PortSim – A Port Security Simulation and Visualization Tool[1]

Daniel B. Koch, Ph.D.
Senior Member, IEEE
Oak Ridge National Laboratory
P.O. Box 2008
Oak Ridge, TN 37831-6086
USA

Abstract – Around the world, there is great concern about the movement of threat materials using seaport shipping containers. The benefit of early detection of weapons of mass destruction is obvious. However, the inspection process needs to be conducted in such a way as to not unreasonably impede normal commerce. Prior to actual deployment of new detection systems, policies, or procedures, it is useful to construct an operational and cost model of the port facility and to run simulations to gage the impact. Using a simulation model beforehand aids decision makers in evaluating trade-offs. *PortSim* was developed at ORNL by the author to allow a user to investigate a number of parameters in order to see the impact on port operations and cost. It consolidates a conceptual operations model, cost information, policy and procedures database, a real-time data acquisition capability, and information flow tracking and provides a visualization of port operations in a geospatial environment. This paper describes the use of *PortSim* to simulate and visualize a typical port.

Index Terms – Port Security, Container Inspection, Visual Simulation, Operations Research, Enterprise Modeling

I. INTRODUCTION

Operations research is a time-honored discipline where simulation is used to make cost-benefit decisions in a number of different fields. Discrete-time, stochastic simulation techniques, coupled with graphical visualization of the results, enable exploring trade-offs between different investment and management strategies in ways understood even by the non-specialist. This paper describes an operational and cost modeling software tool, the Oak Ridge National Laboratory (ORNL) Port Simulator, referred to simply as *PortSim* throughout, that was developed for the U.S. Department of Homeland Security (DHS) to investigate new container inspection procedures at seaports. Some background information is presented first followed by a discussion of the overall software design. Then a description of the program operation is presented to illustrate its use in conducting trade-off studies. The current tool is a work in progress that will be considerably enhanced as time goes on. These enhancements are presented as future efforts in the conclusions.

II. PROJECT DESCRIPTION

Port security is an important element of the DHS mission to improve the overall U.S. national security posture. The U.S. has over 150 ports situated on its coasts and major river ways [1]. In 2006, containerized cargo entering the U.S. exceeded 27 million TEUs (twenty-foot equivalent units) [2]. DHS Customs and Border Protection (CBP) has been tasked with inspecting the contents of all incoming containers [3]. This must be done in such a way as to not hinder normal commerce. The challenge is enormous. A number of DHS sponsored projects exist around the country in which various sensors and inspection methodologies are being tested. Prior

1Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285, managed by UT-Battelle, LLC for the U. S. Department of Energy under contract no. DEAC05-00OR22725.

to actually conducting field trials and possible deployment, simulation and visualization tools enable the identification of promising solutions. *PortSim* was developed to assist in this effort.

III. SOFTWARE DESIGN

The purpose of the software tool is to allow the assessment of the operational and cost impact of introducing upstream monitoring of cargo containers at a port facility. The questions to be answered largely consisted of two: what changes in normal port operations would be necessitated by the introduction of upstream monitoring; and what are the costs associated with these changes as well as the costs of additional equipment and personnel required. In developing the program, an incremental cost approach was taken. That is, it was not necessary to capture the existing baseline cost structure of the port in its entirety, especially since much of this information does not impact operational costs and is proprietary to the port operating company; it is only necessary to calculate the additional costs of changes in port operations, equipment, and personnel to accomplish the proposed upstream monitoring. This was the guiding principle behind the software system design.

It was recognized early on that a cost analysis independent of the operational aspects of the port could not be undertaken. Although there are static, fixed costs associated with upstream monitoring equipment, other costs are intertwined with the movement of containers as they are offloaded from a ship, moved to various locations in the yard, and prepared for delivery. All of these movements are potentially impacted by the introduction of upstream monitoring. Since this is the case, a combined operational and cost modeling tool was developed.

A further requirement was that the software tool itself would be a project deliverable to facilitate the analysis of additional simulation modes that may be proposed in the future. To accomplish this requirement, a custom, cross-platform, graphical-based application was written using the open-source, object-oriented language Python [4] and an associated graphical user interface (GUI) toolkit called wxPython [5]. Similar to Java, Python utilizes a byte-code interpreter but is often much less verbose due to its simple syntax and has a host of scientific and visualization modules. Using open-source software instead of proprietary commercial software often times simplifies distribution and allows users to modify the code to better fit their needs.

