

THE DEVELOPMENT OF A NEW TRACTION POWER SYSTEM SIMULATOR

Daniel P. Truesdell
Gannett Fleming, Inc.
Lebanon, NH 03766

Jian G. Yu
Gannett Fleming, Inc.
Lebanon, NH 03766

ABSTRACT

A new suite of traction power system simulator tools has been developed to meet the engineering and design requirements of railroad and rail transit projects. The simulator has been designed to deal with both dc and ac systems and to track closely with the latest technological developments and trends in the industry.

For the dc applications, features include emulation of wayside energy storage devices, reversible rectifier substations, stray current and automatic grounding switches. For ac applications the tool supports simple, autotransformer and booster transformer feeding systems.

The simulator is designed to operate with other suites of software that are available in the transit rail industry. For example, train performance data can be fed into the traction power simulator from Train Performance Calculator (TPC) or network simulator packages.

This paper discusses the approaches taken in handling some of the more complex issues that arise in load flow analysis, including reduction in solution processing time using efficient computation algorithms, component parallelization to exploit multiple processors and multiple computers, power demand adjustments for underpowered or non-receptive conditions, and flexible and comprehensive textual and graphical post processing.

INTRODUCTION

Utilizing computer technology in pursuit of engineering solutions has been a sought-after approach since the advent of the microprocessor. Using this technology for electric circuit analysis has proven to be a worthy approach for many years. However, the complexity of many electric networks, and their underlying mathematical models, often push the limits of available computing power. As computers evolved into systems with more and more megaflops and megabytes,

electrical networks and their components also evolved into far more complex models than previously possible.

In addition to the innate challenges in electric network modeling, the ever-expanding capabilities presented by advancements in database engines, operating systems, inter-process and network communications, and user interface models have added to the complex nature of engineering systems analysis. The myriad of avenues now available for building efficient packages for engineering analysis create their own unique challenges, requiring a careful balance between functionality and usability.

In an effort to bring well known and dependable electrical network modeling approaches into the 21st century using modern software techniques, simplified user interfaces, and relatively seamless access to multiple traction power data sources, a comprehensive software package has been developed by Gannett Fleming Transit & Rail Systems that meets these requirements while remaining true to the goals of strict quality and ease of use. The program can be utilized to analyze both dc and ac electrical networks, including high voltage supply systems that power the traction systems. The program is also capable of utilizing power demand data from numerous sources, makes use of parallel processing when able, and provides extensive reporting capabilities.

GENERAL OVERVIEW

The program employs a front end that is generally represented in logical user interface panels and tabs. These include interfaces for model information, network management, solution control, and report generation. In addition, a comprehensive Excel™ spreadsheet template was developed, allowing an entire model to be represented in the spreadsheet and loaded into the program in a single step.

Considerable effort has also been focused on ensuring that the program addresses the requirements of traction power analysis. For dc systems, this means that inverter substations and energy storage devices are supported by the simulator. For ac systems, such components as autotransformers, booster transformers and harmonic filters are included on the traction power side; grid transformers, distribution feeders and power generation stations are included on the high voltage ac supply side.

Behind the scenes lies an extensive library of utilities that provide easy access to features such as units conversion and formatting options, data management, default handling, and error handling. These modules work to ensure that the program is intuitive, easy to use, and provides accurate results.

COMPONENT REQUIREMENTS

In order to effectively model modern traction power and supply networks, all of the components found in typical electric networks need to be included and modeled correctly. This includes power distribution elements (transformers, grounding connectors, sectionalization switches, parallel wires, bonding, etc.) as well as power supply and power consumption items (substations, trains, and stationary loads), and must ensure that the required electrical properties appropriate for the conductor network (shunt admittance, series impedance, mutual impedance, etc.) are handled properly.

Electrical Network Elements

The backbone of any electrical system analysis is the conductor network that transmits the required energy. The program utilizes components that explicitly represent any conductor or bus in the network. Each component maintains its required parameters, such as self-impedance, mutual impedance and shunt admittance (for ac conductors), resistance and leakage resistance (for dc conductors), and voltage for bus elements. In addition, the connectivity of the network is maintained through the accurate connectivity of conductors and buses. Conductors and conductor components are identified in *Table 1* below:

Electrical Network Elements	
OCS/Third Rail	Parallel Wires (Passive)
Running Rail	Substation Feeders
Negative Return	Supply Cables
Parallel Feeders	Jumpers
Phase Breaks (ac)	Sectionalization Gaps

Table 1

(Note- Parallel wires (passive) are used to represent conductors that are subject to conductive and inductive coupling with the traction power network. Under normal condition, jumper wires are used to model track cross-bonds or connections between paired conductors at the same voltage level. In addition, they can also be used to model short circuits between paired conductors at different voltage levels under fault conditions.)

