NASS: SYSTEM SIMULATION OF INLAND WATERWAYS

Keith D. Hofseth[(a)], Cory M. Rogers[(b)]

[(a)] U.S. Army Corps of Engineers Institute for Water Resources, Alexandria VA, USA
[(b)] Camp Dresser & McKee Inc., Carbondale IL, USA

[(a)]keith.d.hofseth@usace.army.mil, [(b)]rogerscm@cdm.com

ABSTRACT
The Navigation System Simulation (NaSS) suite of tools has been developed by the Institute for Water Resources of the U.S. Army Corps of Engineers (Corps), as part of the Navigation Technologies Research Program. NaSS is composed of two primary applications, a Monte Carlo simulation model of vessel movements on an inland waterway system and a data processing and analysis tool for mining of historical data. The NaSS tools are used in concert for economic analyses of an inland waterway system. The NaSS suite was designed to answer potential questions of a waterway system under examination, for example; What is the overall system performance of a waterway network under different operating, demand load and reliability conditions?; How effective are alternative lockage polices at reducing delays and delay costs?; How does any single lock improvement project affect delays at other locks? This paper focuses on the Monte Carlo waterway simulation component called BasinSym.

Keywords: Monte Carlo simulation, waterway transportation, engineering reliability, queuing analysis.

1. INTRODUCTION
The United States economy is serviced by a vast network of 12,000 miles of navigable “inland waterways”. These waterways take the form of rivers and canals that connect the inland States to each other and to international commerce moving through coastal ports. Approximately 69 billion gallons of petroleum, 20% of U.S. coal, and 60% of grain exports travel along the inland waterways on a fleet of over 4,000 tug and towboats and 27,000 barges. Should this network of rivers and waterways be inaccessible to commercial traffic, 60 semi-trucks or 15 rail cars would be needed to transport the cargo from one barge (American Mariner 2009). As over one billion tons of transported cargo was recorded on the U.S. inland waterway system in 2007 (Waterborne Commerce Statistics Center 2009), maintaining the inland waterways serviceability is critically important to the nation’s economy, transportation system, and environmental health. Infrastructure capital investment and maintenance,, including locks, dams, and flood protection structures, along with regular channel dredging is necessary to ensure the smooth flow of vessels and commodities.

Improvements to the U.S. navigable waterway channels and infrastructure under the jurisdiction of the Corps must be justified based on analysis of the navigation benefits that the improvements will provide. Within the Corps, economic justification is done within a framework of comparing the “with-project” and “without-project” conditions, effectively discounting the benefits that would accrue if the Corps did nothing to improve the system. These benefits are dependent upon their affect on lock processing time or easing of congestion due to constraints within the system, for example, improving lockage times through rehabilitation or investment in locks, or altering the traffic flow through bend easings and relief of movement restrictions. These benefits are measured and compared to the cost of the particular improvements being analyzed. The BasinSym simulation model allows the estimate of transportation time and cost under varying future conditions.

2. OVERVIEW
System simulation of this problem at the microscopic level is highly complex. In order to simulate the individual movements of the vessels, a meaningful way to describe those movements is required. Tows and the barges they move must be represented as complex aggregations of individual vessels and the commodities they carry. While some tows operate in a shuttling capacity, moving barges from point A to point B and back again, others operate in a long haul capacity and still more are picking up empty barges and dropping them at economically advantageous destinations. Further complications arise, as tows change their barge configurations during their journey in order to effectively navigate the inland system. As well, an understanding of the motivations behind shipper response to congestion (i.e., when is a change in transportation mode likely?) must be gained in order to model the problem correctly.

The BasinSym model captures these complexities to assist with analysis of the problem. BasinSym is designed to simulate inland traffic movements on the waterway system and by making runs with differing configurations and fleets allows for comparative analysis of different conditions. Detailed analysis of potential improvements over a 50-year planning horizon requires determination of not only the existing condition but the operating characteristics of proposed lock improvements, the future servicing fleet and anticipated demand in commodities as well. BasinSym has been designed to provide the analyst with a set of tools to thoroughly investigate these problems in order to provide a justifiable basis for extrapolation of this information 50 years into the future.