IV. PROGRAM OPERATION

The software tool consists of a simulation engine and a GUI overlay. In addition to the normal window items such as menu, toolbar, and status bar, a number of features have been incorporated to help the user successfully navigate the tool. A tabbed interface allows easy access to data values. Labels and fields are also highlighted corresponding to a particular simulation mode. The tool may be used without visualization for fast execution. Processing 1,000 containers, for example, requires approximately two minutes of real time on a typical 2 GHz desktop workstation representing over eleven hours of simulation time. A 2D visualization is available for those cases where it is helpful to see operational details. Using this tool, numerous parametric studies may be undertaken.

When the application is launched, the main Simulation Control Panel, shown in Fig. 1, is presented to the user. This is where most of the interaction occurs. A pull-down list allows the selection of a particular simulation mode. Upon selection, pertinent data fields are highlighted while unrelated fields are dimmed. The number of containers to be used in the simulation may be specified along with the necessary equipment items such as cranes, yard hustlers (i.e., trucks), pickers (e.g., top-picks, side-picks), etc. A simulation may be run with or without inspection so that the impact of upstream inspection may be compared to baseline operations and costs. The simulation run time and time increment may also be specified. Different time values are useful for viewing operations faster than real time or determining the amount of time to completely offload a ship for example.

As the simulation runs, a progress bar shows the approximate time remaining until completion. The output fields are updated incrementally with the results. On the right side of the panel are the operating costs associated with the simulation mode. Some costs are computed on an hourly basis, such as those corresponding to equipment. Other costs are assessed on a per container basis such as the ship unloading charge. Costs are totaled for the port operator and shipper. In the center of the screen are various statistics. The number of simulated alarms for each type of detector is shown along with the average container travel time. The latter is the primary measure of the impact on operations due to upstream

monitoring. This statistic will be discussed more in the example to follow.

Fig. 2 shows the Container Panel where a number of settings may be specified by the user. The radiation levels of the containers may be randomly assigned using a uniform distribution between the lower and upper limits. Containers may also be grouped to show the impact of sequentially unloading a string of similar containers on the inspection process. Other characteristics and costs may also be specified as shown.

Once a simulation has run, the container details field shows a number of elements of interest. For each container, a complete history of the path taken is recorded. Both the equipment item as well as the time spent at each stage of an operation may be viewed for each container. This allows the user to look for delays in the operations. Also recorded are flags showing whether the container caused an alarm at either the primary or secondary inspection site.

A number of equipment panels are available for setting various operational parameters. An example is shown in Fig. 3 for the cranes which offload containers from ships. Both the transfer time and the return time are uniformly distributed between the limits shown. An hourly cost as well as a per operation cost field are available. The other equipment panels are similar.

A number of details are available once the simulation has run. If an operation, such as offloading a ship, has not been completed, the current mode of the equipment item will be shown along with the container number it is currently handling, if any. Some flags have more meaning for other pieces of equipment such as the Cleared flag. This is used to indicate whether a yard hustler or truck has cleared secondary inspection. The Delay flag is also useful to see if the equipment was ever delayed in handing off a container to the next processing station. This can indicate a bottleneck in the process chain. Finally the number of containers processed and the associated costs are shown for each piece of equipment.

Similar to the equipment panels are the detector panels. The settings for the primary and secondary detectors are identical. The panel for the primary detector is shown in Fig. 4. The device type may be entered by the user using whatever designation is meaningful. The threshold setting may be adjusted in conjunction with the container radiation levels to simulate any desired alarm level. The scan and reset times, if pertinent, may also be specified as well as an hourly cost of operation and a per scan cost to account for any consumables required to perform a scan. Similar to the equipment panels, a number of data items are recorded during a simulation and displayed in the details field. The number of scans performed and the number of alarms generated are probably the most interesting.

The Flow Panel is a non-graphical display of container locations during the simulation. It consists of a number of lists which display container numbers. Fig. 5 shows the results of an offload operation in progress where some containers are still on the ship, some have progressed to the yard, while the remaining ones are in various stages of transport by the cranes, yard hustlers, and pickers. Containers being scanned are also indicated. This panel is most useful during the Single Step and Time Step run modes.

