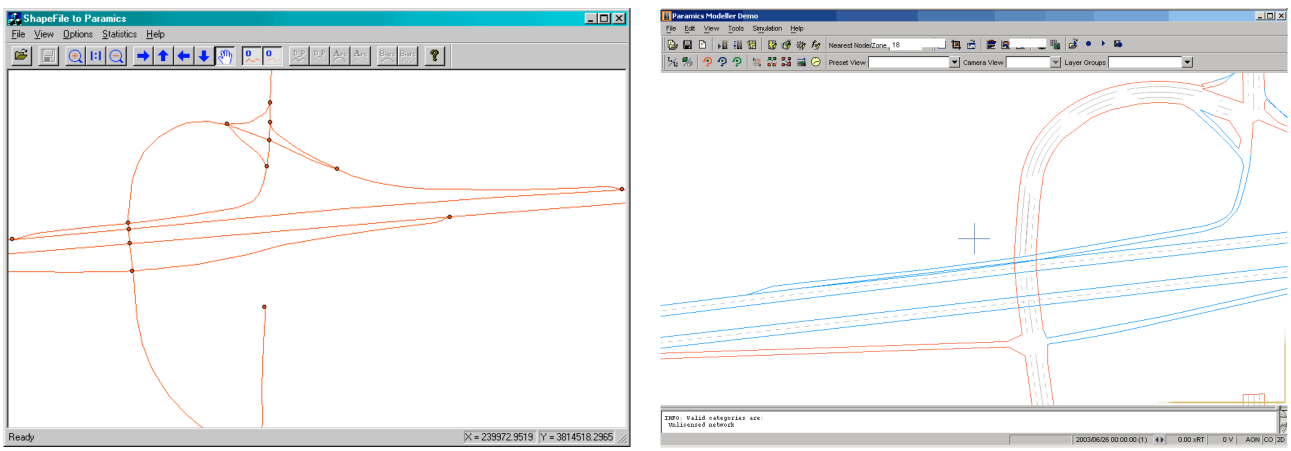



Figure 9. Fairview intersection with GDT data pre-edited, and number of lanes added to DBF.

Most recent GIS representations of street networks are as in Figure 9 (e.g. Thomas Brothers, Knopf Engineering). The highly detailed representation in Figure 8 is a relatively recent development among vendors serving the turn-by-turn vehicle navigation market (e.g. Navtech; GDT has migrated to this over the last 2-3 years). Unless it is possible to automate generalization of these turn-by-turn data topologically to Figure 9 levels, manual pre- or post-editing is necessary, and could multiply the amount of time required to code overpasses and other urban roads (long stretches of highway should require little or no editing).

A critical issue for Caltrans is where to acquire GIS files with the "right" amount of detail. GDT and Navtech are rich in attribute data (e.g. speed limit, feature class code) and reasonably accurate on coordinates and topology, whereas less detailed data may not contain required fields, and some data such as in the Los Angeles example are entirely unsuitable. As Caltrans develops its own road databases, we strongly recommend that Traffic Operations communicate these issues to the GIS division as factors to be considered in internal GIS database design and acquisition.

# 4 — Conclusions and Recommendations

Conclusions from the study fall into three categories: (a) the success of the academic research, (b) the total effectiveness of S2P at the practitioner's level, and (c) future directions.

## Academics

First, at the research level, the challenge was to achieve semantic correspondence between the GIS and Paramics interpretations of road centerlines. The conclusion is that there are several ways to translate polylines into Paramics compatible objects. A small selection of these were implemented as Models 1 and 2. Each method has its particular strengths, and is appropriate to a certain type of shape. Each method relies on a tolerance to specify how closely the Paramics shapes should resemble the GIS polylines. All three methods have proved to be robust and fast. A research question that was explored to some degree was how to select an appropriate tolerance for different types of applications and road shapes. Much more research could be done on this question.

## Effectiveness

The second category of conclusions pertains to the effectiveness of the software in handling a variety of roads, dealing with the special syntax of Paramics (e.g. ramps) and minimizing the manual edit required before the Paramics files are usable. This was addressed in the evaluation report prepared by UC Berkeley (Appendix A). Evaluation and software development proceeded in parallel, and the evaluation was consequently based principally on a dated release (Version 2.0e) of S2P. In some respects the conclusions presented in the evaluation were overtaken by software development — this was as intended, because the purpose of evaluation was ultimately to improve the product, not merely to review it. The evaluation process has some limitations because it was performed by researchers unfamiliar with the test area, without the benefit of aerial photographs or other information that would normally be used in data creation.

Inevitably and intentionally, the evaluation focuses on the *total* performance of S2P in converting a GDT shape file, from the viewpoint of a Caltrans practitioner. In doing so, it effectively tests the strengths and shortcomings of three products (GDT, S2P and Paramics) working together. The evaluation strategy set out by UCSB, was to examine:

- the relative merits of Models A and B, and the response to generalization parameters, in terms of traffic throughput and computational performance of Paramics;
- adequacy of syntax in handling ramps and other special road configurations;
- stability and robustness of the software under large data sets containing complex geometry.

In general the tests employed worst-case scenarios. A winding mountain road was selected to test geometrical semantics, while freeway sections and interchanges were used to test syntax translation.

The evaluation concluded that S2P generated "huge savings of resources" compared with manual data creation, performed without system errors and crashes, and computation time was never an issue. Critical comment focused on problems with short links, and the lack of alignment of start/stop lines. These are important because they necessitate manual editing.

The following findings of the evaluation are generally valid:

- S2P may not produce perfectly usable Paramics files. It is a "first cut" that reduces, but does not eliminate the work required to prepare Paramics files. Much depends on the quality regime of the input shape files. If there is not enough detail, e.g. if the geometry is poor or the DBF file does not specify the speed limit or number of lanes on each stretch of road, then much editing remains to be done, either of the shape or DBF file prior to processing by S2P, or of the Paramics files. In some cases there can be too much detail. It may be a challenge to find a GIS file with exactly the right amount of detail, but such files are processed without trouble (Figure 9).

- S2P is most useful when a large volume of data has to be translated. Single line city streets with simple intersections are most easily and faithfully translated. Long stretches of highway are converted well, with smooth curve sections and transitions.

In two respects the evaluation may have overemphasized the likelihood of error or created incorrect impressions:

- San Marcos Road in Santa Barbara, which was a test section for the evaluation, is a winding, mountainous, 2-lane road that climbs 800 m over a 10 km distance — an average 8% grade. Vehicle flow is rarely if ever greater than 100 per hour, and speed is constrained by curves and gradient to about 50 km/h. These details are not contained in the GDT file, so fictional scenarios were developed for testing. The road was assumed to be 4-lane with a speed limit of 80 km/h and maximum speed of 100 km/h, with hourly vehicle flow of 1000-2000. Simulation at high speed and volume is useful because it tests under exacting conditions. However, when Paramics is asked to render 4 lanes on hairpin bends with lateral separations of 15-20 meters, it understandably creates unusual results (Figures 20-24 in Appendix A). This is not a criticism of the evaluation effort: the test roads were selected by the development team at UCSB, not the evaluation team.

Figure 10. Hairpin bend on San Marcos Road. Average grade over 10 km is 8%, and separation on hairpins is 15-20 m.

- The evaluation concludes that Model 1 is better than Model 2 because it results in quicker traversal of the road. This is not necessarily a matter of superior and inferior. The Douglas-Poiker model is designed to create straight sections. They are necessarily shorter, and allow higher speeds. Neither Model 1 nor Model 2 can claim to be more realistic except as a result of empirical testing.

### *Future Directions*

The evaluation pointed out areas of potential future development. We agree that there are areas in which the total translation experience can be improved, and we recommend that Caltrans consider the following in particular:

- S2P can be supplemented with pre-processors to reduce the frequency of short and fragmented links. An associated problem is to reduce 2-line road sections to single lines. These are not trivial problems, and they must be approached carefully because they negate the trouble taken by the vendors to separate the objects; therefore the processes may potentially discard information.
- Paramics could be improved to accept shorter links and to align start/stop lines automatically.
- S2P can be improved to accept linearly referenced files, such as those commonly used at DOTs, which contain information on number of lanes, lane width, etc. New data models such as UNETRANS (*www.ncgia.cusb.edu/vital/unetrans*) will soon lead to the availability of superior quality data. It is in Caltrans' interest to take advantage of new developments in this area.
- Data now available from remote sensing and GPS, e.g. elevation, should be integrated into Paramics databases for added realism of simulation.
- A process that converts Paramics files into GIS data bases may be useful. Given the resources invested in Paramics database creation, a reverse translation tool would help GIS divisions with their task of creating and maintaining accurate, fully attributed databases.

# 5 — Implementation

## Presentation

Implementation of S2P is exceedingly simple. The product consists of just 3 files, which are easily installed and executed:

- Executable software, S2P.EXE
- Initialization file, S2P.INI
- Dynamic link library, ShapeDLL.dll

Total installed size is just over 1 megabyte. The user's manual is about 0.5 megabyte as a PDF. These files are provided on the CD that forms part of the project deliverable. The executable and documentation are also offered publicly on the project web site: *www.ncgia.ucsb.edu/ncrst/research/microsimulation*

## Applicability

S2P is designed to be generic: metric and imperial units, and right handed or left handed driving systems. We expect that it will be utilized by Caltrans and other DOTs and consultants in the U.S. and abroad.