DC Traction Power Components.

The program can model most components found in a typical dc traction power system. These components are detailed in *Table 2* below:

DC System Components	
Rectifier Substations	Battery Substations
Inverter Substations	Automatic Grounding Switches
Parallel Stations	Trains
Energy Storage Devices	Stationary Loads

Table 2

(Note- Stationary loads are used to represent loads that do not move but can vary with time, such as switch heater loads and yard loads.)

AC Traction Power Components

Similar to the dc side of the network analysis, the program provides for a comprehensive list of viable ac components to allow accurate modeling of an ac network. *Table 3* below lists the components available for ac modeling:

AC System Components	
Simple Substation	Booster Transformers
Parallel Stations	Trains
Autotransformers	Stationary Loads

Table 3

High Voltage Components

To complete the interaction between the traction power network and the high voltage network, the program includes a module that handles the high voltage supply network. The components of the high voltage network include:

High Voltage Components	
Power Generation	Transmission Cables
Buses	Loads

Table 4

Metering Components

Rounding out the list of components to include in the model are the various electric meters, allowing for accurate measurement of voltages and currents throughout the model. *Table 5* lists the meters included in the program:

Meter Components	
Voltage Meter	Current Meter

Table 5

Meters can be inserted at chosen points or at regular distances for one or more chosen conductors.

USER INTERFACE REQUIREMENTS

One of the most important and often underappreciated aspects of any software package is the user interface and the ability to import data into the system efficiently and accurately. This covers a broad range of subjects ranging from importing custom data to managing the model. The program provides two distinct paths that allow efficient model generation through its user interface.

Spreadsheet Import

As a minimum requirement, the program utilizes a library that allows direct data importing from a custom Excel workbook. This workbook contains two primary worksheets (tabs) that provide the basis for the model: a worksheet containing the segment data for the track and a worksheet containing all of the required electrical components. The component data worksheet contains a number of macros that provide context-sensitive column headers and help comments.

In addition to the two primary worksheets, the workbook contains a number of support worksheets that allow the user to control units (both on a macro level and on an individual data column level), allow the user to specify any required curves (such as substation voltage-current curves or stationary load schedules), and allow the user to specify train attributes for specific trains in the model.

Graphic User Interface (GUI)

The program employs the latest Microsoft OS features, including a “ribbon” control and quick access toolbar found on many modern applications. This approach allows for a familiar user interface experience while, at the same time, allowing for a modular approach of the program’s features.

The GUI also has an extensive library behind the scenes that allows for program-wide and parameter-specific management of units and formatting. This ensures consistent GUI behavior and accurate handling of input data.

Data Integrity

In both approaches to data import (spreadsheet or GUI), the program maintains rigorous data validation rules to ensure the resulting model is complete and error free. Issues that may arise are brought to the attention of the user immediately via a common error handling dialog window.

NETWORK CONSTRUCTION & PARTITIONING

The program is designed to model the network components and component behavior as close to the real world as practical, while still retaining efficient processing. Each component specified in the input data is created in the underlying model. Then virtual “wires”, with their associated physical properties, are implemented to connect the various components.

From the beginning, the program was designed with a focus on efficient processing. Where possible, sections of the entire network that are electrically isolated are partitioned into multiple independent sub-networks. This allows for easier and more efficient solver management, enabling the program to take advantage of parallel or remote processing. It also allows for the handling of large

networks that may exceed the memory capacity of the user's hardware.

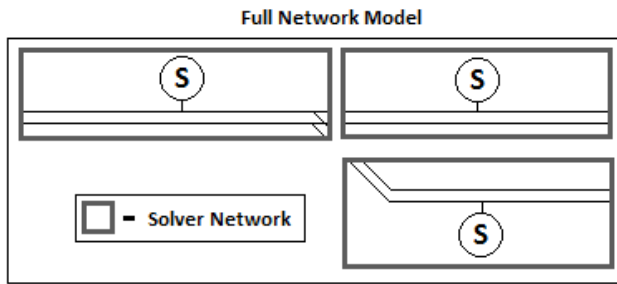


Diagram 1 - Solver Networks

SOLVING THE NETWORK

The computational core of the program contains the mathematical solver and solver controller modules. The various solver controller modules handle all of the necessary solution controls required for a successful simulation.

Train Handling

One of the most important responsibilities of the solver controllers is the ability to manage train performance information efficiently. Since the program can utilize TPC data from a number of different sources, the program will encounter different methods of handling TPC time steps, depending on the source of the data. In some cases, a fixed time step is reported in the TPC data, with the only exception being dwell times. In others, the time step in the TPC data is not fixed. The program handles these cases by interpolating the train data using the running time as the guiding parameter. By handling the TPC data in this manner, the program can ensure consistent results, while eliminating any TPC data that is outside the requested simulation timeframe.