Finally, Fig. 6 shows the Route Panel. This interface serves two purposes. First, it provides a way for the user to specify the routes that the yard hustlers take in the port as containers are offloaded from the ship and stacked in the yard. Second, it presents a 2D visualization of the simulation as it runs. Although only one gang is shown here for clarity, multiple gangs may be specified since this is the usual case in practice. The visualization allows the user to qualitatively determine traffic bottlenecks and resource deficiencies, among other things, in the port.

There are currently three different ways to run a simulation. The fastest method, useful for most cases, is to use the Run Simulation command from the File menu or its corresponding toolbar button. A number of sequential runs may be specified with the results averaged to improve statistical properties. Two additional run modes, Single Step and Time Step, are also available under the Tools menu or by pressing the corresponding toolbar buttons. These are meant to be used in conjunction with the Flow Panel or Route Panel so that containers may be visually tracked during the simulation. The time step and run time parameters on the Simulation Control Panel may be adjusted to run faster than real time in order to see some progression. These modes are useful for investigating delays and bottlenecks during a simulation.

Once a simulation has completed, the user has three options for saving results. Using the Save Results command under the File menu or its corresponding toolbar button results in an annotated text file containing all of the panel settings and results. This format is useful for inserting into reports using a word processor. The Save as Spreadsheet command,

also under the File menu and on the toolbar, saves the simulation results as comma separated values (CSV file format) which may be opened by a spreadsheet program. Text column labels identify the data items. Several runs may be grouped together using the Append to Spreadsheet menu command or toolbar button.

Figs. 7 and 8 illustrate graphs produced by PortSim for two important performance criteria as a function of the expected container alarm rate. For typical parameter values, alarm rates below approximately 5% indicate no significant delays compared to no inspections (0% alarm rate). Above this threshold, delays begin to increase steeply. Although space does not allow a full description of the reasons to be discussed here, the cause is a result of the long secondary inspection process that is needed after a container alarms at the primary detector. Yard hustlers line up waiting to be cleared so that fewer are available to unload the ship. Knowing this, the port operator can allocate more resources (yard hustlers) or more secondary inspection stations. This simple example illustrates the utility of performing simulation prior to deployment.

V. CONCLUSION

PortSim currently enables the investigation of the impact of container inspection on current port operations and the calculation of associated costs. With some additional development, the tool could be extended to simulate regional or national multi-modal transportation systems. Due to the object-oriented design of the tool, it is possible to modify the software to run on a computer cluster. This would allow the simulation of several ports simultaneously along with their interactions. In fact, entire multi-modal transportation systems could be simulated in this manner. Investigations into the most cost-effective methods for inspections could then be made on a regional or national level using high performance computing facilities. In addition, real-time data, such as ship manifest records, could be incorporated into the simulation to enable manpower scheduling by CBP in a just-in-time manner. These are just two possibilities afforded by the tool design.

VI. ACKNOWLEDGEMENTS

The author would like to acknowledge the suggestions incorporated into the software tool described in this paper by ORNL colleagues Dr. Budhendra Bhaduri, Erika McNair, Mark Tuttle, and James White.

VII. REFERENCES

[1] Wikipedia, “List of ports in the United States,” Jun. 2007; http://en.wikipedia.org/wiki/List_of_ports_in_the_United_States.

[2] U.S. Department of Transportation Maritime Administration, “U.S. Waterborne Container Trade by U.S. Custom Ports, 1997-2006,” May 2006; http://www.marad.dot.gov/MARAD_statistics/2005%20STATISTICS/Container%20Custom%20Ports,%201997-2006.xls.

[3] U.S. Government Printing Office, “Security and Accountability For Every Port Act of 2006,” Jul. 2007; http://frwebgate.access.gpo.gov/cgi-bin/getdoc.cgi?dbname=109_cong_public_laws&docid=f:publ347.109.

[4] Python Software Foundation, “Python Programming Language -- Official Website,” Jul. 2007; http://python.org/.

[5] Open Source Applications Foundation, “wxPython,” Jul. 2007; http://www.wxpython.org/.

VIII. VITA

Dr. Koch joined Oak Ridge National Laboratory in 2005 after spending 10 years in industry and 18 years in academia. From 1978 to 1987, he worked as a systems engineer in the aerospace industry for McDonnell Aircraft Company, Emerson Electric Company, and STG Electrosystems, Inc., all located in St. Louis, MO. From 1987 to 2005, Dr. Koch served as a tenured associate professor at the University of Tennessee in Knoxville, TN where he initially founded and led the Electrical Engineering Department *Communications, Information, and Signal Processing Group*. Later he founded and directed the campus-wide *Applied Visualization Center* where research activities primarily focused on creating 3D visual simulations. Academic duties included teaching graduate courses in analog and digital communications, information theory, and digital signal processing. His current efforts are focused on creating a software framework for combining discrete-time simulation with geospatial data in virtual environments for homeland security applications.