It has already been used by Caltrans to code a highway section in and around Bishop, CA. Rough preliminary estimates of time savings were better than 50%: 8 hours using S2P (including editing in Paramics), versus 18-20 hours by entirely manual methods. This was using Version 2 of S2P, in which ramp coding had not yet been perfected, hence ramps would have had to be adjusted manually. The GIS file (from GDT) did not contain the number of lanes; this had to be added to the Paramics files. Other problematic aspects of GDT data (described in the section "Input Data Quality") applied to this test.

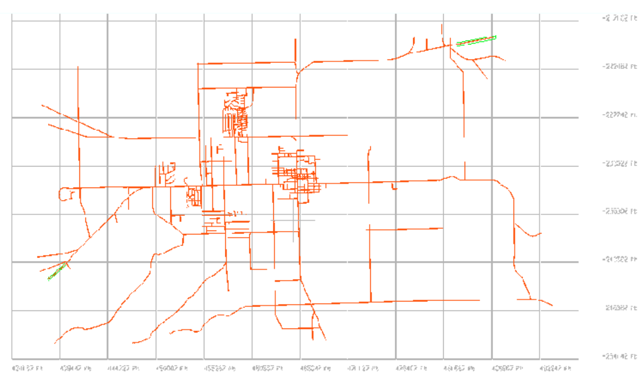


Figure 11. Bishop CA, coded by Caltrans using S2P

We anticipate that widespread use of S2P will facilitate and encourage the use of microsimulation, not just on highways (to which it is currently largely restricted because data creation is such a hurdle), but also in local downtown and residential settings, where it can be applied to evaluate the impact of new construction and other projects, emergency evacuation, offering benefits in construction design, and perhaps even saving lives. Trivially, these benefits can be measured as time savings on current projects (say 5 person-months per year throughout California, or in the region of $35,000). More realistically, because the product enables the use of microsimulation where it would otherwise be considered infeasible, the real benefit in opportunity and avoided cost is at least an order of magnitude greater.

In treating the project as a cost share with the federally funded NCRST program, UCSB has leveraged the research, enabling more development effort than would have been possible on Caltrans funds alone. We strongly recommend that Caltrans continue to work with UCSB to publicize the product, and to note areas for refinement and further development.

# Appendix A — Performance Evaluation

[Performed by Yonnel Gardes, University of California, Berkeley, using S2P Version 2.0e]

The development of S2P included an evaluation task, which was performed in order to assess the performances of the new software. Carried out at the early stages of the project, this evaluation provided useful insight to the developers, which were able to make a number of modifications to the tools before a prototype version was released.

Once the prototype conversion tool was available, the evaluation aimed at testing the performances of the software. The user point of view was the driving force in designing the evaluation process. The purpose was to assess the performances of the conversion tool, and illustrate how to best use it in real life Paramics application projects.

The general methodology that was used consisted in first processing shape files with S2P to create a Paramics network; the resulting network would then be assessed visually in Paramics Modeller; finally the created network would be used to run Paramics and compute travel performance statistics.

This process was intended to address the fundamental question of which translation method and parameters available in S2P would be optimal for future applications. In order to do so, different conversion methods and tolerance parameters were used to create a series of Paramics networks based on the same initial Shape file, and the resulting traffic performances were compared.

This chapter of the final report presents the evaluation methodology, provides some general observations on the conversion tool, describes the details of the experiment plan and presents the results that were obtained. A conclusion section discusses current limitations and ideas for future development.

## Methodology

The evaluation methodology included three major components:

- Software reliability;
- Qualitative network assessment;
- Quantitative traffic performances.

The software reliability was not tested in a systematic way. However, the evaluation team had access to a wide range of different data sets, and therefore had the opportunity to apply the conversion tool to very different cases. It was found that even with large and complex data sets, the algorithms were robust enough, and no software crashing was ever experienced. Depending on the size and details of the shape file, the resulting Paramics network may be very large and complex, but the S2P software can handle any network size. No memory breaking down was ever experienced either.

The second type of evaluation involved the qualitative assessment of the converted network. Experienced Paramics users can rapidly judge the quality of a given Paramics network and detect potential problems with the geometry, even before the simulation is run and any statistics are computed. Microsimulation in general, and Paramics in particular, are extremely sensitive to the quality of network coding. Special attention must be given to the link lengths, the curvatures, the continuity of the network and the transition areas between links. It is not necessarily enough to have links looking well aligned; the "stoplines" location and angles have to be checked to ensure a smooth transition between links. Stoplines in Paramics are points over which the vehicles have to pass at a given angle when moving from the end of a link to the next link.

The third evaluation criteria was related to traffic performances. The idea was to test the performances of the created network by sending some vehicles onto the network and computing statistics on their journey. By comparing statistics obtained with different network, it is possible to assess the sensitivity of the different conversion methods and tolerance parameters available within the software.

### *General Observations*

S2P uses Shape files to create the three fundamental network input files used by Paramics: nodes, links and centers files. The "centers" file is only needed when the network contains curved links.

There is a general trade-off in network coding between the quality of the resulting visualization and the level of details and complexity. For example, a network with few nodes and links, using only straight links and intersections modeled as a single node, will be simple and fast to code. However, such a network will not take full advantage of the graphical capabilities of Paramics. On the other extreme, a network with too much detail will likely result in too many short links, which is a source of problems for vehicles traveling through the network, as will be further discussed in this evaluation section.

A critical general observation about the conversion process is highly sensitive to the quality of the input database. If the Shape file is unreliable, for instance by lack of continuity or too many redundant nodes, the resulting Paramics network cannot cope with these deficiencies. It is therefore critical that S2P users look for the best possible GIS data sets before starting the process of the network conversion.

S2P developers and evaluators do not claim that any Paramics network created with the conversion tool is fully completed and adjusted, and ready to be used as it is. Instead, the conversion process should be regarded as a first step towards network building, providing a simple, robust and extremely fast way to produce a first cut network. Compared to other conventional methods used for network coding in Paramics (based on aerial photos or CAD files), it is a significant savings in terms of resources required to build a network, especially when dealing with large and complex network.

Independently of the conversion method and set of parameters, some general limitations will apply (at least with the current version of the tool). By construction, S2P only deals with nodes and links, and nothing else. Therefore, everything else is left to Paramics, or to the user. For instance, the stoplines location and angle will automatically be generated by Paramics (or modified by the user), but not within S2P. In some cases, particularly with very short links resulting from nodes too close too each other, the automatic process of adjusting stoplines will fail to find an appropriate solution; as a result, stoplines poorly aligned will disturb the smooth transition of vehicles. This problem of stopline adjustment is illustrated on Figures 12 and 13.

In both cases, nodes located too close to each other in the original shape file create the problem. Paramics is struggling to properly fit and adjust stopline positions. With poorly aligned stoplines, vehicles experience unrealistic delays to cross the transition section, which quickly creates a bottleneck and congestion spilling back.

The same problem may cause network discontinuity, when stoplines are so badly adjusted that no transition at all is possible. Figure 14 is an illustration of that situation; the movement from node 347 to node 346 is impossible, and all traffic that would normally travel through this section is blocked.

Another type of problem that was encountered relates to the arc links with a sweep angle close to 180 degrees. In order to be properly handled by Paramics, these half-loop sections (for instance at freeway loop ramps) have to be broken down into two or more links. S2P does not do that at the moment, which generates some unrealistic traffic behavior in Paramics, as shown on Figure 15 on the 170:169 link. The traffic flow is not completely disrupted, but the transition is not realistic at all. Adding an intermediate node half-way through the curve would take care of the problem. This issue has now been solved in the latest version of the code (see Update section at the end of this document).