3. SYSTEM REPRESENTATION

BasinSym utilizes a user-defined network to describe the physical characteristics of the waterway system being studied. The network consists of a series of reaches (channels), locks and docks as well as information about the physical characteristics and capacity of each.

The system is represented as a node-link network. Each link is a reach, representing a portion of the waterway (between nodes) on which vessels travel. Nodes may be referenced to a river mile indexing system, as well as by geodetic coordinates. Upstream and downstream nodes are defined based on direction of water flow.

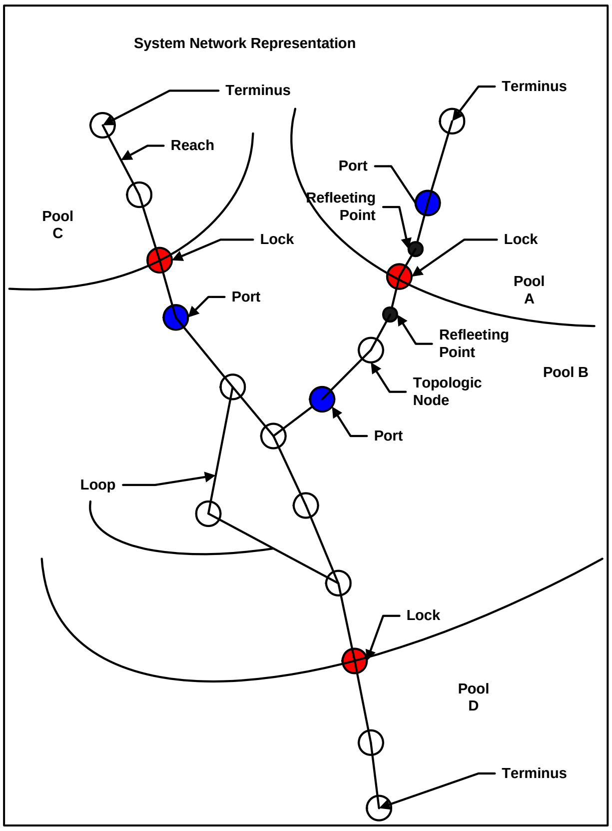

Figure 1: System Network Representation

The following node types are considered:

- Topologic nodes—serving only as start and end points of reaches;
- Port/dock nodes—serve as origin/destination of commodity and/or vessel movements;
- Terminus nodes—serve as entrance/exit points to the portion of the network under study and serve as sources and sinks.
- Re-fleeting Points—locations at which a tow can be reorganized into a different configuration with a different towboat. These can be co-located with a port node, to allow for representation of re-fleeting at a port.

All reaches are connections between a pair of nodes. Reaches may be either Open Channel or Lock Reaches. Open Channel reaches are segments where vessels can move subject only to transit rules while Lock Reaches consist of a lock and associated internal geometry. Reaches may have additional descriptive attributes assigned. In addition, vessel speed is associated with individual reaches, that is, vessels may travel at different speeds in different reaches, as specified by input data. Transit restrictions in a reach are defined by rules associated with the reach, specifically, no overtaking, no meeting, single vessel reach and one-way traffic.

Regions can also be defined by the user, e.g., pools, governmental jurisdictions, port hinterlands, etc., by associating nodes and reaches of the network with such regions. Allowing for user-defined regional definitions is useful for performing disaggregation on external data that is available on a spatial region basis or for aggregations on model-generated data.