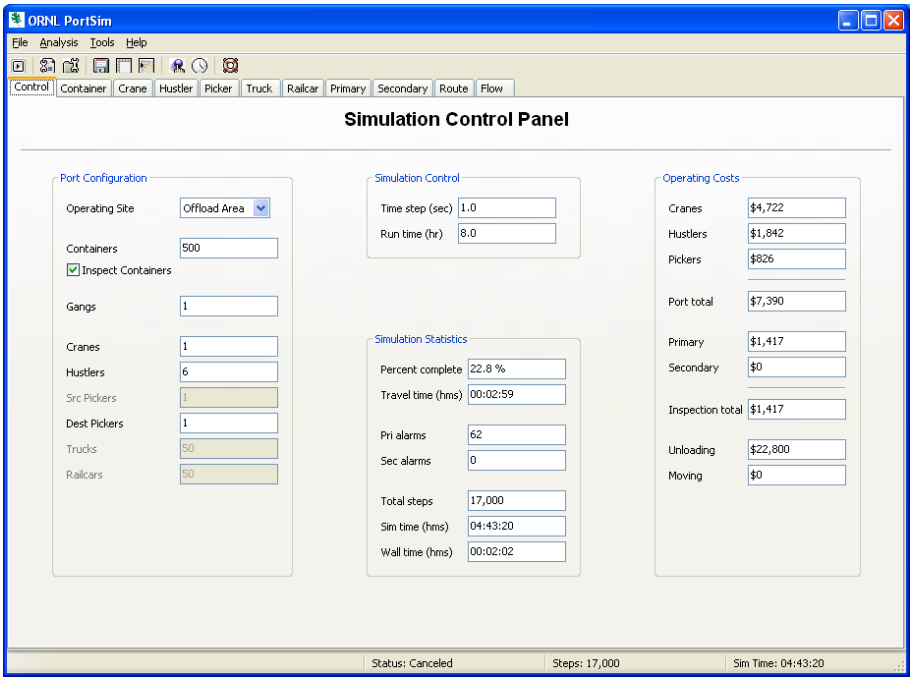

Fig. 1. Simulation Control Panel

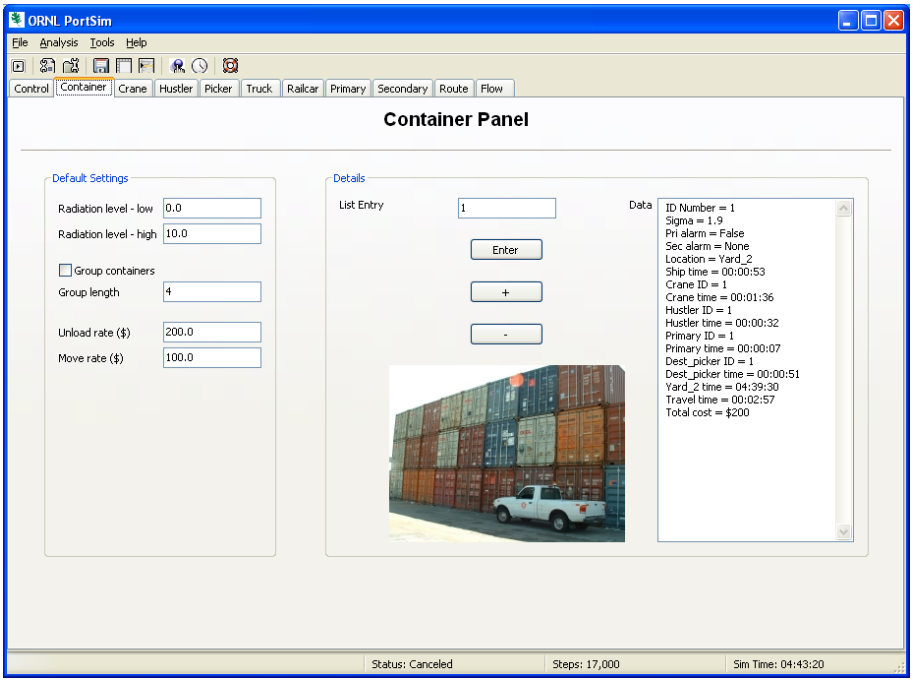

Fig. 2. Container Panel

