REALTIME PREDICTIVE AND PRESCRIPTIVE ANALYTICS WITH REAL-TIME DATA AND SIMULATION

Hosni Adra

Simulation and Optimization Manager/Partner

CreateASoft, Inc.
3909 75th Street, Suite 105
Aurora, IL 60504, USA
www.createasoft.com

ABSTRACT

The past few years have experienced tremendous growth in the use of simulation to improve the work place and the efficiency of every operation. Although processor speeds have increased at a very fast pace, simulation, due to its extensive computational and visualization requirements, have consistently challenged and used the processors to their full power. Moreover, with the evolving 64bit simulation engine technology, simulation models have increased in size and complexity with extensive memory requirements, ever expanded data input sets, along with increased connectivity requirements. In addition, real-time data is becoming more accessible than ever, and it has greatly contributed to simulation models accuracy, validity and usability. This paper discusses the reusability and extensibility of simulation models and their roles in predictive and prescriptive analytics using real time data connectivity as used in SimTrack real time predictive analytics and schedule adherence.

1   INTRODUCTION

Large simulation models, spanning full warehouses, manufacturing plants, hospital campuses, and other types of environments are complex, computationally intensive, and require more time to run using traditional single threaded simulation environments. Take for example a warehousing implementation that involves multiple AS/RS systems, standard racking, AGVs for material handling, automation equipment, high speed lines, and, most of all, human workers consistently interacting with all aspects of the warehouse. To drive such a system, data must be available from the WMS (Warehouse management system), WCS, PLCs, RFID and other data points tracked through either WMS, SAP, barcode, RFID, RTLS, or manual entry. For offline runs, waiting for the model to complete the run is not as critical or demanding. On the other hand, when the model is providing real time predictive analytics, model run turn-around time, including multiple scenarios and Monte Carlo runs, become critical. Whether the implementation is for schedule adherence, predictive bottleneck detection, or other efficiency and improvement alerts, two key factors impact the overall solution; simulation speed, and model validity and accuracy.

Simulation speeds can be improved by better utilizing the underlying hardware system. Whether smaller, independent and interacting models are used or a highly distributed single model implementation is used key challenges arise in model synchronization and real-time data availability. As discussed later, there are many methods that can be implemented to successfully distribute the model and achieve extensive improvement in simulation run speed without disturbing model validity.

Model accuracy, on the other hand, can be even more challenging, especially when real time data is used. In the case of static environments, the modeler is aware of the different process steps and movements that parts or resources are taking. Model constraints can be imposed using the known data and current constraints. In the case of real-time data and real-time predictive environments, models need to be designed in a way that allows them to grow with the system, enable real-time constraint changes, and in some case, expand themselves in order to maintain the correct relationship with live data. In other words, models need to run within an intelligent environment that allows them to modify their constraints or process steps based on data feedback from external data sources and without human intervention. This is a key requirement that enables systems to be used as turn-key, hands-off systems that are constantly evolving within their environment while maintaining a high degree of validity and accuracy.

2   CHALLENGES

The presented technology, although evolving, has successfully been implemented and has overcome many challenges across multiple industries. When simulation models interact, they need to be /intelligently synchronized in order to protect the integrity of the results and validity of the model. Whether the implementation spans a single system using multithreading methods, or multiple systems using inter processors synchronization, the synchronization is one key item that must be controlled.

In addition to the synchronization challenge, simulations interacting with external, real-time data systems, must maintain the integrity of the integration with outside systems. In other words, real-time data needs to be clearly mapped to the model in a way that prevents ambiguity while providing an efficient redundancy level. Whether the data is being retrieved from external data stores (Databases, SAP, WMS, ERP, EMR,…) or received asynchronously through PLCs or RFID data, the simulation model(s) automatically update and the outcomes of the changing conditions are used to identify bottleneck and areas for improvement in real-time.

The most challenging aspect of intelligent models is their ability to constantly evolve based on the real-time environment without disturbing the validity of the model. Such models are developed in a way to adapt to incoming data and add or remove constraints based on the actual environment. Intelligent models consistently learn from the current environment, readjust on-the-fly and consistently perform validation, analysis, and optimization in order to provide a hands free environment.

Examples of real-time model interaction and learning intelligent models are presented.