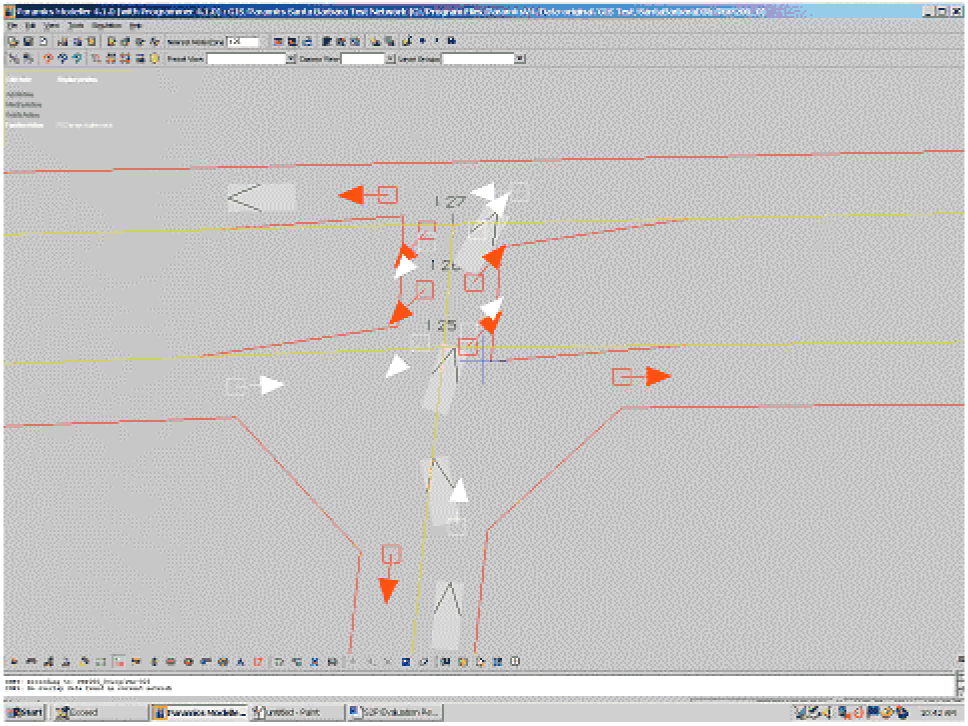


Figure 12. Poor stopline alignment due to short links (example 1)

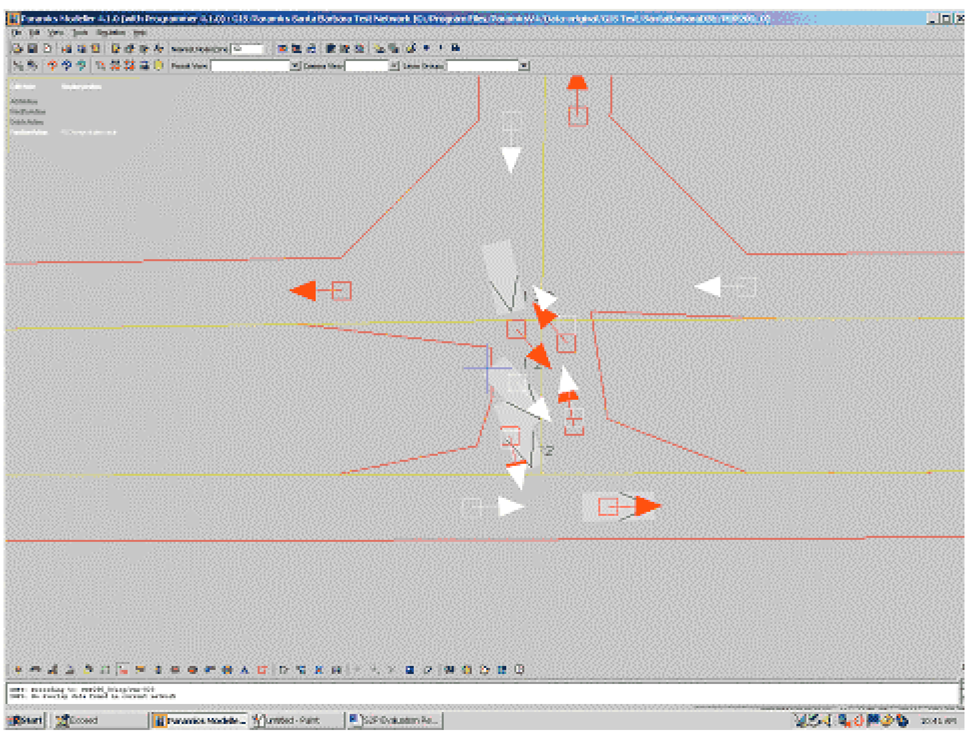


Figure 13. Poor stopline alignment due to short links (example 2)

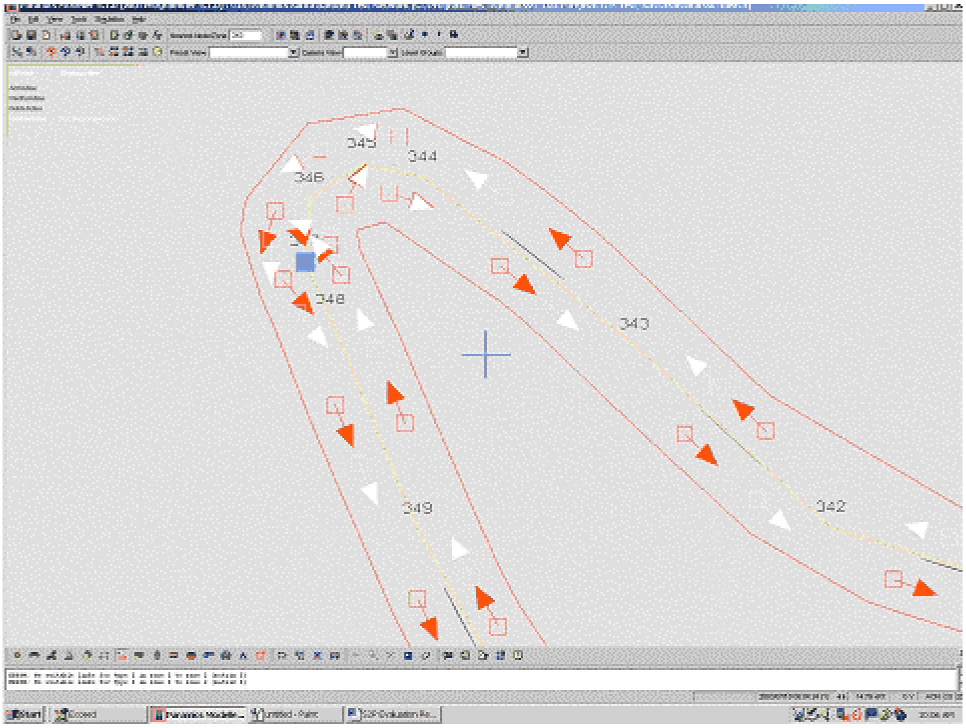



Figure 14. Poor stopline alignment due to short links, leading to network discontinuity (example 3)

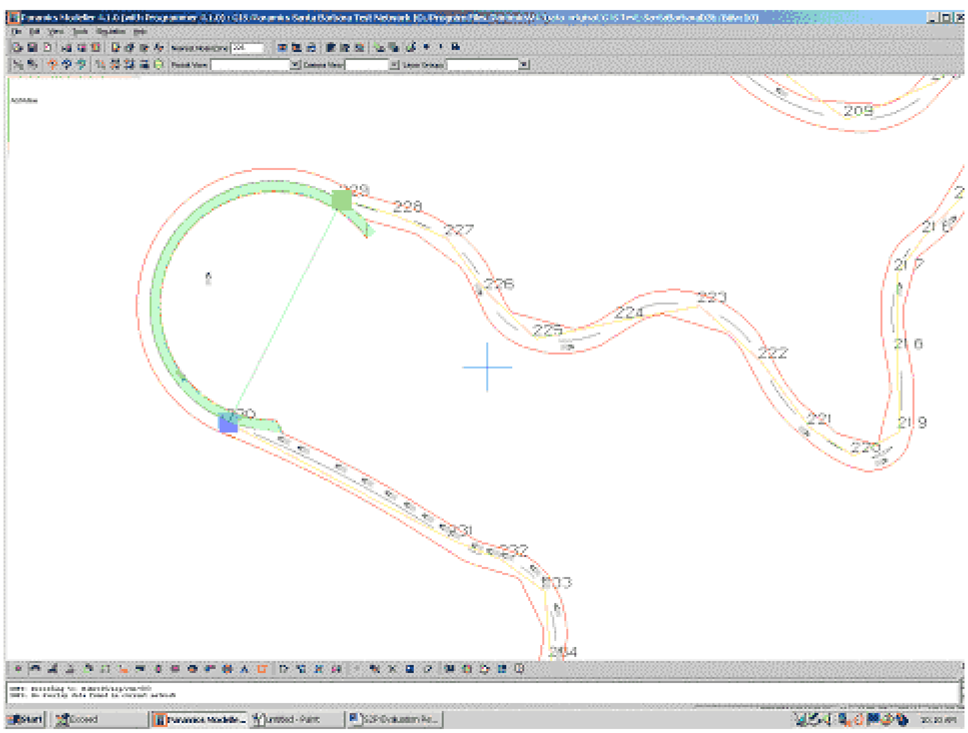



Figure 15. Half-loop Problem

## *Design of Experiment*

An experiment plan was designed in order to investigate the impact of the conversion method and tolerance parameters. The same original Shape File was used to create a series of networks in S2P using the two available translation methods (Polyline Douglas-Poiker, and BiArc) and different tolerance parameters.

The original shape file that was used is shown in Figure 16. It is an extract of a commercially available GDT (Geographic Data Technology) database for an area north of Santa Barbara. This shape file represents a winding mountain road and is therefore complex enough to provide an interesting problem in terms of conversion process.