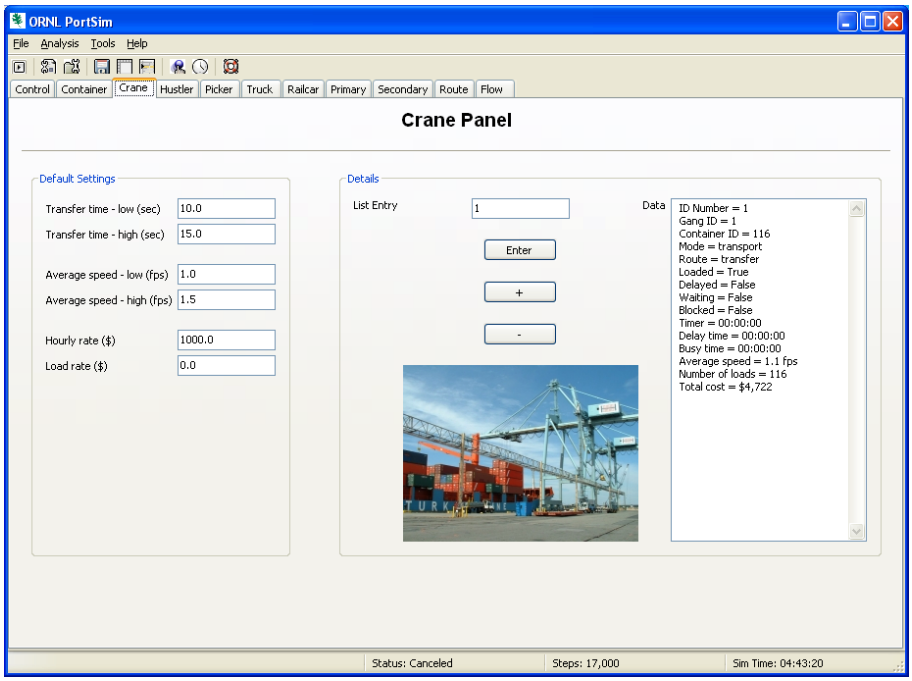

Fig. 3. Crane Panel

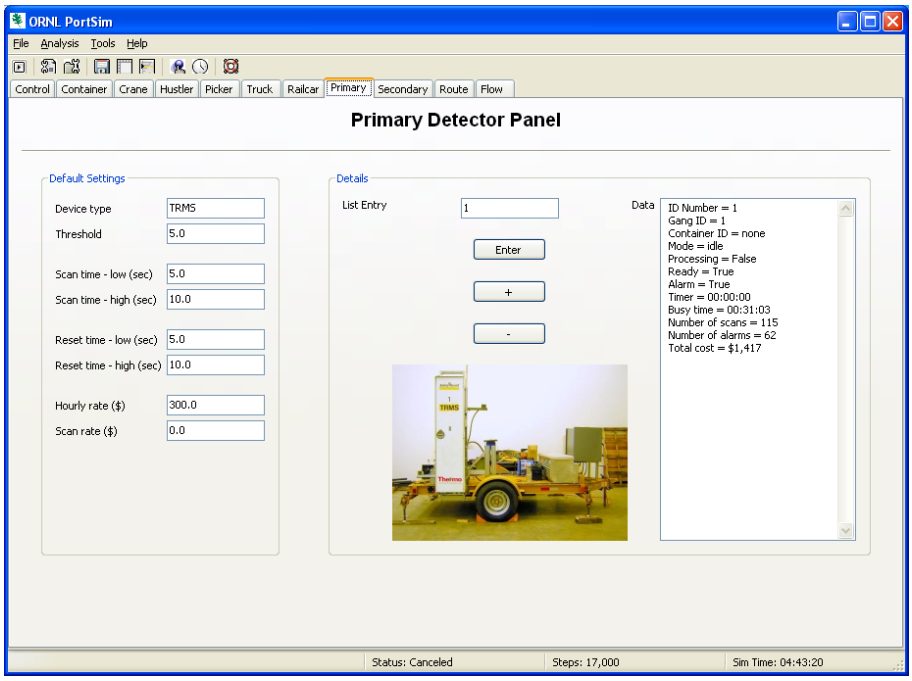

Fig. 4. Primary Detector Panel