4. FLEET DEFINITION

The calling fleet is described as a set of *vessel classes* that contain information on the characteristics of each type of tow and barge. A single classification table houses the data necessary for defining categories of vessels, whether they be barges, tugs, or even recreational vessels. The following data items are used to classify the vessels used during simulation:

- VesselClass - user defined name (e.g., Jumbo Hopper, 3000 HP, etc…)
- CostStaffed - total vessel cost
- CostMoored - cost at-port for barges
- ISPowered - indicates whether class contains powered vessels or barges
- Horsepower - triangular distribution specifying horsepower class
- LOA – triangular distribution specifying overall length for the class
- Beam - triangular distribution specifying beam width for the class

Unique flotillas (power vessels and barges) are stored separately in a table of unique barges and a table of unique powered vessels respectively. For barges, the following fields apply:

- LOA - overall length of barge

ISBN 978-84-692-5416-5

- Beam - beam width
- TPI – tons per inch indicating amount of draft added per ton of load
- LightDraft - draft of barge when empty
- MaxDraft – draft of barge when fully loaded

For power vessels, the following data items are used:

- Name - name of the vessel
- HP - horsepower of vessel
- LOA - overall length of vessel
- Beam - beam width of vessel
- TPI - tons per inch indicating amount of draft added per ton of load
- LightDraft - draft of vessel when empty
- IdleFuelBurn – daily number of gallons burned by vessel when waiting
- RatedHPFuelBurn - daily number of gallons burned per horsepower produced
- VesselCapability - integer indicator of what vessel capability level, valid values are: (1) vessel is capable of pushing barges, (2) vessel is capable of carrying cargo

Powered vessels can incur time at port nodes (for loading and unloading and for tow configuration), re-fleeting points (for re-configuration and change of tow) and at locks (for passage through the queue and lock). It is assumed that vessels do not incur any time when traversing a topologic node, unless they must wait before entering the next reach, due to transit rules.

4.1. Trip Specification

With this foundational information for a particular system in place, BasinSym employs a shipping manifest to provide the simulation with specific tow/barge movements that are subsequently routed through the system. This manifest, or shipment list, includes information on when the vessel departs on a journey, and commodities picked up and/or dropped off along the way. The specification of trips is accomplished through a structure consisting of a set of relational database tables which are slightly denormalized in the tables below for simplification purposes. In the real world, vessels are continuously traveling the waterway; for purposes of simulation an individual trip is defined as the set of reaches between a change in direction of the powering vessel. Even with this simplification, a complex data structure is needed to capture the information relevant to a trip.

Table 1: Trip Specifications

TripID	Trip Date	Power Vessel	Origin Node	Destination Node
1	Jan 11, 2009	1	1	6
2	Jan 12, 2009	8	4	10

The data in Table 1 defines each specific trip of an individual powered tow. This includes the begin date of the overall trip as well as the origin and ultimate destination of the trip overall. The individual node visits where commodities are loaded /unloaded or barges are added/removed along the entire route are specified in Table 2. The node visits are also given a specific order within the overall trip to provide for sequencing. A visit contains 1 or more transactions related to changes in power vessel, additions or removal of barges from the tow, or cargo related operations.

Table 2: Individual Node Visits

Visit ID	Trip ID	Visit Node	Visit Order
1	1	1	1
2	1	6	2
3	2	7	1
4	2	8	2
5	2	10	3

Barge removals/additions to the tow are handled through specification of an add or drop command for a specific number of unpowered barges at pertinent visit nodes as shown in Table 3. It is important to note that although powered vessels are stored and identified individually, this data structure does not indicate specific barges to be transacted but rather a count by class of vessel and by default they are assumed to be unloaded.

Table 3: Add/Drop Command

Action ID	Visit ID	Action	Vessel Class	Count
1	1	ADD	Jumbo Hopper	1
2	1	DROP	EMPTY	2
3	2	DROP	Jumbo Hopper	1
4	3	ADD	Hopper	2
5	4	ADD	Deck	2
6	4	ADD	Hopper	4
7	4	DROP	EMPTY	2
8	5	DROP	Hopper	4

Cargo transactions, displayed in Table 4, account for the final element of the trip specification. Cargo that is loaded or unloaded is described in terms of commodity, quantity in tons, and whether the cargo is to be loaded or unloaded. These transactions coupled with the ordered set of visits and actions within a power-trip fully define a trip specification.

