THE DESIGN AND DEVELOPMENT OF AN ALGORITHM FOR AUTOMATICALLY
CREATING TRANSIT NETWORKS IN TRANSCAD TO MEASURE REGIONAL
ACCESSIBILITY

by

Rishi Lukka, P.Eng., B.A.Sc., Honours Systems Design Engineering & B.A. Liberal Arts,
University of Waterloo, 2007

A Major Research Paper (MRP)

presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Engineering

in the Program of

Civil Engineering

Toronto, Ontario, Canada, 2016

© Rishi Lukka, 2016

<u>Author's Declaration for Electronic Submission of a MRP</u>

I hereby declare that I am the sole author of this MRP. This is a true copy of the MRP, including any required final revisions.

I authorize Ryerson University to lend this MRP to other institutions or individuals for the purpose of scholarly research

I further authorize Ryerson University to reproduce this MRP by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my MRP may be made electronically available to the public.

Abstract

The design and development of an algorithm for automatically creating transit networks in TransCAD to measure regional accessibility. Master of Engineering, 2016, Rishi Lukka, Civil Engineering, Ryerson University

The Metrolinx Economic Analysis team commissioned Arup Canada to create a tool to measure regional transit accessibility called WithinReach. It allows transportation professionals to quickly evaluate and compare existing services versus potential projects, using availability metrics. The calculation requires the summarization of all transit route operations across a geographic area into performance standards for one service day.

In the Greater Toronto-Hamilton Area (GTHA), detailed timetables are available in the General Transit Feed Specification (GTFS) format, however manually preparing this data for WithinReach is labour-intensive. Due to the frequency of changes to vehicle and route schedules, and the number of regional agencies, maintaining an up-to-date summary of operations is cumbersome. This report covers the design, development, testing, procedure, and validation of a Transit Network Builder (TNB) to automate this effort. The TNB abstracts GTFS schedule data into performance benchmarks for use with the WithinReach Tool and the TransCAD transportation planning environment.

Acknowledgements

Table of Contents

List of Tables

List of Figures

1   Introduction

Metrolinx was created in 2006 as an agency of the Government of Ontario to "improve the coordination and integration of all modes of transportation in the Greater Toronto and Hamilton Area (GTHA)," *(1)*. Initially, the governing board of the agency included both private-sector technical experts and local political representation, which provided the cross-party support required to create a new regional transportation vision called, "The Big Move," *(2)*. The Toronto area suffers from some of the worst congestion in North America, where commuters spend on average 66 minutes getting to and from work every day, hampering economic growth to the tune of six billion dollars per year in lost productivity *(3)*. Therefore, the cornerstone of the plan was to improve mobility efficiency in the GTHA by building over 1200 kilometres of rapid transit within a 25-year timeline *(2)*.

The plan and Metrolinx governance model were seen as an improvement on decades of regional transportation planning failures due the ambitious geographical coverage (something for everyone), the backing of municipal and provincial decision makers, and the insertion of independent private sector experts to the decision making process *(4)*. In 2009, Metrolinx's role as politically independent transportation planning advisors was cemented when the province restructured the accountability structure of the agency. Ontario's Bill 163 removed electable political representatives from the Metrolinx board *(5)* intending for transportation decision making to be separated from the shorter term thinking of election cycles. While the agency was not given any overarching authority, it has continued working closely with the province, planning departments in each municipality, and transit operators to improve regional connectivity.

Beyond the overall "Big Move" plan, ongoing projects cover policy, capital, and operating initiatives including Regional Express Rail, Investment Strategy, Fare Integration, Wayfinding, Regional Relief, and Project Evaluation to name a few *(6)*.

The "Big Move" plan identified the overall vision for regional connectivity and specific corridors for investment, it did not include sufficient details that decision makers could use early on to evaluate project alternatives. To ensure that individual projects were consistent with the overall objectives of the "Big Move" and to provide return on investment guidance for a range of potential rapid transit implementations, the Metrolinx Economic Analysis team, following the lead of Transport for London *(7)* began to publish Business Case Analyses (BCAs) *(8)*. These cover *"transportation user benefits compared to the financial impact; good value for tax-payer dollars; environmental, economic and social benefits of the various alternatives; the impacts that a project has on communities; and alignment with the current policy objectives, (8)."* While the business case approach has been praised for its use of the Multiple Account Evaluation (MAE) technique to quantify project costs versus anticipated benefits *(9)*, the analysis is non-trivial, taking months to complete for a single transportation corridor *(10)*. It provides the most value when evaluating capital projects but it omits any analysis on the impact of service frequencies and the quality of on-the-ground operations. Given this limitation, the Metrolinx Economic Analysis team, led by Daniel Fisher and Michael Sutherland, commissioned the development of an accessibility tool, called WithinReach, to evaluate transit service availability across the region.

2   Conceptualization of WithinReach and the Transit Network Builder

The WithinReach project was conceived by members of the Metrolinx Economic Analysis team, who had used accessibility metrics when previously working for Transport for London (TfL) *(11)*. Based on their experiences there and local priorities, three application areas were envisioned for the tool *(12)*. The first was a geographical representation of the difference in service frequencies to compare regional transit availability at-a-glance and identify "transit deserts" (see Figure 1).
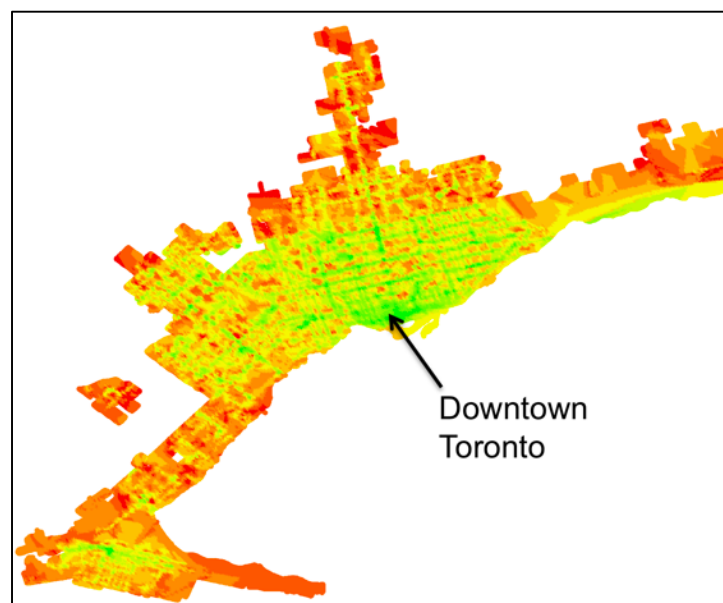
Figure 1 – Transit Service Availability Sample (13)

The second application was to use the service availability data alongside demographic, socio-economic, zoning, and density data in Geographic Information System (GIS) to further identify areas for further transit investment prioritization. The last piece was to allow transportation planners to evaluate how transit accessibility scores would change (and in theory improve) if any combination of transit improvements were put into operation.

In WithinReach, two types of GTHA accessibility scores are supported and the region is broken up into a grid to simplify the calculation and decouple it from nodes on the transportation network, which could change over time. The first metric, known as Accessibility to Transit (ATT), is based on the Equivalent Doorstop Frequency (EDF) calculation used at TfL *(11)*. It evaluates the total access time form every grid-point to every available transit service within a catchment area. For grid-points with multiple transit services available, the combination of all services provides the score. Once this is evaluated for all points, the user is able to choose whether they'd like the information displayed on a linear or logarithmic colour scale. *(14)*

The second metric, known as Accessibility to Destinations (ATD), focuses on the distance travelled by transit within a fixed time budget from a set of origins to a set of destinations (O-D). It extends the ATT calculation by adding the shortest path travel times for each O-D pair to the access times, and allows analysts to see the impact of the speed of service. More specifically, it showcases the modal advantage of being located closer to higher order rapid-transit as opposed to higher frequency, but slower surface routes *(14)*.

The ATT and ATD accessibility calculations require detailed information about the transit routes serving each grid point. Additionally, the results may be investigated alongside other geospatial data. The required breadth of analysis capabilities meant that only certain speciality tools could meet the requirements to combine transportation planning, public transit routing, a GIS engine, custom scripts for data processing, and graphical user interface (GUI) automation. Due to license availability, the team's familiarity with the tool, and the support for the aforementioned features *(15)*, Caliper Corporation's TransCAD software was chosen as the implementation medium by Metrolinx *(12)*.