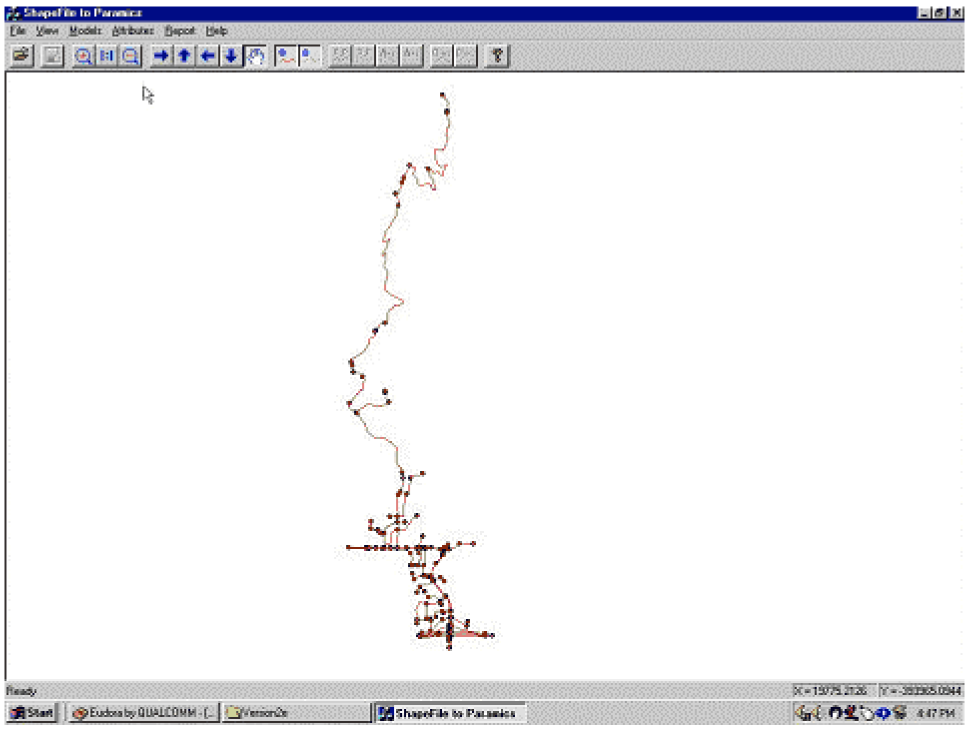



Figure 16. Original Santa Barbara shape file

In order to allow for a meaningful comparison between the conversion techniques, it was decided to use directly the network files generated by S2P. Even when obvious problems with network geometry were detected, the evaluation team would continue testing the network without manually modifying them. This guaranteed an objective evaluation of the algorithms, everything being held constant except the conversion method and tolerance parameters.

Table 1 shows the various conversion methods and parameters used in the analysis. A total of 11 scenarios were considered, six with Method 1 and five with Method 2. The default settings for the tolerance parameters correspond respectively to Scenario 1 for Method 1, and Scenario 9 for Method 2.

The speed limit was set to 50 mph, whereas the top vehicle speed was set 100 km/h (62 mph). All links and vehicles had the same characteristics. Traffic was generated from the southern end to the northern end of the network, a trip of 10 km. Two levels of demand were considered: 1000 vehicles per hour, and 2000 vehicles per hour. As all links have only one lane, these demand levels would be representative of medium and high traffic conditions.

The routing was based on an all-or-nothing assignment, ensuring that all vehicles would make the same route choice based on free-flow travel times.

Each simulation was run for one hour. Statistics to be collected at the end of the simulation included the following:

- number of vehicles actually loaded onto the network; number of vehicles prevented from starting their trip; number of vehicles that have reached the destinations by the end of the simulation.
- minimum, maximum, average and standard deviation of trip times to traverse the entire network.

## *Results and Discussion*

### Qualitative Analysis

The results are first analyzed in terms of general nodes and links characteristics.

The eleven Paramics networks created by S2P using the various methods and parameters presented in Table 1 are compared in terms of number of nodes, links and arcs.

Table 1 . Conversion methods and tolerance parameters

| **Method 1** | **Douglas Poiker tolerance (m)** | **Circular arc tolerance (m)** |
|---|---|---|
| Scenario 1 (default) | 200 | 0 |
| Scenario 2 | 50 | 0 |
| Scenario 3 | 200 | 100 |
| Scenario 4 | 100 | 100 |
| Scenario 5 | 100 | 300 |
| Scenario 6 | 50 | 300 |
| **Method 2** | **Biarc tolerance (m)** | |
| Scenario 7 | 3 | |
| Scenario 8 | 5 | |
| Scenario 9 (default) | 10 | |
| Scenario 10 | 15 | |
| Scenario 11 | 20 | |

Results are presented in Table 2.

Table 2. Comparison of general network characteristics

| **Method 1** | **Nodes** | **Arcs** | **Links** |
|---|---|---|---|
| Scenario 1 (200/0) | 180 | 0 | 198 |
| Scenario 2 (50/0) | 200 | 0 | 218 |
| Scenario 3 (200/100) | 180 | 0 | 198 |
| Scenario 4 (100/100) | 187 | 1 | 205 |
| Scenario 5 (100/300) | 182 | 4 | 200 |
| Scenario 6 (50/300) | 184 | 9 | 202 |
| **Method 2** | | | |
| Scenario 7 (3) | 714 | 514 | 732 |
| Scenario 8 (5) | 606 | 428 | 624 |
| Scenario 9 (10) | 474 | 322 | 492 |
| Scenario 10 (15) | 407 | 262 | 425 |
| Scenario 11 (20) | 363 | 216 | 381 |

The first overall observation relates to the wide range of values obtained with regard to the number of nodes and links. If the results are relatively homogeneous for the first six scenarios (with Method 1), they vary widely with the results obtained with Method 2.

Method 1 always produces networks with fewer nodes and links, meaning that the links are longer on average than those obtained with Method 2.

Within Method 1, it is confirmed that a circular tolerance of 0 results in a network with no arc links. For a given Douglas Poiker tolerance, the number of arc links increases with the circular arc tolerance. Even with very high circular tolerance relatively to the DP tolerance, the number of arc links remains low (not more than 9 arc links in our scenarios).

The higher Douglas-Poiker tolerance parameters lead to fewer links. However, it was found that the parameters settings in Method 1 did not significantly modify the overall characteristics of the network. This is demonstrated by the fact that the total number of links remains fairly stable among the first six scenarios.

On the contrary, within Method 2, the impact of varying the tolerance parameter is very strong. The resulting total number of links ranges from 381 to 732. This observation is depicted graphically on Figure 17.

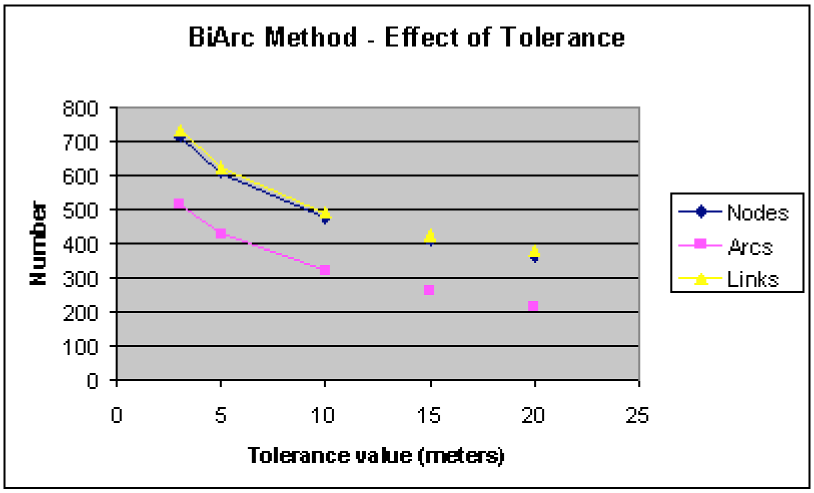



Figure 17. Method 2 (BiArc) – Effect of tolerance parameter on number of nodes and links

The number of arc links is proportionally much higher than with Method 1; for instance in scenario 7, 70% of the links are curved links.

The network with fewest links is produced with the highest tolerance parameters. But even in this case, the network has a lot more links that any networks created with Method 1. The contrast between both Methods is sharp, even when comparing the networks qualitatively. Method 2 produces networks that are better looking (more nodes, more arcs, more details) but also more prone to generate problems from a traffic flowing perspective.

## Quantitative Analysis

Based on the experiment design described under section 4, the Paramics simulation was run with each of the eleven networks created with S2P.

Traffic statistics were collected over the one-hour simulation period in Paramics Modeler.

The results are presented in three steps:

- Network loading/unloading
- Trip time performance
- Effect of demand level

### Network loading/unloading

Table 3 shows the results of the vehicle loading and unloading analysis. The idea was to compare how many vehicles were starting and finishing their trip within the hour of simulation. The demand level was constant at 1000 vehicles per hour. Because Paramics uses a stochastic vehicle release process, one would not expect exactly 1000 vehicles to be released onto the network after one hour, but a number fairly close to 1000. This is what was observed as can be checked in the "Vehicles In" column.