In addition to the interpolation methods, the solver controllers implement a filtering algorithm that eliminates any TPC data that is geographically outside of a given solver controller. This ensures that processing time and memory are not wasted on nonessential computations.

Solution Control

Once the proper train and static load data have been applied to a given solver controller, the solution process is started. For models that are well behaved, the solution process is a relatively simple matter of solving the network and saving the results. However, in cases where

the model is not so well behaved, such as cases where the power demands of the trains and/or the stationary loads exceed the available power, the solver controller has more work to do.

Both the trains and the stationary loads maintain a set of voltage taper parameters that allow the adjustment of the train's or load's power in the event that the supplied voltage drops below the taper level (or above the taper level for trains that employ regeneration). These parameters are used to adjust all trains and loads that fall into the taper region. This allows the solver controller to find a viable solution using the given power supply capacity. In the event that the power demand of a train or a load is adjusted during a given time step, the results will reflect the adjusted power.

Managing Solver Controllers

The solver controllers for each time step are generated by the individual solver networks. To facilitate the most efficient use of the available processors, individual solver controllers dictate whether or not the next solver controller for the next time step can be generated. If a particular solver controller does not have a device that the next solver controller is dependent on (such as an energy storage device), then the solver controller notifies the solver network that the next solver controller can be generated. However, if a solver controller does contain a component that the next solver controller depends on, then the second solver controller will not be generated until the first one has completed.

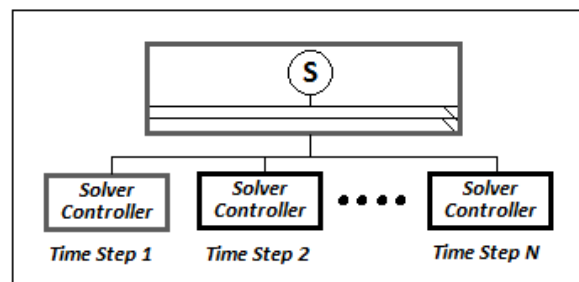


Diagram 2 - Solver Controller Management

Managing the Processors

Sitting on top of the solver controllers is the Process Queue Manager. The Process Queue Manager is responsible for shepherding the various solver controllers for all of the simulation time steps.

While the solver networks are responsible for generating a solver controller for each time step, the

Process Queue Manager handles the actual processing of the solver controllers. By handling all solver controllers (ac or dc) via a common manager, the use of the available processors can be maximized.

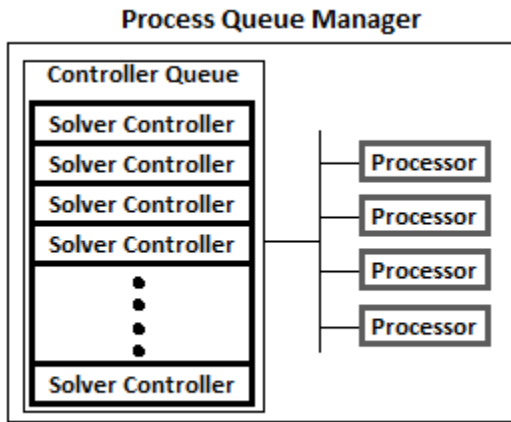


Diagram 3 - Process Queue Manager

Results Collection

When a solver controller finishes its computations, the results are posted to a common repository. This allows for generic processing of all results through a single point. All related data is assembled to generate a seamless representation of the results.

In addition to the pertinent results for the components in the model, the results also include the parameters used by the solver controllers, as well as any error and warning messages that may have been generated during the simulations.

POST PROCESSING

As with any good analysis software, post-processing results is a critical aspect of the program. Both text and graphic reporting are available. Each report type allows the user to specify what is displayed or plotted, and gives extensive control over the content of the report.

Each result set generated from a simulation is stored in a single file for subsequent retrieval. Any results set can be loaded into the report manager, allowing for comparison of results between simulations. In addition, the underlying model that generated the results is also included with the results set so that user can review any of the parameters associated with the results.

SUMMARY

The challenge of any engineering analysis software is to provide a tool that is comprehensive, accurate, and easy to use. These underlying features are at the core of the design of the program. Successful implementation of network power adjustments, parallel processing, and accurate data acquisition and handling has led to a robust and capable package that meets these requirements. The ability to model a broad spectrum of electric networks means that proper analysis can be accomplished effectively. Taking advantage of available technologies means that the analysis can be accomplished *efficiently*, as well.