In the GTHA, transit schedules are published using the GTFS (General Transit Feed Specification) standard for each of the ten service operators (see Section 5 for GTFS details). The GTFS file set for each agency therefore provides a convenient source location for obtaining the transit routes operating at each stop, as required for the ATT and ATD calculations. However, the schedules in this format are incredibly specific and cover individual vehicle level performance (i.e. absolute stop-by-stop timestamps) for every transit run for every route across a service day (see Section 5 for GTFS details). It was decided that requiring vehicle-run level of detail would be overkill for both the baseline (i.e. current network) generalized accessibility measure as well as future, potential transit improvements. Requiring this level of precision for what could be a large number of hypothetical transit projects/alignments/service definitions would have undermined the value of WithinReach as a rapid analysis tool. Consequently, the transit network definition required to run the ATT and ATD calculations became overall route-by-route performance summary for a user-specific time period on a typical service day. The key metrics being the average vehicle headway and the average travel time to the next stop on a single vehicle run of the route *(12)*.

For the first version of the WithinReach tool delivered by Arup to Metrolinx, the GTFS transit schedules for each agency were manually summarized using a combination of the built-in TransCAD functions, as well as database and spreadsheet operations in Microsoft Excel and Microsoft Access. The GTFS file manipulation and pre-processing for a single set of transit schedules from Fall 2014 took several weeks of for an intermediate consultant, with a further week to import and prepare the final geospatial data so that the TransCAD environment could be used to run the ATT and ATD calculations on the existing network, *(12)*. The delivered tool

was sufficient for a period of months, however the renewed focus by the province and GTHA municipalities on improving transit operations to address short-term need *(16)(17)*, has resulted in the baseline regional accessibility scores calculated using the Fall 2014 GTFS file set becoming dated.

While the same manual network building process from before could have been repeated, there was no ongoing added-value in doing this continually to stay on top of schedule improvements that could happen as often as every six weeks. For this reason, the teams at Arup and Metrolinx partnered to build a Transit Network Builder (TNB). The TNB would programmatically and automatically pre-process new GTFS file sets and prepare the TransCAD environment for accessibility calculations within hours of computing time, instead of weeks of manual person-hours. This would allow Metrolinx (and in the future, other transportation planners) to keep baseline networks easily up-to-date, while focusing efforts on evaluating the viability of future projects *(18)*.

The remainder of this report will outline the algorithm design, development, and testing processes. It is broken down into sections to help the reader follow the decision making process and provide some procedural guidance on working with the GTFS standard and using the GISDK scripting language within the TransCAD environment.

3   System Requirements and Overall Procedure

The overall purpose of the Transit Network Builder (TNB) is to make it easy for Metrolinx transportation planners to keep the base-line transit network up-to-date so that ongoing analysis in WithinReach always considers the current state of transit operations in the GTHA. This means that the final output of the TNB has to be a TransCAD workspace that contains the transportation network and all transit services including performance details of each route.   While the automation of both the GTFS analysis and the preparation of the TransCAD environment would grow the development time of the project, this was the best way to ensure the data was being prepared consistently every time and to minimize any future effort on this piece.

From a data input perspective, summarized transit performance data for each route is required to run the WithinReach accessibility tool. In the GTHA, all ten transit operators provide detailed schedule information in the GTFS file structure. Therefore, in order to limit the scope of development and focus on Metrolinx's needs, it was decided that the first version of the TNB would only summarize schedules provided in the GTFS format *(18)*.

Due to the length of time required to analyze the GTFS schedules, route performance averaging was only completed for the AM-Peak service period during the manual preparation of the data. With the computational algorithm directly replacing the hands-on stages, this limitation was removed. During the kick-off meeting, six standard time periods for GTFS summarization were defined (see Table 1). Further to this, the option to specify up to three custom time periods was added to the TNB requirements by Metrolinx *(18)*. This addition of the temporal flexibility further cemented the need for the automated TNB. Due to the labour required, processing time, and (in some cases) data availability, Metrolinx BCAs and planning efforts typically focus on peak-

hour analysis *(8)*. As the availability of transit service normally fluctuates throughout the day,

accessibility metrics can now reflect the regional differences in on versus off peak service.

Table 1 - Six defined standard Time Periods (18)

Time Period Label	Time Period Start (24h clock)	Time Period End (24h clock)
All-Day average	0600	2100
AM Peak average	0600	0900
Midday average	0900	1600
PM Peak average	1600	1900
Evening average	1900	2359
Overnight average	0000	0600

The second key piece of input data was the geographical representation of the integrated

transportation network and region of analysis in TransCAD. To load and process transit routes,

the TransCAD workspace requires an underlying line layer symbolizing the overall transportation

network *(19)*. Additionally, the WithinReach ATT and ATD metrics are calculated based on

subdividing the region into grid-points.   For this application, the Ontario Ministry of

Transportation (via Metrolinx) provided a geographic shape file, called the Ontario Road Network

(20). While this included all of the pavement and roads in the province (and therefore region), it

was missing the heavy rail tracks that GO Transit runs on and the TTC Metro/Subway network.

This was added by Arup GIS specialists in the early stages of the initial project and kept up-to-

date through the opening of new transit-only links like the Mississauga Transitway. Lastly,

because rail transit stations are typically located off the explicit road grid, Arup GIS specialists

created "walking-only" links to connect stations to the roads, and ensure that the ATT and ATD

calculations could evaluate the walking times (i.e. access times) accordingly. Thus, an integrated

transportation network file, including links for roads, heavy rail, metro/subway, and walking-only

links, is an essential requirement for TransCAD. From the perspective of the TNB, it is assumed

that these files have already been prepared, prior to any run of the algorithm *(18)*.

The last requirement was to minimize user intervention during the execution of the TNB.

The volume of data in each GTFS file set is substantial and the previous experience processing

the data manually using Microsoft Access provided the guidance that the algorithm could take

several hours to finish executing. Instead of requiring periodic user inputs to specify data at the

stage where it's needed, a single all encompassing dialog box (see Figure 2) was desired *(18)*.

Figure 2 - Comprehensive Dialog Box (36)

The last system design consideration is TransCAD's built-in GTFS file set. Since 2012,

TransCAD has supported the direct import of GTFS files *(21)*, however experimenting with this at

the start of the TNB process indicated that the implementation would not be appropriate. The

TransCAD GTFS import only supports once agency at a time and so a multi-agency region has to

be merged after the fact. This is not straightforward. Secondly, the GTFS import includes all of

the detailed schedule information and not the summarized route performance needed for WithinReach. Metrolinx required more control over the route summaries, specific time periods, and traceability in the calculations so the built-in functionality was not used *(18)*.

Given the above requirements, the TNB can loosely be broken down into two major segments. The first is the pre-processing and summarization of the detailed GTFS schedule data into an average frequency/headway based aggregation, and the second is the creation of the amalgamated and representative workspace in TransCAD. The next section will outline decisions around the tools selected for implementation.

4 Programming Language Selection

TransCAD is built on the GISDK (Geographical Information System Development Kit) engine, offering a built-in compiler, scripting engine, and the ability to immediately run commands alongside the native GUI (graphical user interface) commands. With the two stage separation of the TNB, there was the opportunity to consider and evaluate external tools that could be used for the first stage of GTFS summarization. In the end, the decision was made by the developer to write the entire algorithm in the GISDK scripting algorithm instead of trying to use an external .NET caller *(18)*.

The primary justification was that GISDK would still have to be learned and used even with a .NET calling structure, which would have complicated the development and troubleshooting process. Additionally, the simplicity of seamless integration between GISDK and TransCAD, the built-in table management and SQL functions, and the GUI interaction were all advantages of maintaining all development within GISDK *(15)*. Using .NET would have added an additional calling and remote execution layer.

The GISDK function set is limited compared to a fully fledged programming language. It is a scripting language that supports 850 functions *(15)*, inclusive of TransCAD GIS and transportation planning specific functions. Not all standard functions, such as array sorting, are normally available, and due to the limited domain of the tool, support resources are not readily found on the internet. This made the initial ramp-up period of using GISDK difficult and initially an external language was preferred for the GTFS step.

The algorithm needed to be launched from a TransCAD dialog box, meaning that even if .NET was preferable for GTFS processing, GISDK would still be needed for the initial setup. In the

end, because there was no avoiding GISDK throughout development, it made sense to go through the initial pain of learning the language up-front. As knowledge of the syntax and functional specifications was gained, the pace of development accelerated. Lastly, the Caliper Corporation TransCAD and GISDK support is excellent, providing an outlet for further resources and training as needed.

The seamless integration of the scripting language with TransCAD was a bonus in using the software. Code could be written in any text editor and was compiled and run directly from TransCAD context menus. There was also the "Immediate Execution" functionality that allowed code to be run immediately on any items open in the TransCAD GUI. This allowed for rapid prototyping throughout development and the ability to try out short sections of code and syntax with instant feedback. For example, if a filter command needed to be tested, the file could be opened using the TransCAD context menu, and "Immediate Execution" used to see if the intended code worked correctly. If an external .NET compiler was used, the algorithm would not have this functionality and a special development environment would need to be used, with some back and forth required to test functionality.