Table 3. Network loading and unloading

| Scenario | Demand (veh) | Vehicles In | Un-released | Vehicles Out | % out/in | Note |
|---|---|---|---|---|---|---|
| 1 | 1000 | 1068 | 0 | 954 | 89.3% | |
| 2 | 1000 | 983 | 0 | 871 | 88.6% | |
| 3 | 1000 | 1022 | 0 | 914 | 89.4% | |
| 4 | 1000 | 1015 | 0 | 911 | 89.8% | |
| 5 | 1000 | 1005 | 0 | 48 | 4.8% | See Fig 7 |
| 6 | 1000 | 980 | 0 | 48 | 4.9% | See Fig 7 |
| | | | | | | |
| 7 | 1000 | DISCONTINUITY (no route found at link 347:346) | | | | See Fig 3 |
| 8 | 1000 | DISCONTINUITY (no route found at link 298:297) | | | | See Fig 3 |
| 9 | 1000 | 955 | 0 | 592 | 62.0% | See Fig 4 |
| 10 | 1000 | 1003 | 0 | 806 | 80.4% | See Fig 8 |
| 11 | 1000 | 1000 | 0 | 602 | 60.2% | See Fig 4 |

The "unreleased" vehicles column would keep track of any vehicles prevented from starting their trip because of congestion spilling back to the origin zone. At the demand level of 1000 veh/h, this phenomenon never occurred.

The "Vehicles out" column displays the number of vehicles that have actually reached their destination within the hour of simulation. It is not expected to be equal to the "Vehicles In" value because some vehicles are still traveling when the simulation is stopped and the statistics are computed. But, accounting for this fact, the Out/In ratio would be expected to be fairly high, because the trip length is relatively short (10km) in comparison to the simulation period (one hour).

When looking at Table 3, three kinds of situations emerge. Scenarios 1 to 4 lead to a very high Out/In ratio, a good sign that the networks are smooth flowing and trouble-free. Scenarios 5 to 8 produce networks with serious flaws, preventing all or most of the vehicles to reach their destination. Finally, networks produced in Scenarios 9 to 11 are intermediate, they do not operate as well as the first four but do not break down either. Among the last three scenarios, Scenario 10 seems to perform better with a Out/In ratio of 80%.

Based on the qualitative analysis previously presented, it was expected that Method 1 would be producing networks with better traffic performances, because of longer links and less transition zones. One problem that does occur with Method 1 and high circular arc tolerance (300 m) is shown on Figure 18. The conversion method failed to produce a reasonable circular link between nodes 59 and 60, and as a result vehicles are almost totally blocked at this section. This explains why so few vehicles managed to reach their destination in Scenarios 5 and 6.

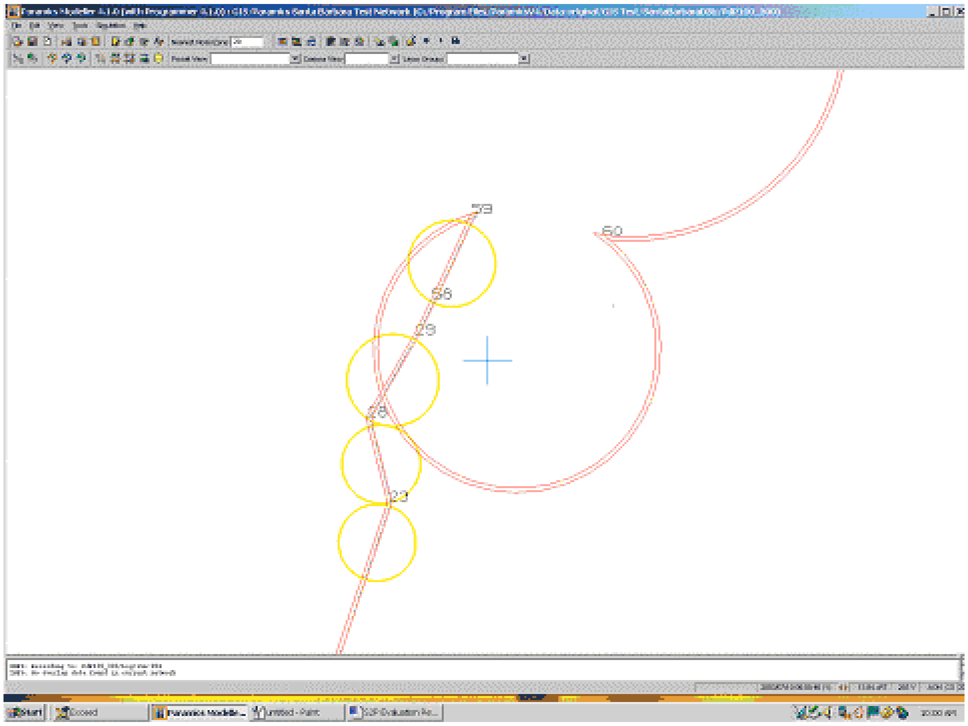


Figure 18. Network transition problem (Scenario 5-6)

Method 2, producing networks with more links and a very high proportion of arc links, is much more likely to result in traffic flowing problems. This is especially true with low tolerance values, as could be expected based on Table 2.

The most critical problem was observed in Scenarios 7 and 8. The traffic was totally disrupted as no route was found for the vehicles to reach their destination. This was due to the stopline break down previously described under section 3 and illustrated in Figure 14. The constraints imposed by the short links did not allow for a suitable stopline adjustment within Paramics, which created a network discontinuity.

Scenarios 9 and 11 perform better in the sense that 60% of the traffic did make it to their destination. However, this value is low compared to what was expected given the problem characteristics. A closer analysis revealed that these two networks experienced the "half-loop" problem previously described (see section 3) and graphically captured on Figure 15.

Scenario 10 produces a reasonable rate of vehicles reaching their destination. The "half-loop" problem was avoided in this case. It should be noted, however, that some unexpected delays were experienced at another curved section (see Figure 19) due once again to a stopline adjustment issue.

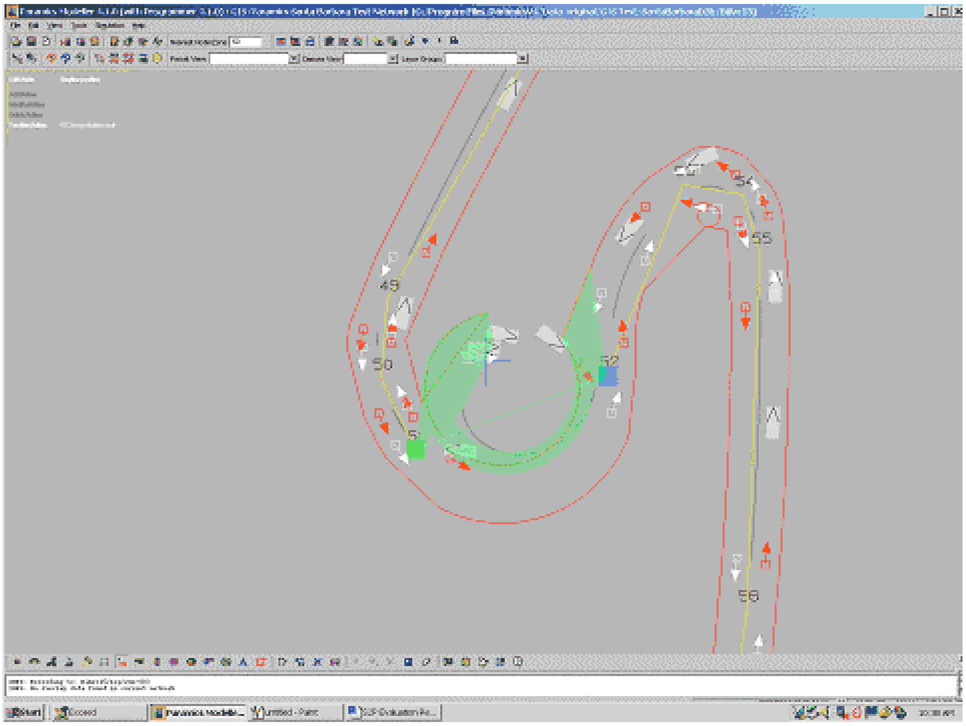


Figure 19. Network transition problem (Scenario 10)

### Trip time performance

Even though a lot can be learned just by looking at the general network characteristics and the loading/unloading patterns, further insights can be obtained through a detailed analysis of the trip time performances.

As indicated earlier (section 4), trip time results were computed for all vehicles released onto the network. The statistics are presented in Table 4.

Table 4. Trip Time Statistics

| Scenario | Minimum Trip Time (H:MM:SS) | Maximum Trip Time (H:MM:SS) | Average Trip Time (H:MM:SS) | St Dev Trip Time (H:MM:SS) | Average Speed (km/h) | Note |
|---|---|---|---|---|---|---|
| 1 | 0:05:35 | 0:06:37 | 0:06:04 | 0:00:11 | 100.7 | |
| 2 | 0:05:55 | 0:07:02 | 0:06:34 | 0:00:10 | 93.1 | |
| 3 | 0:05:34 | 0:06:34 | 0:06:02 | 0:00:10 | 101.1 | |
| 4 | 0:05:56 | 0:07:00 | 0:06:29 | 0:00:11 | 94.3 | |
| 5 | 0:07:31 | 0:57:32 | 0:32:34 | 0:14:52 | 18.8 | See Fig 3 |
| 6 | 0:06:45 | 0:56:16 | 0:31:32 | 0:14:40 | 19.4 | See Fig 3 |
| | | | | | | |
| 7 | NETWORK DISCONTINUITY | | | | | See Fig 7 |
| 8 | NETWORK DISCONTINUITY | | | | | See Fig 7 |
| 9 | 0:10:09 | 0:21:32 | 0:15:42 | 0:02:42 | 38.9 | See Fig 4 |
| 10 | 0:10:19 | 0:12:05 | 0:11:04 | 0:00:24 | 55.2 | See Fig 8 |
| 11 | 0:09:39 | 0:22:24 | 0:16:28 | 0:03:20 | 37.1 | See Fig 4 |

These results obviously confirm what was previously hinted. The networks produced with Method 1 perform significantly better in terms of traffic flowing. This is mainly due to the fact that they have fewer nodes, fewer links, and less transition sections likely to generate traffic slow-downs or disruptions.