2.1   Model Building

There are three methods in which to classify the models as interconnected network of simulation models; model(s) output(s) feeding one or many model inputs, tightly synchronized models in a multithreading environment, or distributed processing with synchronized models across the overall system.

When real time data connectivity are used, the methodology is not as critical as the modeling approach. When real time data and controls are used (such as RFID, GPS, and other RTLS time systems), simulation models should be built and defined to allow for the external tracking system to automatically update and modify the model behavior and constraints. In other words, if a fork truck item has been identified to be transitioning to a new pallet location the model should be smart enough to allow the movement and automatically update the model constraints with the new pallet location.

The same set of models will be used in multiple ways ;

–   Real time visibility showing the location of each item, progress and analytics for every entity within the system.

- Predictive mode where the model constantly runs in the background, consuming historical data, processing real time constraints, and generating predictive analytics while providing alerts and notifications as to the impact on the future state of the operation.

- Prescriptive mode where the model optimizes the future state with processing changes required to improve the efficiency of the operations' predicted future-state. Whether rescheduling jobs, labor, deliveries, or re-sequencing the line with updated WCS or PLC logic, or modifying process flow and processing constraints a prescriptive mode provides for an optimized future state with minimal human interaction and no offline analysis required.

2.2   An Intelligent model with Real time connectivity

Whether the models are connected to an HL7 data feed, WMS, WCS, ERP, or RTLS systems, each one of them needs to be able to modify itself in order to support the data feed. Self-Modifying models are able to detect changes in the external environment and constantly evolve in order to support the new constraints. An example of an evolving model would be adding new rooms to an ED department, changing the pick path in a warehouse, adding a new rack, inserting a new machine into the flow, or generating a new assembly type.

Dynamic Self-Evolving simulation models are built using Simcad Process Simulator, a dynamic environment where models automatically update and modify their constraints and flow during the simulation run. In addition, since Simcad Pro allows models to auto-recompute the distributions based on historical data, it is used as the underlying simulation engine to drive the Monte Carlo analysis and is incorporated in both predictive and prescriptive environments.

Extending the models used in offline analysis, SMARTLS and SimTrack provide the connectivity to real time tracking systems, or multiple data systems as needed. In addition, SimTrack includes a PLC connectivity module that enables the model to connect to real time PLC information. The PLC connector can be setup as a dual channel connector allowing the SimTrack environment to read and write to the PLCs.

3   SYSTEM GENERATED RESULTS

As the system evolves and models interact, analytics and performance tracking metrics are dynamically generated. The generated analytics and tracking metrics are then used for tracking and visibility, predictive and prescriptive analytics.

3.1   Real-Time Visibility

- This model view provides real-time visibility to the tracked environment. Each entity (equipment, product, people, resources, others) are displayed in a visual representation and are equipped with a hover feature or reporting dashboard that enables a user to check the status and properties of each entity within the system.

- Model metrics are computed in real-time and displayed to the user via web enabled dashboards based on current model constraints.

- System tracking, visibility, and metrics are published to a web interface in order to provide users access to the interface and display the information on desktop and mobile devices. System visibility representation is presented in either 2D or 3D space depending on the underlying hardware infrastructure and visualization requirements.

3.2 Predictive Analytics

As described in this document, the model has the ability to learn from current environment and evolve itself in order to better improve its predictive values.

- o The model starts out initially based on the initial constraints defined within its settings. This is considered the system startup, and occurs once per model per installation. This initial runs serve 2 main purposes
  - Generate the initial current model state
  - Collecting external constraints that are influencing the model progress, including downtime, cycle times, arrival patterns and others.

- o As the SimTrack predictive model evolves, the following operations occur
  - General distributions of cycle times, arrival patterns, downtime, and other factors are computed at each run. Historical distributions are computed based on change factors and patterns that relates to the validity of the data. As an example, far history are weighed less then near history, and current daily events have more impact than either of the historical data sets. Known datasets containing near-future schedule or shipping requirements also impact the generated analytics.

  - Multiple Monte Carlo runs are performed in order to avoid the variability created from the distribution and to establish a valid and accurate representation of the future of the operation.

  - Data generated from the predictive model is displayed directly to web enabled dashboards providing great insight on the current and future status of the operation. The predictive environment generated is highly accurate as it is derived from real-time constraints driven by a distribution of current, historical and known future data sets.