The built-in file and data management was another plus for TransCAD. The initial intention was to use arrays to process the GTFS data, but this drastically increased the processing time and was inefficient. Converting the GTFS .txt files to binary files using TransCAD's built-in file functions allowed the files to be accessed efficiently, including being able to use SQL commands on the filtering and processing. While the TransCAD SQL functions are limited, they are at least integrated and available without needed an external engine as an add-on.

5   Introduction to GTFS

Although each transit agency uses their own software to manage routes, allocate/schedule vehicles (run cutting), and assign drivers (block cutting), the creation of the General Transit Feed Specification (GTFS) by Google *(22)*, has simplified transit schedule reporting and analysis. As discussed in the system requirements, all ten GTHA agencies that Metrolinx works with, use GTFS as a common output format. This streamlined the programming of the TNB as only one type of timetable standard had to be supported as a data input.

The GTFS reference is available online *(23)* and provides information on a common structure for saving and exporting transit schedules. The website includes information on the naming conventions of files and individual fields within each file, as well as required versus optional inputs.

A detailed reproduction of the GTFS reference is beyond the scope of this report however the author has included some high-level guidance on the standard. The notes in the remainder of this section were based on the programmer's experience in using and interpreting the file reference in an actual project. They are being shared to introduce readers to the standard's terminology within the context of the report and may also serve to help future GTFS users in their own understanding of the specification.

While GTFS is intended to be a standard the implementation in practice contains inconsistencies that had to be understood in depth before code could be written to automatically process the files for the TNB. GTFS files are typically published as .zip files on each agency's website or made available online at the parent municipality/government organization's websites

(see Table 3 below). The TNB uses six core GTFS files for analysis, and a short description of these files is provided below.

<u>calendar.txt and calendar_dates.txt</u> *(23)*

The calendar files define which group of transit services, known as "service_ids" operate on a given date. A published GTFS file set includes every transit vehicle run within a date range, for all dates, including weekday versus weekend versus holiday schedules. The simplest use of the "service_id" field is to categorize trips by these three broad service categories. The GTFS standard recommends using the calendar.txt file to specify the typical service_ids that operate on a specific date, based on the day of the week, across a date range, whereas the calendar_dates.txt file should be used to identify specific dates where an exception occurs. Most exceptions are when holiday service is being run on a weekday instead of typical services, implying that the calendar_dates file should be processed after the calendar file.

<u>routes.txt</u> *(23)*

The routes.txt file is an informational summary relating a numerical and unique "route_id" identifier (used in other GTFS files) to the commonly known public route information, such as route name and number. It also specifies the transit technology for a specific line in the "route_type" field.

stops.txt *(23)*

The stops.txt file, like routes.txt, is an informational summary of all transit stops (current and sometimes historical) for an agency and is usually the third largest (longest) file within a file set. The stops file provides a unique stop identifier called "stop_id" and relates it to the commonly (publicly) known stop or station name. It also provides the longitude and latitude coordinates of each stop, which is needed when importing transit services into GIS.

trips.txt *(23)*

This trips.txt file is the first of two major amalgamation files that define transit operations in the GTFS standard and is usually the second longest/largest file. It lists every possible vehicle run (i.e. every time a vehicle starts a run on a route) that exists for every route and all service_ids across all dates that the file set is intended to cover. Each trip is given a unique identifier called trip_id, and each trip also has fields that define the route_id and service_id parameters outlined in the calendar and route files for the sake of cross-referencing.

stop_times.txt *(23)*

The stop_times.txt file is the largest of the GTFS files in a set and connects the geospatial location of transit stops (stop_id) with individual transit vehicle runs (trip_id). The stop_times file provides the order of stops visited by a vehicle in a given run, as well as the absolute 24-hour time stamp that each vehicle is expected to arrive at each stop. It is the most critical file for the TNB because it contains details about stopping patterns for each route as well as time elapsed for a vehicle run.

As an example, a file by file relative size breakdown of the the Toronto Transit Commission's (TTC)

GTFS file set from August 26th 2015 *(24)* is provided in

Table 2. The TTC publishes a new GTFS file set every 6 weeks or so and the data summarized

below includes all services operating between September 6th 2015 and October 10th 2015.

Table 2 - TTC GTFS Summary example (25)

Files	Descriptive GTFS Summaries
Calendar.txt	4 service IDs: 1, 2, 3, 4 Representing Weekday, Saturday, Sunday, and holiday service
Calendar_dates.txt	1 exception date on Monday September 7th (Labour Day Holiday)
Routes.txt	192 routes
Trips.txt	127,895 total transit trips across all dates/service_ids and routes 42,567 transit trips operating on service_id = 1, a typical weekday
Stops.txt	10,737 TTC stops total
Stop_Times	4,800,543 total stop_times for all possible trip_ids 1,563,801 stop_times listed for those trip_ids that operate on service_id=1 dates.

A database relationship diagram which connects the six main files is provided in Figure 3

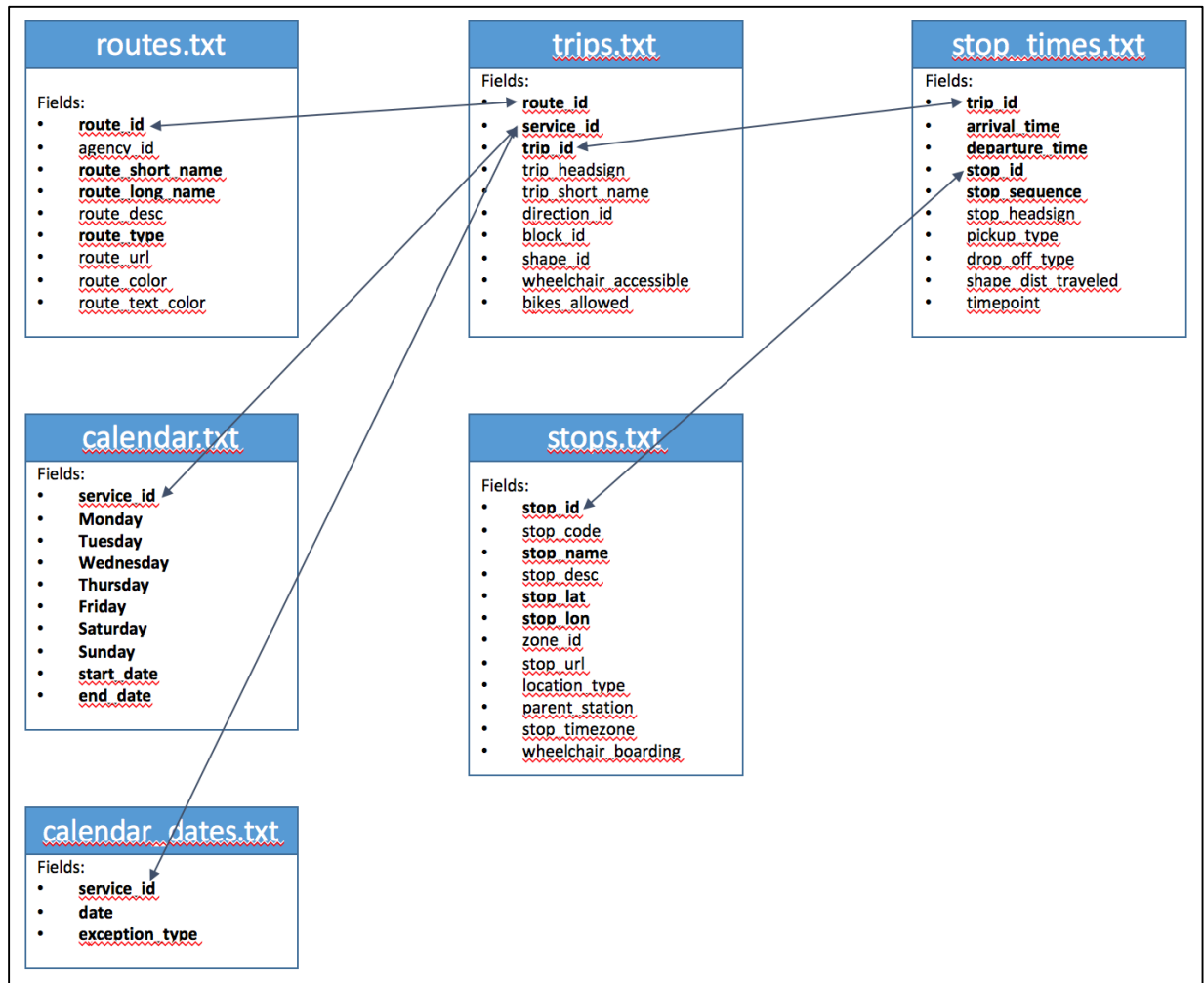below, with the GTFS required fields in bold and the optional fields in normal font.