Table 4: Cargo Transactions

Action ID	Commodity	Qty (tons)	Load/Unload
1	CORN	10	LOAD
3	CORN	10	UNLOAD
4	COAL	12	LOAD
5	MECH EQUIP	8	LOAD

6	COAL	30	LOAD
8	COAL	30	UNLOAD

5. ROUTING

Vessels travel on a designated route from origin port to destination port. Where alternative routes are possible (in a network with loops) it is necessary to determine which route is chosen. This is done adaptively as a vessel observes congestion and chooses an alternate route. Route selection in the simulation of barge traffic represents a balance between computational tractability and emulation of a real-world decision making process. Three major questions arise during assessment of the decision making process – When does a tow determine its route? When will a tow change its route and what triggers the route change? Will a tow always choose the least expensive route and what metric determines cost of a route? These questions and the assumptions made as to the answers form the foundation of routing in the BasinSym model.

First, it is necessary to understand how tow movements are specified in the simulation. As stated earlier, a tow is assigned a trip which consists of an ordered set of node visits. The order is assigned and cannot be changed as the order of cargo and barge transactions must be preserved to avoid the case in which barges are dropped off at a node before they are picked up. Thus, a trip is an ordered set of (potentially) smaller voyages or legs between nodes to be visited. BasinSym assumes that the initial (preferred) route for the entire trip is established prior to the first movement. Based on this assumption, prior to the first movement, a trip route consisting of the summation, with order preserved, of all leg routes, is created for the tow/trip.

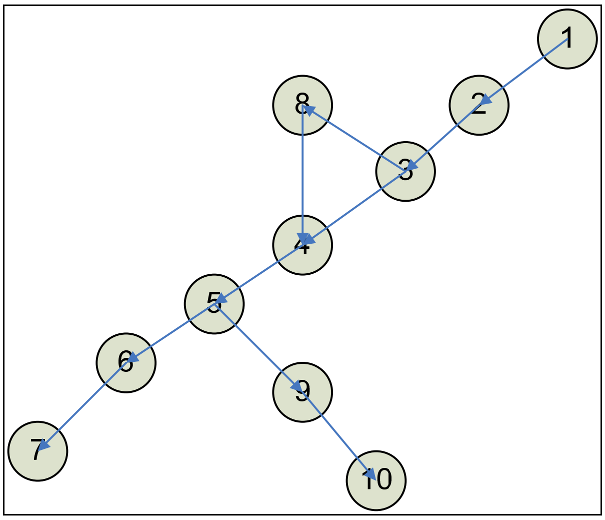

Figure 2: Trip-Visit Route

An example of a trip-visit route is shown in the figure above. A tow is given a trip starting at node 1 and ending at node 7. It has visits, in order of 1, 8, 9, and 7. Thus, we would determine routes between nodes 1 and 8, 8 and 9, and 9 and 7. Those routes would appear as follows:

1. (1 to 8): 1-2-3-8
2. (8 to 9): 8-4-5-9
3. (9 to 7): 9-5-6-7

The overall trip route would be 1-2-3-8-4-5-9-5-6-7. Tows store their visits in a FIFO queue structure and remove visits from the queue as they are completed, the leg routes are stored as FIFO queues as a subset of the visit storage structure.

5.1. Algorithm

BasinSym processes the shipment list by date and routes the movements on a least-cost path from origin to destination using the A* (A-Star) algorithm (Hart, Nilsson, Raphael 1968). A* utilizes a distance- plus cost heuristic for determination of the route. The cost portion of the heuristic within BasinSym is based on estimated transit time along the route to take into account any network congestion that may exist.