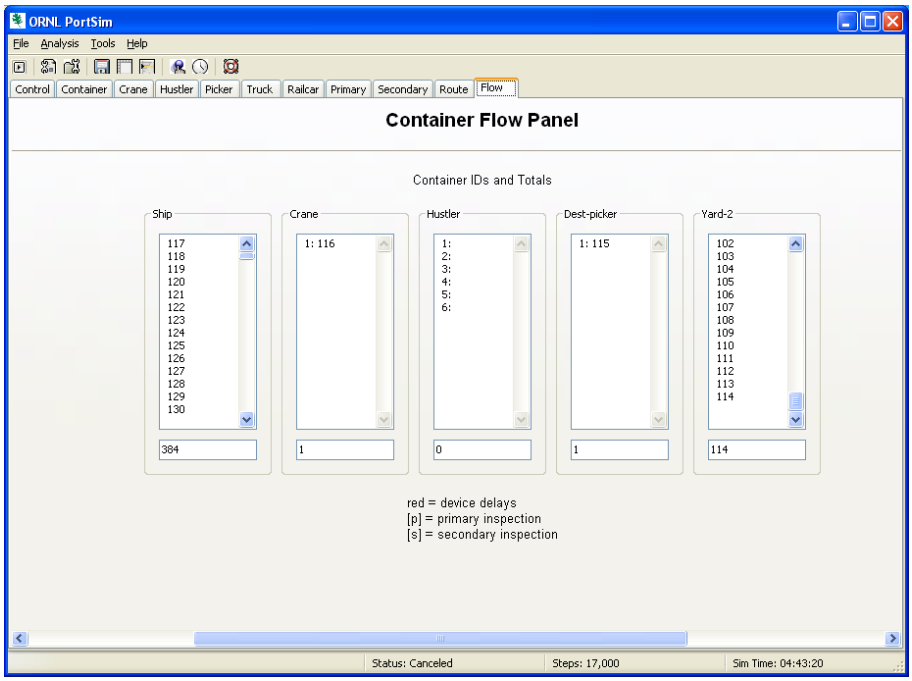

Fig. 5. Container Flow Panel

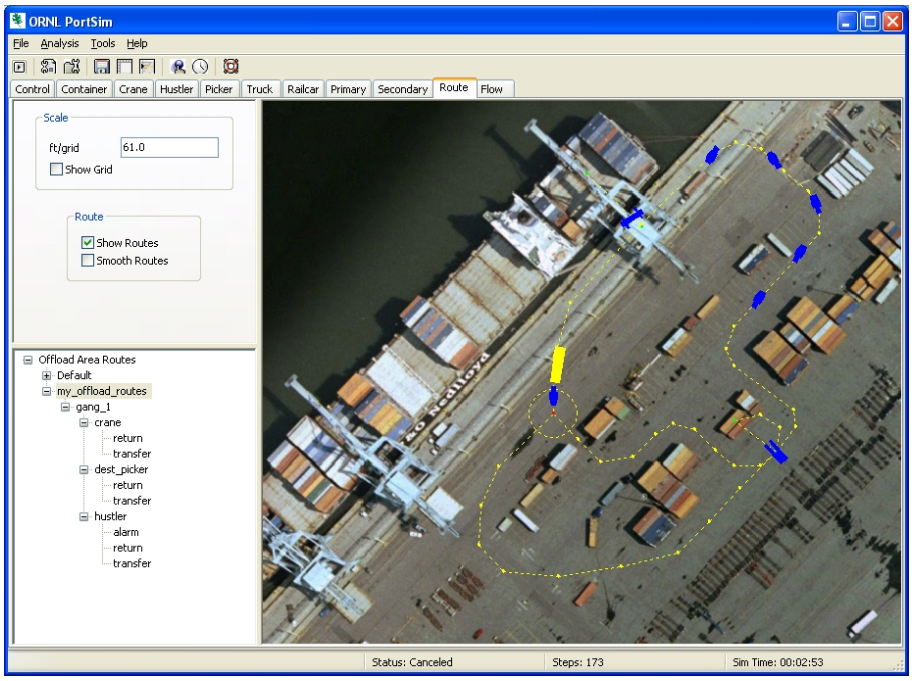

Fig. 6. Route Panel

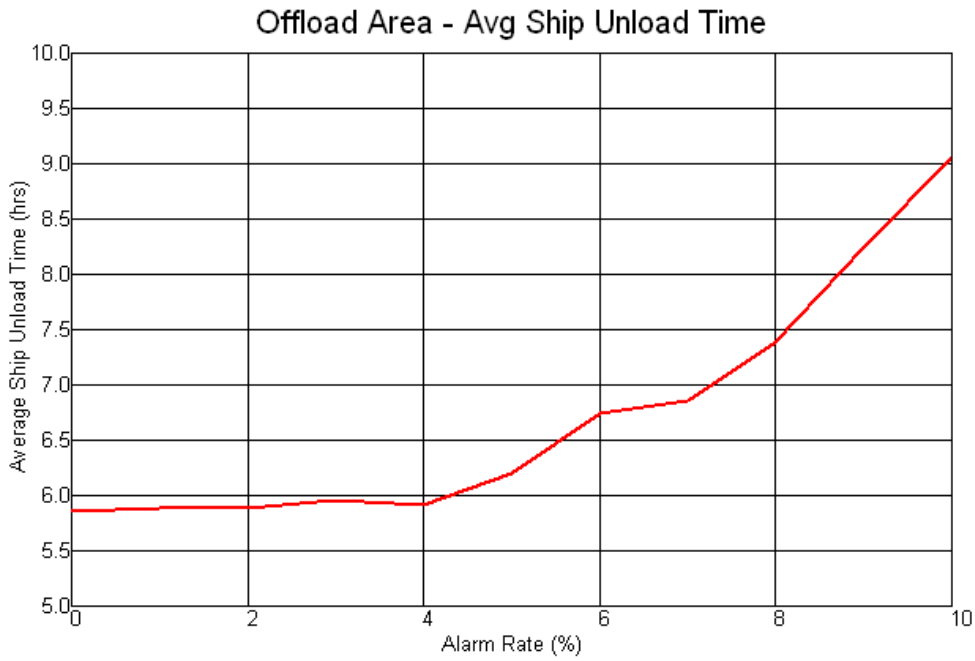