Figure 3 – GTFS File Field Relationships (26)

6   GTFS Inconsistencies

The GTFS reference *(23)* recommends certain implementation practices for the files and fields. However, because the file sets are generated from 3rd party scheduling software and the output is text-based, these rules are not enforced. This can result in each agency's file set being exported with slight differences, which can be problematic when trying to write a general algorithm. This section highlights specific inconsistencies to be wary of when working with GTFS files from different agencies.

GTFS File Availability & Updating

One of the most challenging aspects of keeping a transit network up-to-date is the manual process for obtaining each agency's GTFS file set and the awareness of new releases. There are no reliable centralized locations containing the latest version for each agency, and categorization online is erratic, rendering the use of Google and common search terms useless. Some GTFS files are easily found on agency/municipality websites, whereas others could only be obtained by contacting the operators directly and/or signing a data sharing agreement. The few aggregator websites that do exist are frequently out-of-date or incomplete.

This "fetching" nature of obtaining GTFS files is further complicated by the differing schedule update cycles between agencies (even within the same region). Some agencies, like the TTC provide updated/improved schedules every six weeks *(24)*, whereas others only update a couple times per year, which means that the recognition of a pattern can take a while Some guidance can be gleaned from the range of dates in the calendar files, however, in general multiple web locations need to be intentionally checked on a regular basis. Table 3 provides the

websites where the GTFS files can be found for the ten GTHA transit agencies.

Table 3 - GTFS File Locations for GTHA Agencies

Agency Name	GTFS File Source Location
TTC *(24)*	http://www1.toronto.ca/wps/portal/contentonly?vgnextoid=96f236899e02b210VgnVCM1000003dd60f89RCRD
GO Transit *(27)*	http://www.gotransit.com/publicroot/en/schedules/DeveloperResources.aspx
Mississauga MiWay *(28)*	http://www.mississauga.ca/portal/miway/developerdownload
Brampton Transit *(29)*	http://www.brampton.ca/EN/City-Hall/OpenGov/Open-Data-Catalogue/Pages/Data-Set-Details.aspx?ItemID=1
Oakville Transit *(30)*	http://www.oakville.ca/data/oakville-transit-route-information.html
Burlington Transit *(31)*	http://cob.burlington.opendata.arcgis.com
Milton Transit	Not available publicly online – Contact agency or Town of Milton for more information
York Region Transit *(32)*	http://www.yorkregiontransit.com/en/aboutus/GTFS.asp
Hamilton Street Railway *(33)*	https://www.hamilton.ca/city-initiatives/strategies-actions/open-accessible-data
Durham Region Transit	Unavailable online/publicly – requires data sharing contract with Region of Durham

Data management

Vehicle level trip details are exported from industry standard scheduling into text format. The

standard cannot assert any rules with respect to data validation, cleanup, and management. It

is therefore up to the individual transit schedulers at each agency to manage and remove stops

that are no longer operational, eliminate trips and service dates in the past, and to not store

schedule versions or variants as different "service_ids." All of these scenarios were encountered

when analyzing the ten GTHA agencies and resulted in several identifiers existing in one file, but

not in the others that are cross-referenced *(26)*. The TNB had to be flexible enough to handle

data that is imperfect.

Calendar Files and Service IDs

The calendar and calendar_dates files are intended to work in tandem to define which service_ids operate on each specific date. According the GTFS reference *(23)*, the calendar file should be read first to define the typical service that operates, with the calendar_dates used to identify exceptions such as holidays or extended service changes such as a major detour lasting more than a couple days. The recommended practice is for each date to have one service_id however some agencies sub-divide service_id operations by geography, detours, route types (express/local), and/or vehicle categories. This means that any automation software must be able to handle a single date having multiple service_ids.

Other agencies, such as GO Transit *(27)*, use the calendar_dates method of handling every date as an exception, omitting the use of calendar.txt entirely. Handling every calendar date as an exception can be advantageous if there are different services being offered every single day of the year, however in practice this is unlikely and is be a customer service and way-finding nightmare. Any automated software needs to be able to automatically detect both variants.

Missing Stop Time Data

The stop_times.txt file *(23)* is intended to provide the time of arrival and departure for every transit vehicle on every run for every trip in the schedule. Some transit agencies instead use timing points and only provide absolute arrival and departure times for only a portion of stops on a trip. In the GTHA, three of the ten transit operators use this method of reporting schedules.

For consistency in importing and analyzing the schedules, the missing stop times will need to be backfilled. This can be done by calculating the total running time of a transit trip, and using the total travelled distance and the assumption of constant speed between timing points to linearly interpolate and fill in the missing data.

Route Identification

In GTFS, the routes.txt file *(23)* contains descriptive route information to connect the unique GTFS identifiers to the public route names. One of the issues is that GTFS, and by corollary the scheduling software, does not explicitly distinguish between route branches, route variants, and scheduled short turn operations. These different stopping patterns are reported in GTFS as the same physical route number. This lack of detail is problematic in the accessibility tool as the variants each have unique service frequencies and travel speeds. The routes will need to be separated given the level of analysis being done.

Data Format and Column Order

The GTFS file format specifies that the files in the set must be output as .txt files, and requires certain column labels to appear *(23)*. It does not specific the order of the columns or the data types in each. Lastly it does not specify the formatting standard for the text files. Within the GTHA, two different text formats are used, ISO-8859-1 and UTF-8-VOM, and there is inconsistency in the presence and order of data fields.

Because certain fields (data columns) are key in GTFS analysis, all fields first need to be scanned to determine order before comma-delimiting and the conversion to a table can occur.

Also, the ISO text format prepends text files written in its format with a 3-character identifier code. In simple text editors this can cause an error when looking for specific column names that don't exist because the first entry in the text file has additional characters. When writing software, an extra conditional processing step will need to be written and executed if an exact column match does not occur.

7   Algorithm Summary

The previous sections have outlined some of the difficulties encountered when working with the GTFS file set and requirements for preparing the TransCAD environment to run the WithinReach accessibility. In the end, the final TNB algorithm exceeded 4900 lines of GISDK code, including comments *(34)*. While the code itself cannot be included in this report due to confidentiality, this section provides a high-level step-by-step breakdown of the algorithm's operations. Figure 4 is an overall process diagram, showing the dialog box, intermediate stages, and the look of the final TransCAD environment when the accessibility tool is ready-to-run.
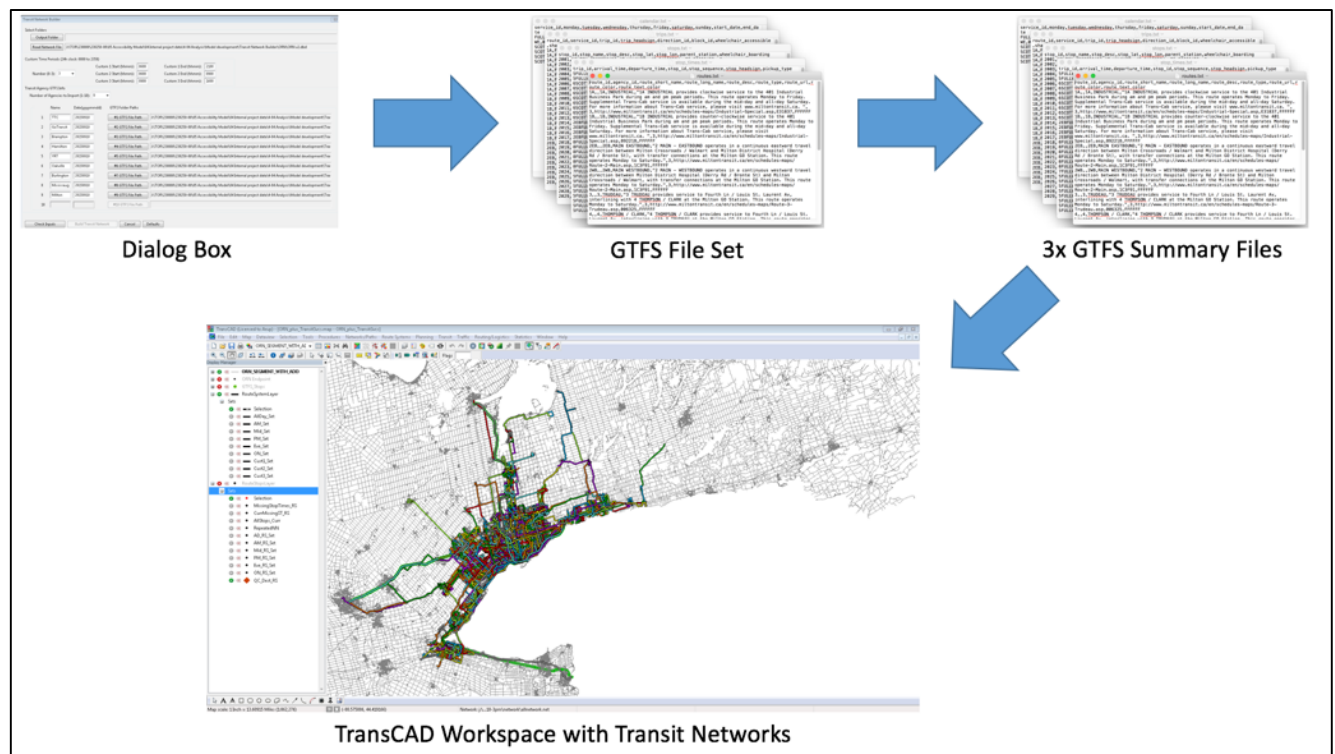