Fastest traversal times are obtained in Scenarios 1 and 3, both in terms of minimum trip times and average trip times. But the first four scenarios are fairly close in terms of traffic performance, with vehicles being able to travel at their top speed of 100 km/h (ie 62 mph). The travel times are relatively homogenous, with standard deviation of 10 seconds.

Method 2 could not produce the same kind of performances. The minimum traversal time was close to 10 minutes, which translates to a speed of 60 km/h or 37 mph. When compared to the initial free flow speed of 50 mph set in the Paramics network, it clearly shows that the geometry generated some friction preventing the vehicles from traveling at their desired speed. With the effect of traffic demand, delays increased slightly for an average traversal time of 12 minutes and 5 seconds in Scenario 10. In addition to the general characteristics of the Method 2 networks, statistics gathered for Scenarios 9 and 11 are strongly impacted by the “half-loop” problem previously described, which generated a bottleneck and high delays.

### Effect of demand level

The same eleven networks were tested with a higher demand level, 2000 vehicles per hour (instead of 1000 veh/h in all previous investigations). Everything else remained unchanged. This demand level of 2000 vehicles is very high when compared to the capacity of the network, which is built with one-lane links only. It was expected that the traffic performances would be further reduced because of the additional effects of traffic congestion in addition to the recurrent problems with the geometry previously described.

The results were analyzed in the same way than with the initial demand level. Similar tables were produced and are shown as Tables 5 and 6.

Table 5. High Demand - Network loading and unloading

| Scenario | Demand (veh) | Vehicles In | Un-released | Vehicles Out | % out/in | Note |
|---|---|---|---|---|---|---|
| 1 | 2000 | 1998 | 0 | 1764 | 88.3% | |
| 2 | 2000 | 2028 | 0 | 1767 | 87.1% | |
| 3 | 2000 | 1911 | 8 | 1704 | 89.2% | |
| 4 | 2000 | 2056 | 0 | 1808 | 87.9% | |
| 5 | 2000 | 1121 | 864 | 48 | 4.3% | See Fig 3 |
| 6 | 2000 | 1133 | 894 | 49 | 4.3% | See Fig 3 |
| | | | | | | |
| 7 | 2000 | DISCONTINUITY (no route found at link 347:346) | | | | See Fig 7 |
| 8 | 2000 | DISCONTINUITY (no route found at link 298:297) | | | | See Fig 7 |
| 9 | 2000 | 1682 | 274 | 619 | 36.8% | See Fig 4 |
| 10 | 2000 | 1402 | 593 | 900 | 64.2% | See Fig 8 |
| 11 | 2000 | 1356 | 636 | 599 | 44.2% | See Fig 4 |

Most of the remarks previously made still apply to the high level scenarios. Table 5 indicates that the three same groups of scenarios can be identified. Scenarios 1 to 4 perform very well, with a rate of arrivals close to 90%. This shows that the capacity resulting from this network was able to accommodate this kind of demand level, at least for the first hour of simulation. Congestion effects would have likely been felt more intensively had the simulation been performed over a longer period. Scenarios 5 to 8 failed for the same reasons previously exposed. And scenarios 9 to 11 show mixed results, with a high fraction of the vehicles being unreleased due to congestion spilling back to the origin zone. For these last three scenarios, the increased traffic demand clearly exacerbated the geometry problems previously highlighted and led to a sharp decline of arrival rate (out/in) compared to those shown on Table 3.

Table 6. High Demand - Trip Time Statistics

| Scenario | Minimum Trip Time (H:MM:SS) | Maximum Trip Time (H:MM:SS) | Average Trip Time (H:MM:SS) | St Dev Trip Time (H:MM:SS) | Average Speed (km/h) | Note |
|---|---|---|---|---|---|---|
| 1 | 0:05:48 | 0:21:13 | 0:07:23 | 0:01:44 | 82.8 | |
| 2 | 0:06:08 | 0:20:31 | 0:07:34 | 0:01:20 | 80.7 | |
| 3 | 0:05:21 | 0:17:21 | 0:06:36 | 0:00:51 | 92.4 | |
| 4 | 0:06:00 | 0:22:28 | 0:08:04 | 0:01:47 | 75.7 | |
| 5 | 0:07:12 | 0:57:29 | 0:32:24 | 0:15:01 | 18.8 | See Fig 3 |
| 6 | 0:06:10 | 0:57:55 | 0:32:14 | 0:15:28 | 18.9 | See Fig 3 |
| | | | | | | |
| 7 | NETWORK DISCONTINUITY | | | | | See Fig 7 |
| 8 | NETWORK DISCONTINUITY | | | | | See Fig 7 |
| 9 | 0:10:25 | 0:41:27 | 0:25:33 | 0:08:24 | 23.9 | See Fig 4 |
| 10 | 0:10:18 | 0:38:11 | 0:21:30 | 0:06:05 | 28.4 | See Fig 8 |
| 11 | 0:09:13 | 0:52:14 | 0:26:03 | 0:09:28 | 23.5 | See Fig 4 |

Table 6 shows the trip time statistics for the high demand level case. As would be expected, the minimum trip times are similar to those displayed on Table 4, reflecting the fact that the first vehicles traveling through an empty network experienced the same network conditions. However, major differences appear in terms of maximum and average trip times when compared to Table 4. This is due to the fact that, as the simulation progresses, there is more and more traffic loaded onto the network, producing more and more friction between these vehicles. Any problems with the network geometry will result in more delays and more vehicles being affected, which all contribute to a general degradation of the recorded traffic performances.

## *Summary of Findings*

S2P offers a new and very powerful approach towards network coding in Paramics. Provided that a good quality GIS database is available, the conversion tool allows for a first-cut network to be constructed extremely quickly. Compared to the more traditional methods of network building relying on manually coded nodes and links based on a background overlay (aerial photo or CAD), the GIS conversion software generates huge savings of resources. This is true for networks of any size, but is obviously more relevant for large area applications.

The experience of testing S2P with very different input shape files revealed that the software is robust enough to handle networks of any size and complexity. The computing time in S2P has never been an issue either.

It is recognized, however, that any network created by S2P can only be considered as a first step in the network building process. S2P only generates nodes, links and centers files, and all other input files have to be either automatically generated by Paramics Modeller, or coded by the user. The networks created by S2P, independently of the conversion method used, all require some further modifications and editing from the user before any meaningful simulation can be carried out.

One fundamental problem that arises, especially with the Method 2 conversion technique (BiArc), is that of too many nodes and links in the created network. With nodes being too close to each other, links are much shorter and more transitions from link to link are necessary. These transitions between links, especially between curved links, are the source of potential geometry problems resulting in traffic slowing down or being stopped. Reducing the number of nodes and links results in networks that are less prone to traffic disruption. This is the reason why Method 1 was found to perform better in terms of traffic performances. The visual appearance of a Method 2 network may be better, with more details and curved links, but the traffic performance suffers from the high number of transition areas, high proportion of curved links and shorter links.

Based on the results of this series of test, Method 1 should be recommended in future applications of S2P; the default tolerance parameters of 200 meters (and a circular tolerance of 0) provides a excellent starting point, with a network that is well balanced in terms of number of nodes and links. No circular links will be restituted (if the circular tolerance is 0) or very few (with a higher circular tolerance), but they can easily be added if the user decides to do so. The graphical aspect of the network can be gradually improved by modifying or adding links if needed. This approach seems more efficient than the opposite one, which would be starting with a very complex and detailed network typical of Method 2, and working backwards to simplify and adjust it manually.

### *Recommendations for Future Developments*

The testing reported here was carried out using the initial version of S2P. A number of limitations that have been identified during the course of this evaluation could potentially be addressed in future versions of the tool, if the development effort is continued. The tool as tested here already provides an excellent supporting module for any Paramics users facing the task of building a new network.

One of the main problems identified in this evaluation is related to nodes being too close to each other. In many cases, just merging this nodes together would solve the problem. It would be extremely desirable to develop a feature that would look for nodes that are redundant and merge them together. The user could specify a minimum link length; if the conversion cannot fit a link below that threshold, that would mean that two nodes are too closely spaced, and must be merged. This feature would eliminate many of the transition problems resulting from poor stopline adjustment.

Another option to consider would be for S2P to also generate stopline adjustments directly. After identifying a potential problem with stopline location or angle, S2P would automatically generate the appropriate correction in the stopline file, as opposed to letting Paramics doing it automatically, not always successfully.