- o Real-Time **schedule adherence** is generated based on the current status and predicted future of each operation. Schedule deviation, slippage, and on-time deliveries are dynamically computed and displayed to the user. Events and alerts are generated when unexpected events occur or schedule slippage is detected. The predictive model is able to determine the amount of generated delays, their cause, and potential impact on other sections of the operation.

3.3 Prescriptive Analytics

- o In the prescriptive model mode, real-time optimization and validation is performed in the background. For each potential optimization method, and similar to the predictive environment, a set of Monte Carlo simulation is run in order to determine the viability of the solution. The number of required runs is determined by the system and is driven by model convergence factors as they relate to the optimized values.

  Examples of real time optimization include identifying the best pick route and pick schedule for a warehouse, modifying put-away and replenishment sequences, modifying schedules, man-power requirements or maintenance schedules in a manufacturing

environment, or re-scheduling OR patients based on OR load levels, delays and current state impact.

- o During the real-time optimization, a number of potential improvements could be determined. Each viable solution requires a number of changes to the real system in order for the predicted results to be valid. The system can react to the optimization suggestion in different ways depending on its configuration.

  - Accept the requested change that fit within the "allowed" change constraints, propagate the change to external system, and proceed with the tracking.

  - Present the user with a number of options and allow the decision to be made by the manager in charge. When a solution is accepted, the system can either push the change to external systems, or monitor the environment for change. In this case, it is the manager's responsibility to effect the change on the external system

- o Another key benefit of prescriptive analytics is its impact on schedule adherence. As different optimization options are selected, an insight into the future of on-time delivery is presented along with potential impact of change from the current state. In other words, each optimization method impacts the on-time delivery and efficiency of the environment. Those changes, delays or improvements, are posted on live dashboards with varying degree of detail depending on the dashboard audience.

4   MULTIPLE COLLABORATION MODELS

For the system to be effective, the implementation should be divided into multiple interconnected models working together in order to provide the final results. The models can run in 3 different modes;

Synchronized mode, where multiple models are synchronized in time, constraints and behavior in order to provide a faster execution state in a more controlled environment. There is no limit on the number of synchronized models that are run, and multiple hardware platforms may be used in order to speed up the execution.

Sequential mode, where multiple models build on each other results. When the first model completes, it creates the input data set for the next model to run. Upon model start, the second model reloads its constraints, auto computes its distribution and proceeds with the execution. Note that in a sequential mode, a model run may consists of multiple Monte Carlo runs working together to generate the input to the next model.

The third mode is a hybrid implementation of sequential and synchronized modes. Based on the system requirements and result definition, the model execution can switch from synchronized to sequential multiple times during the execution cycle.

In all modes, the underlying scheduler in SimTrack takes care of all required synchronization and sequential execution as defined in the system properties. The burden of synchronizing the model is now a standard functionality of the controlling system hence reducing the requirement on the model developer to implement more stringent synchronization methods. The resulting environment is scalable in execution speed, model complexity and overall system accuracy.

5   SUMMARY

The presented environment is designed to provide a complete predictive and prescriptive environment built on dynamic, real-time changing, simulation models. The presented SimTrack is designed to take simulation

environment to the next level and make it an integral part of any organization toolset. The generated data set, whether in real-time visibility, predictive or prescriptive environment, enables organizations to utilize the power of simulation in order to maximize the operational effectiveness and improve the on time delivery and scheduling. Existing systems implemented in healthcare, manufacturing, and warehousing have proven to be indispensable and have generated an ROI that far exceeds the initial implementation cost within the first year of the implementation.

AUTHOR BIOGRAPHIES

HOSNI ADRA is the co-founder of CreateASoft, Inc. He has been involved in Process Improvement and Simulation for the past 25 years. Hosni has applied his process improvement expertise to multiple industries including healthcare to increase efficiency and reduce operating risk. As the holder of several patents in the fields of dynamic simulation and tracking, Hosni has been a sought after expert in these fields and has presented multiple papers on process improvement using simulation and implementing lean concepts. With his dedication to the use of technology to improve efficiency and output, he has positioned CreateASoft as a leader in the process improvement industry. His email address is hadra@createasoft.com.

Simcad Pro®, SimTrack®, CreateASoft®, and Dynamic Simulation® are registered trademarks of CreateASoft, Inc.