Dialog Box   GTFS File Set   3x GTFS Summary Files

TransCAD Workspace with Transit Networks

Figure 4 - Algorithm Summary (35)

Dialog Box Inputs (see Figure 5)

Figure 5 - Dialog Box including sample inputs (36)

- Final Outputs Folder Location

    o Final location for all GTFS Summary files (including intermediate files generated by TNB process), and TransCAD Workspace files

- Transportation Network File Location

    o Input that specifies the location of the transportation network shape/geographic database file

- Custom time periods analysis

    o Dropdown menu allowing to user to add from 0-3 additional time periods for network performance averaging

        ▪ Additional fields to define the start and end times of each custom time period become available depending on the number selected above.

- Transit Agency inputs

  o Ask the user how many transit agency GTFS file sets (1-10), that will be used to generate the summary files and then imported into TransCAD

    ▪ The user is asked to name each agency, specify the calendar planning date for that agency, and specify the folder location of the GTFS files

Generation of 3 Summary Files

Three GTFS summary files are generated in the algorithm in the first stage of the process that will be filled in as the averaging proceeds.

1. List of all stops (AllStops)

   o Creates a master stop list in one file that assigns a unique stop_id to each entry

   o Opens the stops.txt file for each agency

   o Also includes the following stops.txt data:

     ▪ Agency Name

     ▪ Stop ID as written in the GTFS stops.txt file for each agency. This field is not checked for uniqueness as multiple agencies may use the same stop identifier, however the first field - unique stop id field allows for the separation

     ▪ Stop name

     ▪ Latitude

     ▪ Longitude

2. List of all Route Information (RouteInfo)

- o Will be filled in as routes are summarized in the first stage of the process, but needs to include the following fields:

  - Unique Overall Route ID - unique identifier for each individual route variant discovered by analysing stop_times.txt stopping sequences for all agencies

  - Agency Name

  - Route ID as identified in the GTFS

  - Route Variant – sub category of route IDs based on unique stopping patterns

  - Route Technology – transit technology identifier from GTFS

  - All Day average headway

  - AM Peak average headway

  - Midday average headway

  - PM Peak average headway

  - Evening average headway

  - Overnight average headway

  - Custom Time Period 1 average headway

  - Custom Time Period 2 average headway

  - Custom Time Period 3 average headway

  - Route Short Name

  - Route Long Name

  - Stopping Sequence

3. List of all Routes including the stops (RouteStops)

- This is the equivalent of an overall stop_times.txt file for all agencies but only have 1 trip or vehicle run for each unique route variant

- Starts as blank that will be filled throughout the algorithm with the following fields:

  - Unique Route Stop identifier

  - Unique Overall Route ID (matching the RouteInfo file)

  - Agency ID

  - Route ID as identified in the GTFS

  - Route Variant

  - Stop Order Number

  - Stop ID as identified in the Agency's GTFS

  - The ID of the next stop in the route

  - All Day average travel time to next stop

  - AM Peak average travel time to next stop

  - Midday average travel time to next stop

  - PM Peak average travel time to next stop

  - Evening average travel time to next stop

  - Overnight average travel time to next stop

  - Custom Time Period 1 average travel time to next stop

  - Custom Time Period 2 average travel time to next stop

  - Custom Time Period 3 average travel time to next stop

<u>Stage 1 Summary of GTFS files</u>

- Determine the set of Service IDs that are active on the planning date for each agency

    o Take the user inputted planning date and determine day of the week

    o First try to open calendar.txt for each agency if it exists, if it doesn't go straight to calendar_dates.txt

    o If calendar.txt exists

        ▪ Use the date range and day of the week to determine the normal/standard service ID set

    o Always open calendar_dates.txt if it exists

        ▪ Search for the planning date in the file. If it doesn't exist, then there are no exceptions and leave service ID set as is

        ▪ If the planning date exists in the file, add/remove the service ID previously specified in calendar.txt based on the Exception_Type column in the file

- Filtering trips.txt files

    o For each agency, merge the routes.txt and trips.txt files and save the resulting master file as a fixed format binary file

    o Using the set of "active' service IDs for each agency on the chosen planning date, filter the combined trips file, removing any trips that don't run on the planning date

    o Store the list of active/running trip_ids

- Filtering stop_times.txt file

    o Open and save the stop_times file as a fixed format binary

- Use the filtered list of active trip_ids to remove any trip_ids from stop_times that don't match, creating a smaller data set

- Identifying Route Variants, Trip Time of Day operations, Average Travel Times

  - Progress through each agency's stop_times file only once, line by line, as they are large

    - Sort the stop_times file by trip_id and the stop_sequence value

    - Create an intermediate file used to track unique route variants for each agency (AgencyRouteVariantMaster)

    - Create a temporary column to identify the route_variant within the an overall route_id identifier

    - Create column(s) to identify if trips occur within any of the 6-9 time periods for analysis

    - Calculate the relative travel time between stops based on stop 1 arrival time minus stop 0 departure time, and store in newly created TTNS (time to next stop) columns for each time period

      - If an agency uses timing points instead of every stop stop_times, then skip this step and instead fill in the TTNS based on the entire trip run time, with the backfilling to be done later once the distance travelled information is available

    - Keep track of the order of stop ids to establish a trip's stopping sequence and store as a temporary variable

- When the last stop in a trip is encountered (there will be a blank in the next stop column), the stopping pattern for the trip is compared versus the AgencyRouteVariantMaster file
  - If stopping pattern already exists, then fill the route_variant to match
  - If it does not, then add the new route variant to the AgencyRouteVariantMaster file, give it a unique id, copy the stored stopping pattern, and fill in the temporary route_variant column with the new route_variant sub ID
- Route Information summaries (RouteInfo)
  - Using the temporary columns in the previous step, use the length of each time period, and the number of transit vehicle trips for each route variant within each time period to compute the headway for each
  - Fill in the Route data, including route variant identifier, average headway for each time period, and the route variants stopping pattern to the RouteInfo summary file
- Travel Time averaging (RouteStop File summary)
  - Re-open the stop_times binary file that was created including the temporary columns
  - Create a temporary column that is a text-based aggregate of the agency name, route id, route variant, current stop, and next stop

- Using the built-in GISDK aggregate table function and compute the average time to next stop for each time period using this temporary columns unique name (which is the route variant specific performance for a given stop pair)

  o Create an entry in the RouteStop file for each route, route variant, and stop in order.

  o Fill the average TTNS for each pair as defined above which will reflect the average performance of a route

Stage 2 – TransCAD integration

- Create a new TransCAD map/workspace

- Import the transportation network geographic files

- Using the attributes table for each link in the network, use SQL commands to generate a subset of all links that are:

  o Walking Links (combination of roads, walking-only, but not freeway or restricted access)

  o Metro/Subway Only Links

  o Road Links (that can run private vehicles or transit buses or LRT/streetcar)

  o Regional Commuter Heavy Rail

- Generate a TransCAD network file for walking

- Import the AllStops summary file into TransCAD as a point layer, so all transit stops in the region are now available

- Group the transit stops by the transit technology each stop supports (based on the GTFS route_type column in the RouteInfo column)

    o Bus_streetcar_LRT stops

    o Metro_Subway stops

    o RegionalHeavyCommuterRail stops

- Create new groupings for the transportation network links based on whether there are transit stops identified in the summary files that actually operate on the planning date. This will create a subset of links that are active for transit in the analysis then create a TransCAD network file for transportation network links that operate transit on the planning date

- Create a TransCAD route system by importing the RouteInfo and RouteStop files

    o This will create the transit route "spaghetti" as can be seen in Figure 6

    o Create RouteSystem subsets based on the route variants that operate in each time period

- Based on the overall route systems created in the previous step, create a TransCAD "transit network" for each of the 6 (or up to 9) time periods of the day