A minor fix would be to solve the "half-loop" issue by making sure that any arc links with a sweep angle of more than 140 degrees would be broken down into two links. This should not be too difficult to implement, and would be very valuable. It has now been solved in the latest version of the code (see Update section at the end of this document).

Another type of issue is raised by freeway facilities. Ramp merging are not well treated at this stage of development of S2P. The ramp merging is coded as an intersection, as opposed to an on-ramp with an accelerating link. Paramics has a very special way of dealing with freeway ramp merging areas, and S2P should be able to mimic this. An additional issue is the fact that all non-separated links, including freeway ramps, are coded as two-way links.

The status of importing pertinent features encountered in the dataset when available (such as number of lanes, posted speed limit, one-way links, merging lanes…) is unclear at this stage, as the evaluators did not have access to a database containing any relevant information. This feature would be very desirable to further support the coding process and take full benefits of all pertinent data contained in the GIS files.

## Update Section

[Based on Version 3.0c]

A new version of the code became available at the end of the evaluation phase. Some of the quantitative observations and qualitative results had to be updated in light of the changes made.

Figures 20 through 24 and Tables 7 through 9 reflect some of the results obtained when evaluating the updated version of S2P.

Table 7. Comparison of general network characteristics (updated version)

| Method 1 | Nodes | Arcs | Links |
|---|---|---|---|
| Scenario 8n (5) | 610 | 419 | 628 (was 624) |
| Scenario 9n (10) | 491 | 329 | 509 (was 492) |
| Scenario 11n (20) | 373 | 229 | 391 (was 381) |

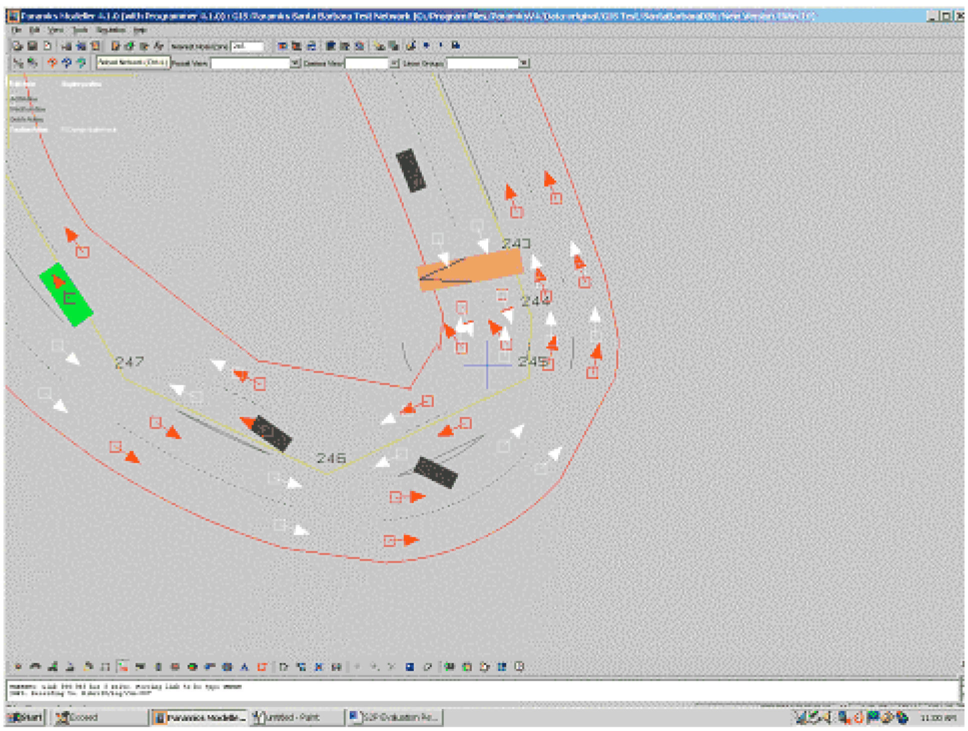



Figure 20. Stopline adjustment problem with BiARc 10 (Scenario 9n)

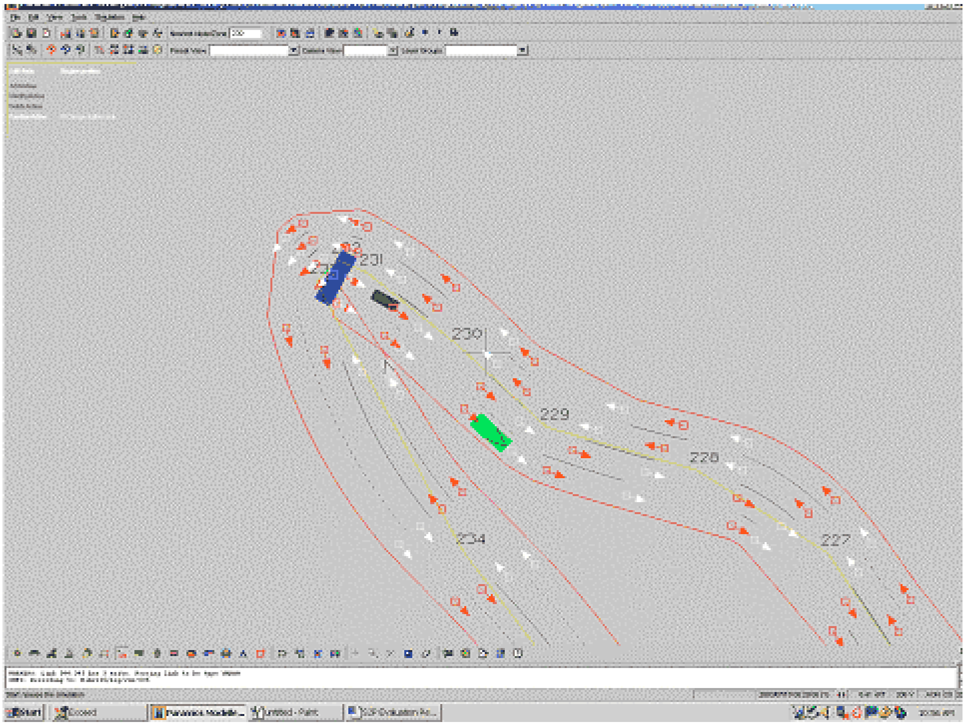



Figure 21. Stopline adjustment problem with BiARc 10 (Scenario 9n)

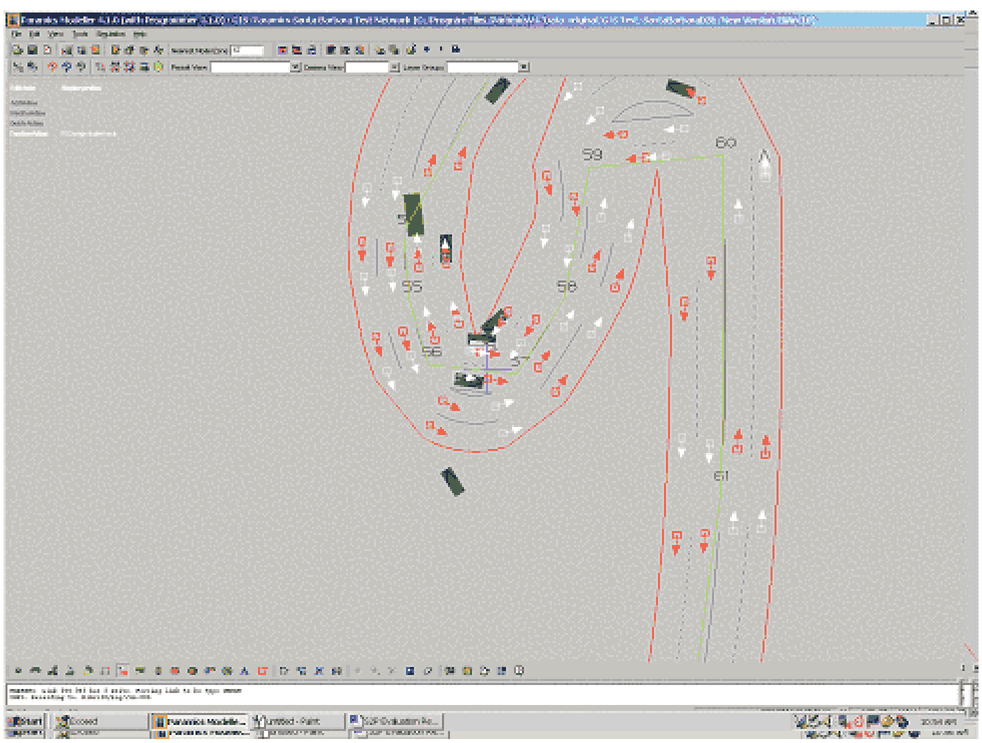



Figure 22. Stopline adjustment problem with BiARc 10 (Scenario 9n)

The new code was primarily tested with the Method 2 conversion because of the earlier findings showing that this method was much more prone to generate trouble in the flow of vehicles. Overall, it was found that the updated version of S2P did lead to better results in terms of traffic throughput performances. This is best illustrated by looking at the results presented in Tables 8 and 9, in comparison with those obtained with the older version of the code (Tables 3 and 4).