A* is both optimal and complete given that the evaluation function f(x) is admissible. The evaluation function in A* is a compound function consisting of g(x), the cost of the route from the origin to the current node, and h(x), the estimated cost of the remaining route from the current node to the destination. While this sounds simple, there are considerations. The heuristic, h(x), must be optimistic in that it must never over estimate the remaining travel cost. Poor construction of the heuristic will negatively impact the algorithm. In the worst case scenario, h(x) is set to a constant value of 0. This degrades the search from Best-First to a simple Depth-First search.

In the transportation modeling domain, the straight-line distance between two nodes is often used as a heuristic for optimistic shortest path given that the shortest distance between two points, in simple Euclidean geometry, is a straight line. The heuristic, h(x), implementation within BasinSym is the straight-line distance between the two points divided by the fastest speed limit on any reach for the vessel class of the power vessel of the tow. This requires that all nodes have valid geospatial coordinates assigned to them in a format that allows the determination of distance. The Great Circle Distance formula is applied to determine "straight-line" distance on the surface of the Earth. The distance obtained is divided by the top speed of the tow as determined by its power vessel class in order to provide a measure in hours as opposed to miles. The choice of the top speed is made as the evaluation function must be optimistic in order to be admissible and thus never overestimate the path cost.

The second component of the evaluation function, f(x), is the cost of the path from the origin or g(x). In that the evaluation function must be optimistic, each of its components must also be optimistic to include the cost of the path traversed. G(x) within BasinSym uses the vessel's top speed, governed by reach speed limits, over the length of the reaches traversed. This is implemented through use of a weighted graph structure to capture the nodes and reaches of the navigation network. Since simple weighting of the arcs without consideration to limiting factors is insufficient for effective route production, the vessel class of the tow's power vessel is used to set arc weights. This allow for establishment of extreme levels of impedance where a tow would be violating simple reach rules. The weights are stored in each arc as a dictionary of traversal times by vessel class. Cost to traverse a lock represents a slightly more complex situation. Given that the estimate must be optimistic, the cost is determined by the lower bound of any processing distributions for the fastest chamber at the lock given that there are no vessels in any queues. In order to simplify this, the lock objects contain a method called *Lock.EstimatedTraversal* based on vessel class. The result of this method is used to weight the inter-node arcs where the reach is a lock. Time to traverse the path is summed across the reaches to produce a consistent unit of measure – time to traverse.

5.2. Route Generation and Selection

Each tow retains the necessary information to describe its complete route. The route is stored as a set of intermediate destination nodes within the trip-visit structure of the tow. As each route is generated, it is stored in a dictionary of routes indexed by origin, destination, and class of vessel. These routes are determined upon creation of the tow and potentially added to when congestion or closure dictates a route alteration is desired. The predetermined routes are unchanging regardless of network condition or situation. These fixed routes are not purged under any circumstance for the duration of a simulation. This implementation avoids the resource cost of searching for a route that has already been found and provides a commensurate improvement in execution speed.

Periodically, a tow will need to alter the initial route due to congestion on the waterway or closures at a lock. Route selection is altered or recomputed when a high impact event such as a lock closure, an event that makes any route impassable, or an estimated lock traversal time that exceeds some user defined threshold. Upon receipt of a high impact event message, all tows in the system recalculate the best route from their current position to the completion of the current leg as well as any remaining voyage legs. When the simulation encounters a high impact event, the route dictionary is purged of all prior information with the exception of fixed routes. The model subsequently iterates through all tows that are active in the system. For each of those tows, routing information is invalidated and the tows regenerate routes from their current position through the end of the voyage.

6. LOCKS

BasinSym provides the user the ability to define all locks in the system to be as one of three types depending upon the level of detailed analysis required at a particular lock in the system. First tier locks are the simplest and utilize a user-defined distribution to define the amount of time it takes for a particular tow class to transit the lock. Processing distributions are fed to the model for upbound and downbound movements of tows respectively. An additional distribution set exists for the upbound and downbound movement of recreation vessels. All lockage tiers utilize a separate distribution to represent the time it takes to fill and empty the chamber.