Once all of these steps are complete, the TransCAD map will have five layers as seen in Figure 6 below, as well as a file folder that includes two types of basic network (transit links, and walking links), as well as between six and nine transit network files. These files are used along with the five geographic layers in the accessibility tool.
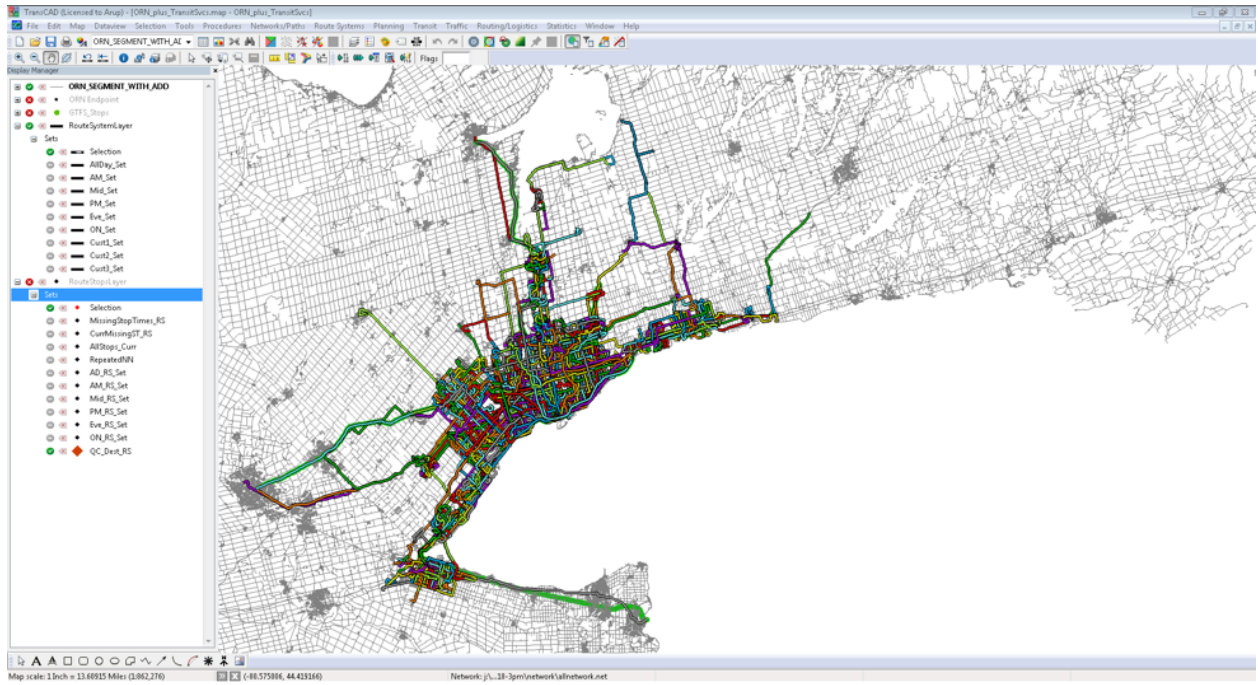
Figure 6 - Final TransCAD output (37)

8   Testing and Validation

The development of the algorithm was based on the previous manual method for summarizing and averaging the GTFS files, followed by a TransCAD import. A comprehensive and detailed set of notes were not available as the initial version proceeded on a trial-and-error basis, however Arup consultants were able to provide some ongoing guidance and rough notes for the process to be replicated in GISDK *(38)*. The algorithm was programmed sequentially by a single developer, breaking down individual steps to match the the operations and outputs generated manually using Microsoft Excel. Development notes were kept in rough format in a hard copy notebook *(39)* and in a quality control summary file in Excel (see Figure 7 below).

Due to the amount of data in a single GTFS file set, a scaling process was used to validate the initial filtering operations. This was done by varying the size of the GTFS input files from a limited custom set, to a smaller agency, to a larger agency, and then finally multi-agency. The custom GTFS file set was created by choosing a specific service_ID and set of route_ids from the smallest agency (Milton) and filtering/deleting data using Excel across the six core GTFS files to match. This drastically reduced the file sizes to a couple kilobytes each (dozens of entries), and meant that TNB outputs in each stage could be validated visually against the raw data. Once the algorithm worked for the "custom" small transit agency, each GTFS file (for Milton) was restored one at a time, and rerun. If it worked using the full Milton data, then a larger agency like Mississauga or Toronto was used. This process worked well because larger datasets could always be filtered in Excel or TransCAD and compared to the files generated by the algorithm.

Agency Num	Name	Last GTFS Update	Which Planning Date?	Service IDs	Rte Trip Merge (Tot Trips)	# Stop Times	# Routes	# Route Vars	Avg Vars/Rte	Stops /Trip	# Stops	Seconds to Gen Intermed	Minutes	hours	ST/sec	trips/sec
1	TTC	Aug 29 15	Thurs Sept 10	1	42567	1563801	192	1129	6	37	10737	27119	452.0	7.5	57.7	1.6
2	GO Transit	Aug 19 15	Thurs Sept 10	4579-Thu, 4601-Thu	2676	42606	101	283	3	16	1796	609	10.2	0.2	70.0	4.4
3	Brampton	Aug 11 15	Thurs Sept 10	150907-MULTI-Weekday-01	4695	179200	110	206	2	38	2832	2522	42.0	0.7	71.1	1.9
4	Hamilton	Aug 27 15	Thurs Sept 10	1_merged_1016403	3686	168433	66	132	2	46	2452	2371	39.5	0.7	71.0	1.6
5	YRT	Sep 11 15	Thurs Sept 10	1, 2461001, 2461101, 2461201	4560	176894	138	461	3	39	4813	2675	44.6	0.7	66.1	1.7
6	Oakville	Aug 14 15	Thurs Sept 10	01-Weekday	1193	40825	26	83	3	34	1160	569	9.5	0.2	71.7	2.1
7	Burlington	April 2015	Thurs Sept 10	WKD	1040	36719	33	76	2	35	899	514	8.6	0.1	71.4	2.0
8	Mississauga	August 15	Thurs Sept 10	15SE07-CPBlock-Weekday-11, 15	6300	263159	88	275	3	42	4516	4015	66.9	1.1	65.5	1.6
9	Milton	Sept 8 15	Thurs Sept 10	FULLW, SCOTTDETOUR	218	9532	14	25	2	44	352	76	1.3	0.0	125.4	2.9
10	Durham	Sept 2015	Thurs Sept 10		2682	113116	86	193	2	42	2595					

Figure 7 - QC Summary of Sept 10 Planning Date (40)

Algorithm validation for the latter stages proceeded in a similar fashion, typically starting with a limited data set and then scaling up to an entire GTFS schedule. Due to the size of the data, a randomized spot-checking method was employed along with the generation of temporary intermediary summary files *(39)* (see Figure 8). Once one of the larger, summary operations was performed, a number of individual routes and stops within an agency were queried in Excel and checked in the intermediary files to ensure that the results "made logical sense." While this required some faith in the back-end aggregation and math operations of GISDK, the instantaneous feedback of developing the code line-by-line using the Immediate Execution mitigated the concerns.

The final transit network was completed after the importing and creation of the regional "Route System" in TransCAD. Twenty-five random intersections across the region were selected at random and queried on the TransCAD map to provide a list of routes serving each intersection for the amalgamated transit network built from the TNB GTFS summaries (see Figure 9). The list of routes at each intersection in TransCAD that manually checked against each agency's route/system maps (see Figure 10).
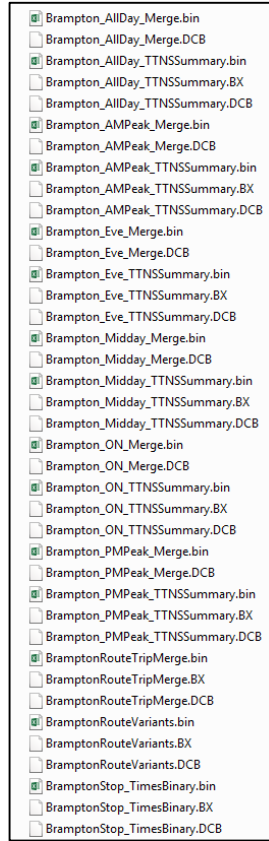
Figure 8 - Example of Intermediate Files used for validation (41)

Figure 9 - Checking routes servicing each intersection (42)