Table 8. Network loading and unloading (updated version)

| Scenario | Demand (veh) | Vehicles In | Un-released | Vehicles Out | % out/in | Note (see Table3) |
|---|---|---|---|---|---|---|
| 8n | 1000 | 945 | 0 | 761 | 80.5% | was 0 |
| 9n | 1000 | 1012 | 0 | 825 | 81.5% | was 62% |
| 11n | 1000 | 999 | 0 | 836 | 83.7% | was 60.2% |

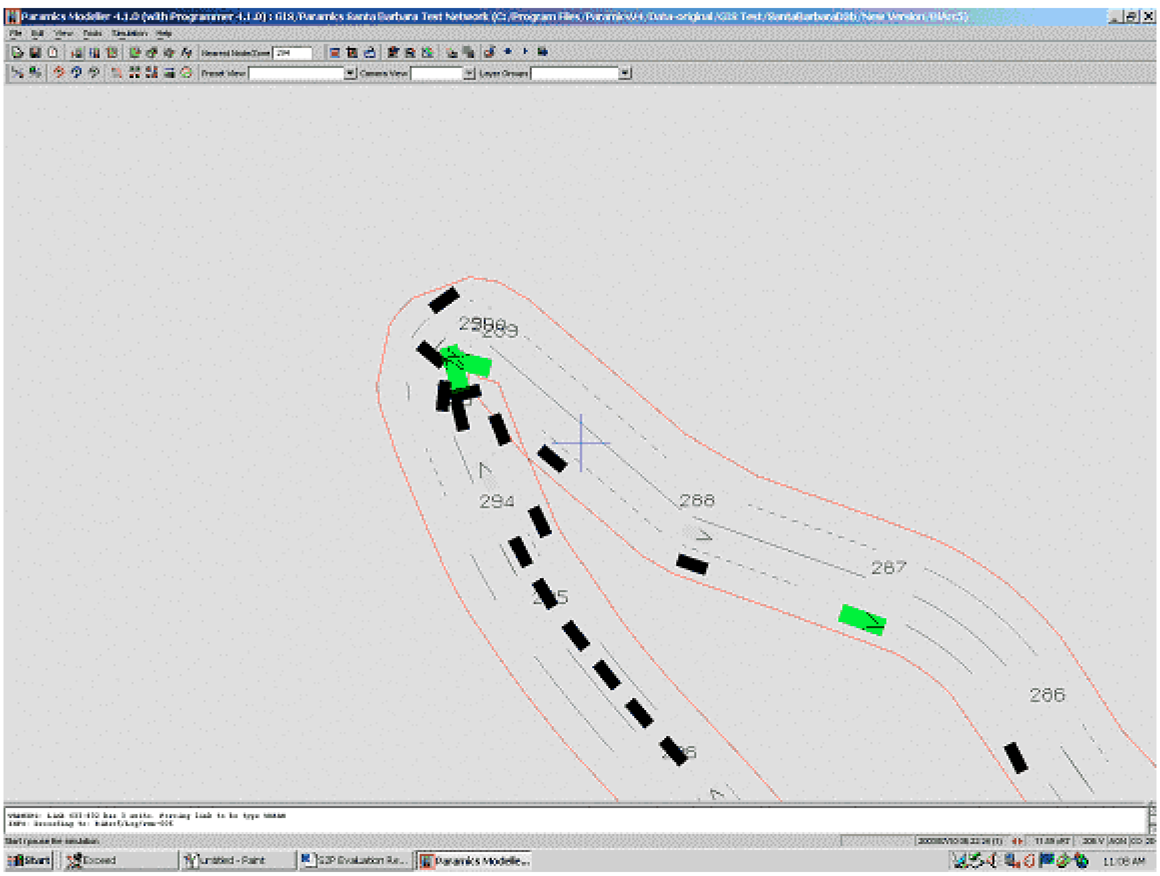



Figure 23. Stopline adjustment problem with BiARc 5 (Scenario 8n)

Table 9. Trip Time Statistics (updated version)

| Scenario | Minimum Trip Time (H:MM:SS) | Maximum Trip Time (H:MM:SS) | Average Trip Time (H:MM:SS) | St Dev Trip Time (H:MM:SS) | Average Speed (km/h) | Note (see Table 4) |
|---|---|---|---|---|---|---|
| 8n | 0:11:10 | 0:13:33 | 0:12:18 | 0:00:26 | 49.6 | was 0 |
| 9n | 0:10:15 | 0:12:55 | 0:11:26 | 0:00:29 | 53.5 | was 38.9 |
| 11n | 0:09:04 | 0:10:57 | 0:09:59 | 0:00:19 | 61.2 | was 37.1 |

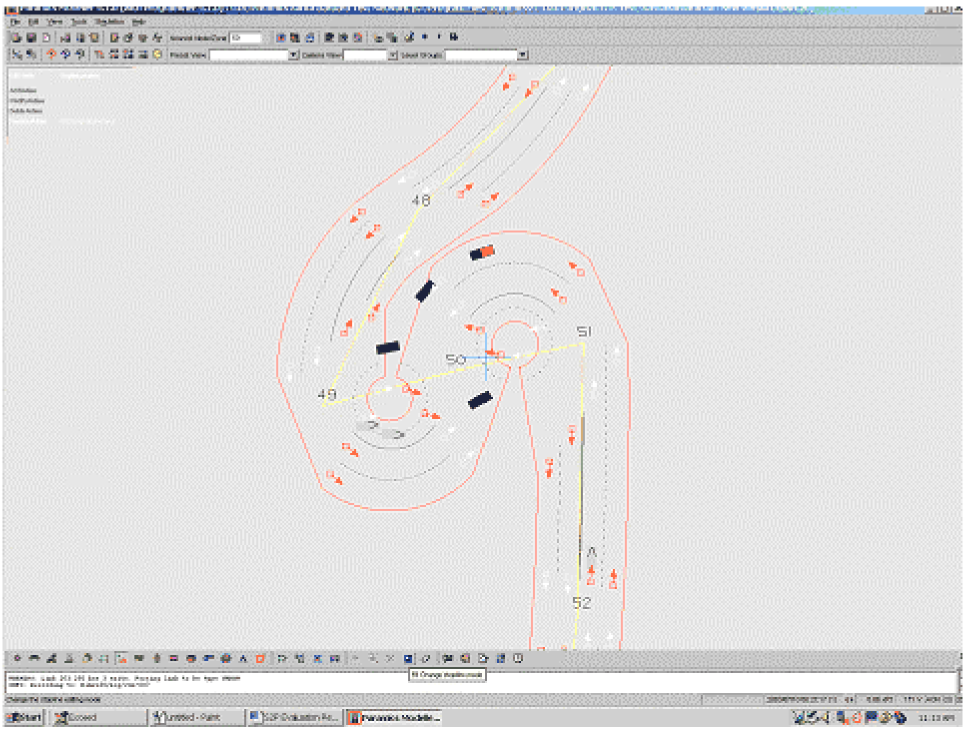



Figure 24. Stopline adjustment problem with BiARc 20 (Scenario 11n)

The half-loop problem previously identified with Scenarios 9 and 11 no longer exist with the updated code. Also, the major network discontinuity observed in Scenario 8 with the older code does not exist anymore, and the vehicles are able to to find a suitable route to reach their destination.

The updated version of the code obviously generates better results; however, most of the comments previously made still apply. In particular, the critical and recurrent issue of the stopline adjustment remains present, as illustrated in Figures 20 through 24. Once again, nodes being too close to each other create this problem; it results in traffic slowing down at transition between links, which produces a drop of vehicle speed and network capacity. Even with the improvements made in the code, Method 2 never generates trip times below 9 minutes, whereas with Method 1, more realistic minimum trip times in the order of 6 minutes could be achieved.

# References


Douglas DH, Poiker TK 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature *Canadian Cartographer* **10** 112-22

GDT 2002. Dynamap/Transportation User Manual. Version 4.3 Geographic Data Technology Inc.

Noronha VT 1999 "Towards ITS map database interoperability — database error and rectification" *GeoInformatica*, Special Issue on GIS-T and ITS, **4** (2) 201

Noronha V, Goodchild MF 2000 "Map accuracy and location expression in transportation — reality and prospects" *Transportation Research C*, Special Issue on GIS-T, **8** 53-69

Meek DS, Walton DJ 1995 "Approximating smooth planar curves by arc splines" *Journal of Computational and Applied Mathematics* **59** 221-231

Parkinson DB, Moreton DN 1991 "Optimal bi-arc curve fitting" *Computer Aided Design* **23** 411-419

Rosin PL, West GAW 1980 "Segmentation of edges into lines and arcs" *Image and Vision Computing* 109-114

VITAL 2000 "The LRMS Linear Referencing Profile — Technical Evaluation." United States Department of Transportation, FHWA Contract DTFH61-91-Y-30066

Yang X 2002 "Efficient circular arc interpolation based on active tolerance control" *Computer-Aided Design* **34** 1037-1046

Vehicle Intelligence and Transportation Analysis Laboratory
National Center for Geographic Information and Analysis
University of California, Santa Barbara CA 93106-4060

URL www.ncgia.ucsb.edu/vital
Phone +1.805.893.8992
E-mail vital@ncgia.ucsb.edu