Fig. 7. Ship Unload Time vs. Alarm Rate

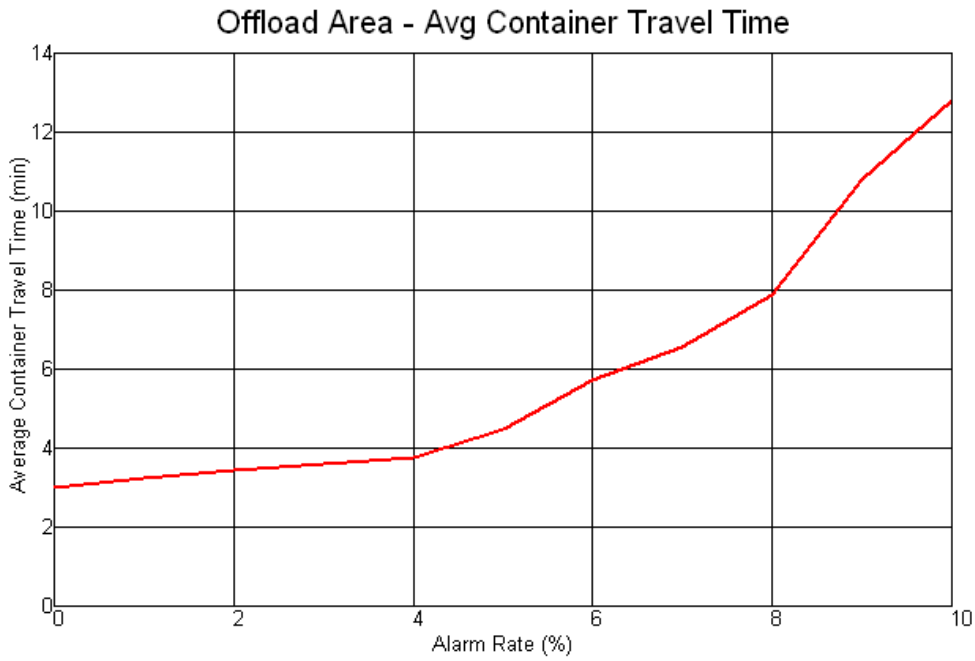

Fig. 8. Container Travel Time vs. Alarm Rate