Main operator	Municipali	ORN Endpoint "ID" Field	Location	Routes serving Intersection	# of Route Variants in TransCAD to check Route IDs
Brampton	Brampton	247328	Airport Rd & Castlemore Rd/Bovaird	5A, 30, 5, 505, 35, 35A	18
Brampton	Brampton	200827	Bovaird Dr & Worthington Ave	4, 4A, 5, 5A, 9, 505	45
Burlington	Burlington	59375	Brant St & Plains Rd	3, 3x, 5, 52, 51, 21, 25, 1, 1x, 101	53
Burlington	Burlington	150051	Appleby Line & Upper Middle Rd	11, 15A, 15B, 51, 12, 12x, 48	13
Burlington	Burlington	295648	Appleby Line & Spruce St	20, 40, 50	5
DRT	Ajax	331035	Fairall St & McMaster Ave	226, 224, 224D, 218, 218D, 916	8
DRT	Oshawa	277069	Harmony Rd & Adelaide	410, 412	6
HSR	Hamilton	325386	James St N & King William St	20, 2, 3, 4, 7, 8, 9, 6	13
HSR	Hamilton	83550	Upper Gage Ave & Mohawk Rd E	23, 41, 41A	12
Milton	Milton	149660	Main St E & Thomson Rd S	1A, 1B, 2	10
Milton	Milton	225843	Bronte S S & Derry Rd	2, 7, 8	14
MiWay	Mississauga	53511	Erin Mills Pkwy & Britannia Rd	9, 48, 87, 39, 10	24
MiWay	Mississauga	283342	Dixie Rd & Queen St	4, 5	24
MiWay	Mississauga	46939	Hurontario & Bristol Rd	103, 10, 19, 19A, 19B, 65	17
Oakville	Oakville	193340	Third Line & Upper Middle Rd	3, 6	6
Oakville	Oakville	359392	Maple Grove Dr & Cornwall Rd	4, 11, 21, 25	12
Oakville	Oakville	326831	Trafalgar Rd & QEW	1, 13, 19, 20, 24, 26, 120, 190	25
TTC	Toronto	142578	Islington Ave & Bloor St	2, 50, 110+, Miway -	92
TTC	Toronto	23562	Spadina Ave & Queen St	510, 501	30
TTC	Toronto	282135	Markham Rd & McNicoll Ave	102B, 102C, 102D	11
TTC	Toronto	61708	Bayview Ave & Eglinton	103, 141, 74, 51, 54A, 54B, 56A, 34, 100C	36
TTC	Toronto	137496	Weston & Finch	165+, 36+	30
YRT	Vaughan	194188	Bathurst & Rutherford Rd	88, 88A, 16, 85, 85C, 23	24
YRT	Keswick	139336	Woodbine Ave & Biscayne Blvd	51, 424	

Figure 10 - Validation of final GTHA TransCAD Route System (43)

The final test was to see if the TNB would build a network for a set of agencies based on GTFS files that had never been processed through the algorithm. In this case, the GTFS files for the four agencies operating transit services in the Montreal area and a geographic database were obtained. The resulting TransCAD workspace (see Figure 11) was built without any errors, and proves that the algorithm can be used to prepare networks for WithinReach that it wasn't designed with.
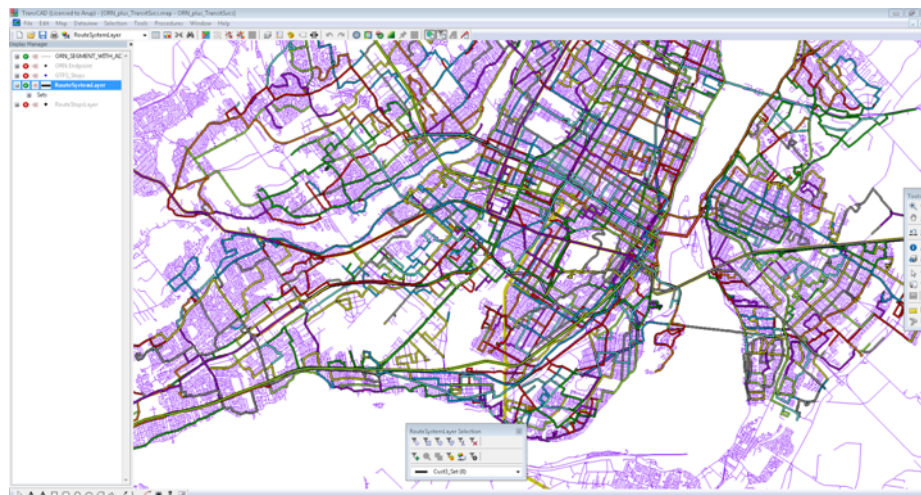

Figure 11 - Montreal TransCAD Network built with TNB (44)

9   Conclusion

As outlined in this report, the development of an automated Transit Network Builder involves expertise in the GTFS file format, pre-existing knowledge of each agency's operations, and detailed knowledge of the TransCAD/GISDK environment. Overall the project, including the initial training to understand the GTFS file set, TransCAD/GISDK familiarization, algorithm conception, development, and testing was a full-time job for an intermediate transportation consultant for four months. This allowed for the development of a complex algorithm that can generate a complete and accurate TransCAD based transportation network, which can be used for multiple forms of analysis within hours instead of weeks. The automatic summarization of route performance based on low-level scheduling details from GTFS files was a key component of the algorithm and including analysis for multiple time periods was a large added bonus. Although the tool development required substantial investment, the effort will be recuperated within the first few runs/updates of a baseline transit network due to the time it saves consultants to focus on more value-added tasks instead of data processing. The intention of the report was to supply the reader with an introduction to the GTFS file standard, considerations when working with the file set, decision making around TransCAD, and an outline of the algorithm that was developed. It can and should be used as reference when considering any project that includes the GTFS standard, or if using GISDK in TransCAD.

10   Recommendations

The Transit Network Builder was purpose built for Metrolinx's ongoing analysis of transit availability in the region. The exercise served as an excellent introduction to both the GTFS standard and the GISDK environment and those skills will be reusable in the future. The Montreal example also proves that the algorithm was flexible enough (in general) to build a TransCAD workspace from GTFS files that were not part of its design and validation process. The following are recommendations to the reader for software development using GISDK and TransCAD, working with GTFS, and further algorithm improvements.

1. When first learning GISDK, the functional support is limited and does not include many examples. Additionally, online resources are few and far between. To become proficient at GISDK, it is recommended that a user use a combination of:

    a. Properly documented/commented example code

    b. A good mentor/reference

    c. Access to TransCAD/GISDK e-mail support

    d. and/or Complete a formal training class

2. The implementation of transit networks in TransCAD for both display and processing requires a geographic line layer, the use of the route system creation/import tool, and the creation of a Transit Network file. Similar to the guidance provided for GISDK, formal training should be pursued to completely understand why the software has these different components and the interplay/advantage of its methodology

3. Understanding the GTFS format to be able to analyze it is not a simple task. To make learning the intricacies of the standard easier, the recommendation is that users start

analyzing a network they are deeply accustomed to. It will be more obvious how the different files and fields relate if there is some pre-conceived context and familiarity to the names and places.

For the TNB algorithm the following recommendations are suggested.

1. Allow more flexibility in how the geographic file is prepared including:

   a. Automatic creation of walking links where transit stations are off the road grid

   b. Flexibility in naming fields, and giving the user choices around the sub-categorizations of route types

   c. Automatic detection of layer and field names to be added to a dropdown menu in the dialog box

2. Open up the schedule importing to support non-GTFS file formats

3. Currently LRT and BRT lines are classified as standard in-medium road based services or as metro-subway rail. A new category may be required as these types of lines come online in the GTHA to support the many implementation variants

4. Automatic validation – it would be beneficial to develop a test to link the summarized GTFS files and TransCAD route systems back to an agency's system map or route listing to help an end-user verify their own region's imported transit networks

Provide the user suggestions on the best planning date to use. Today the user guide advises that a typical non-holiday weekday is the best date to select, however more detailed guidance could be beneficial for those regional rail agencies where the services being run are slightly different every day.

References

(1)   "Metrolinx Overview," Metrolinx, Toronto ON, n.p., n.m., Accessed – Jan 14 2016, Web:
        http://www.metrolinx.com/en/aboutus/metrolinxoverview/metrolinx_overview.aspx

(2)   "The Big Move – Transforming Transportation in the Greater Toronto and Hamilton
        Area," Metrolinx, Toronto ON, Published – November 2008, n.m., Accessed – Jan 14
        2016, Web:
        http://www.metrolinx.com/thebigmove/Docs/big_move/TheBigMove_020109.pdf

(3)   "TORONTO'S VITAL SIGNS REPORT 2015 – Getting Around," Toronto Foundation,
        Published – Oct 2015, n.m., Accessed – Jan 14 2016, Web:
        https://torontofoundation.ca/sites/default/files/OP-TVS%202015-Full-Report-
        PRINTING.pdf

(4)   Levy, E., "Rapid Transit in Toronto – A Century of Plans, Progress, Politics, & Paralysis,"
        BA Consulting Group, Toronto ON, 2015, Book.

(5)   "Bill 163 - An Act to amend the Greater Toronto Transportation Authority Act, 2006 and
        to make consequential amendments to another Act," Legislative Assembly of Ontario,
        Province of Ontario, Royal Assent & Published – May 14 2009, n.m., Accessed – Jan 14
        2016, Web:
        http://www.ontla.on.ca/web/bills/bills_detail.do?locale=en&BillID=2167&detailPage=bil
        ls_detail_status

(6)   "Regional Planning," Metrolinx, n.p., n.m., Accessed – Jan 14 2016, Web:
        http://www.metrolinx.com/en/regionalplanning/regional_planning_index.aspx

(7)   "Business Case Development Manual," Mayor of London UK - Transport for London
        Programme Management Office, Published – May 2013, Web:
        https://www.whatdotheyknow.com/request/197881/response/495035/attach/3/Busin
        ess%20Case%20Development%20Manual%20May%202013%20Redacted.pdf

(8)   "Business Case Analysis," Metrolinx, n.p., n.m., Accessed – Jan 14 2016, Web:
        http://www.metrolinx.com/en/regionalplanning/projectevaluation/benefitscases/benef
        its_case_analyses.aspx

(9)   Schabas, M., "REVIEW OF METROLINX'S BIG MOVE," Neptis Foundation, Toronto ON,
        Published - December 2013, n.m., Accessed – Jan 14 2016, Web:
        http://www.neptis.org/sites/default/files/metrolinx_review_2013/neptis_schabas_repo
        rt_dec_2013_finaljuly23.pdf

(10)   Fisher, D., "Next Generation Transportation Business Cases: Supporting the Optimization of Transportation Investments in the GTHA," Metrolinx, Presented at Transportation Futures – Road Pricing and Planning Workshop, Hilton Doubletree Hotel, Toronto ON, Nov 24 2014

(11)   "Measuring Public Transport Accessibility Levels," Transport for London, Published – May 2010, n.m., Accessed – Jan 14 2016, Web: http://data.london.gov.uk/dataset/public-transport-accessibility-levels/resource/86bbffe1-8af1-49ba-ac9b-b3eacaf68137/proxy

(12)   "Ongoing meetings and discussions, including iterative development details, between Metrolinx and Arup Canada as part of services contract", Fall 2013 to December 2015. Internal Arup Meeting Notes and Records.

(13)   Lukka, R. and Camargo, F., "WithinReach output sample from TransCAD," Arup Canada, Toronto ON, Oct 14 2015. Internal Arup Document.

(14)   Camargo, F., "Technical Note: Accessibility Model methodology," Arup Canada, Toronto ON, July 17 2015. Internal Arup Document.

(15)   "TransCAD Transportation Planning Software," Caliper Corporation, Newton MA, n.p., n.m., Accessed – Jan 14 2016, Web: http://www.caliper.com/tcovu.htm

(16)   "TTC to rollout 61 new and improved bus routes across the city to all-day-every-day and overnight bus, streetcar service", Toronto Transit Commission, Published - May 24 2015, n.m., Accessed – Jan 14 2016, Web: https://www.ttc.ca/News/2015/May/5_24_015_NR_service_improvements.jsp

(17)   "Improving our service," Metrolinx GO Transit, n.p., n.m., Accessed - Jan 24 2016, Web: http://www.metrolinx.com/en/regionalplanning/rer/rer_service.aspx

(18)   "Meeting Notes – Transit Network Builder Kickoff, August 2015," Attended by: Rishi Lukka & Felipe Camargo of Arup Canada, and Anna Kramer of Metrolinx. Toronto ON. Internal Arup Meeting Minutes.

(19)   Runey, L., "Route System Coding in TransCAD," ITRE, Raleigh NC, Published – Oct 25 2005, n.m., Accessed – Jan 16 2006, Web: https://connect.ncdot.gov/projects/planning/TPB%20Model%20User%20Groups/ITRE%20Route%20System%20Coding.pdf

(20)   "Ontario Road Network GIS Shape File," Ontario Ministry of Transportation, Provided to Arup Fall 2014 by Metrolinx

(21)   "TransCAD 6.0 is Released," Caliper Corporation, Newton MA, Published – Aug 14 2012,
         n.m., Accessed – Jan 16 2016, Web: http://www.caliper.com/press/pr20120814.htm

(22)   Roth, M., "How Google and Portland's TriMet Set the Standard for Open Transit Data,"
         StreetsBlog SF, Published – Jan 5 2015, n.m., Accessed – Jan 16 2016, Web:
         http://sf.streetsblog.org/2010/01/05/how-google-and-portlands-trimet-set-the-
         standard-for-open-transit-data/

(23)   "General Transit Feed Specification Reference," Google Inc., Mountain View CA, n.p.,
         Last Modified – Feb 2015, Accessed – Jan 16 2016, Web:
         https://developers.google.com/transit/gtfs/reference

(24)   "TTC Routes and Schedules," City of Toronto Open Data Catalogue, Published – August
         26 2015, Last Modified – Jan 11 2016, Accessed – Jan 14 2016, Web:
         http://www1.toronto.ca/wps/portal/contentonly?vgnextoid=96f236899e02b210VgnVC
         M1000003dd60f89RCRD

(25)   Lukka, R., "Summary of TTC August 26 2015 GTFS File Set – based on TTC Open Data
         (24)," Arup Canada, Toronto ON, September 2015

(26)   Lukka, R., "GTFS File Relationship diagram based on GTFS Reference (23)," Arup Canada,
         Toronto ON, December 2015

(27)   "GOtransit.com – Developer Resources," GO Transit - a division of Metrolinx, n.p., Last
         Modified – Jan 8 2016, Accessed – Jan 16 2016, Web:
         http://www.gotransit.com/publicroot/en/schedules/DeveloperResources.aspx

(28)   "MiWay Developer Download," City of Mississauga MiWay, n.p., Last Modified – Dec 4
         2015, Accessed – Jan 16 2016, Web:
         http://www.mississauga.ca/portal/miway/developerdownload

(29)   "Open Data Set Details," City of Brampton, n.p., Last Modified – Dec 23 2015, Accessed
         – Jan 16 2016, Web: http://www.brampton.ca/EN/City-Hall/OpenGov/Open-Data-
         Catalogue/Pages/Data-Set-Details.aspx?ItemID=1

(30)   "Dataset: Oakville Transit Route and Schedule Information," Town of Oakville, Published
         – May 1 2014, Last Modified – Dec 15 2015, Accessed – Jan 16 2016, Web:
         http://www.oakville.ca/data/oakville-transit-route-information.html

(31)   "Open Data Catalogue," City of Burlington, n.p., Last Modified – Jan 15 2016, Accessed –
         Jan 16 2016, Web: http://cob.burlington.opendata.arcgis.com

(32) "Google Transit Feed Specification (GTFS)," York Region Transit VIVA, n.p., Last Modified – Jan 12 2016, Accessed – Jan 16 2016, Web: http://www.yorkregiontransit.com/en/aboutus/GTFS.asp

(33) "Open & Accessible Data," City of Hamilton, n.p., n.m., Accessed- Jan 16 2016, Web: https://www.hamilton.ca/city-initiatives/strategies-actions/open-accessible-data

(34) Lukka, R. & Camargo, F., "Automated Transit Network Builder Source Code," Arup Canada & Metrolinx, Toronto ON, December 2015

(35) Lukka, R., "Transit Network Builder General Algorithm Flow Diagram," Arup Canada, Toronto ON, December 2015

(36) Lukka, R., "Transit Network Builder Dialog," Arup Canada, Toronto ON, December 2015

(37) Lukka, R., "Sample of final TransCAD Workspace environment," Arup Canada, Toronto ON, December 2015

(38) Camargo, F., "Notes for preparing GTFS files for TransCAD import," Arup Canada, Toronto ON, August 2015

(39) Lukka, R., "Personal Notebook for Arup projects including development notes for TNB," Arup Canada, Toronto ON, Used April 2015 to present.

(40) Lukka, R., "Quality Control of TNB Development – Sept 10 Planning Date," Arup Canada, Toronto ON, December 2015

(41) Lukka, R., "Sample of Intermediate File Generation during TNB Execution for Validation," Arup Canada, Toronto ON, December 2015

(42) Lukka, R., "Final GTHA Network Validation – Services at each intersection," Arup Canada, Toronto ON, December 2015

(43) Lukka, R., "Final GTHA Network Validation – Cross referencing system maps – Excel QC Sample," Arup Canada, Toronto ON, December 2015

(44) Lukka, R., "Montreal TNB generated TransCAD Network," Arup Canada, Toronto ON, January 2016