Flotillas traversing the inland waterways of the US are often larger than the locks they must pass through. To pass through the lock the flotilla must be broken into "cuts". Second tier locks add the complexity of processing vessels using a set of distributions depending on the number of cuts required to completely lock the vessel through. For example, the first cut through a lock would utilize a different processing distribution than the second and/or third cut. Cut determination is handled using a packing algorithm which will be discussed in detail later in this document.

Tier three locks are the most detailed and move locks through using a set of processing distributions for tow/barge class and number of cuts but these distributions are broken out by lockage type on approach and exit as well.. For example, a particular distribution exists to represent the processing time for an upbound tow making a "fly" approach while another distribution exists for the same tow utilizing a "turnback" exit. Additionally, tier 3 locks have the added capability to model gate or approach area interference as well. Multiple lockage polices are available for use at the lock chamber level for tier three locks. While first and second tier locks use a First-In-First-Out (FIFO) queue, third tier locks chambers can be defined to utilize FIFO, shortest processing time and n-up/m-down lockage policies as well.

7. CUT COUNTING

Within the BasinSym, several of the lock simulation models require determination of the number of operations or 'cuts' that will be required to completely pass a tow in order to simulate the lockage, which is a function of the number and size of barges in a tow, and the manner in which the barbes can be packed into a lock chamber. The development team originally employed a min/max technique for simulation of cut-level lockages in which a pseudo-barge whose dimensions are determined based on greatest overall beam and LOA was tiled into the chamber to determine the cut count. While this method worked well when tows of uniform composition were processed, it would overestimate the cut count, potentially drastically, with tows of heterogeneous barge compositions. In order to reduce this estimation error, a new technique based on bin packing has been implemented.

7.1. Chamber Packing

The problem, put simply, is how to fill a rectangular space (the chamber) the fewest number of times given a required set of rectangles that must be included (the tow). Computer science lends us candidate solutions in the form of a family of spatial packing algorithms known as strip packing algorithms. Within the strip packing family of algorithms, there are two sub-families: online and offline. In online strip packing, a stream of rectangles is placed in the strip as they are presented to the algorithm. An example of this would be the loading of a truck with parcels from a conveyer belt. The offline sub-family of algorithms assume that the entire input stream is fully known to the processor prior to placement of the first rectangle. An example of this would be the packing of a truck where the loader is shown the pile of boxes to be loaded. Given that a tow operator has knowledge of locks that a tow will pass through, advanced knowledge of the tow configuration prior to lockage leads to the selection of an offline strip packing algorithm. The First Fit, Decreasing Height (FFDH) algorithm (Cardiff University 2008), was selected due to a balance between processing time and packing efficiency. In this algorithm, the barges are sorted based on length with the longest barge first in the list. The list of barges is processed in order (longest to shortest). A "strip" is created with its maximum length being that of the chamber. On that strip, "levels" are created. The maximum width of a level is the width of the chamber while the length of the level is determined by the first barge placed on that level.. The barge at the head of the processing list is placed in the first level that can accommodate its width. In the event that there is insufficient width in any of the existing levels, a new level is created if there is sufficient remaining length on the strip. If there is not sufficient remaining length, the strip is ended, cleared, and the cut count incremented. This is the basic functionality of the FFDH strip packing algorithm.

Within BasinSym, a distinction is made between a chamber with some form of assistance and a chamber without. In a chamber featuring assistance, the power vessel is included in only one cut while chambers without any form of assistance require the presence of the power vessel on each cut. For reasons of simplicity, the power vessel is processed in the first cut.

7.2. Chamber Packing Example

The step-wise execution that results from the specific implementation of the FFDH algorithm in BasinSym is as follows:

1. The barges are sorted based on decreasing length yielding a processing list of: A1, A2, A3, A4, B1, B2, B3. The power vessel is not included in the processing list.

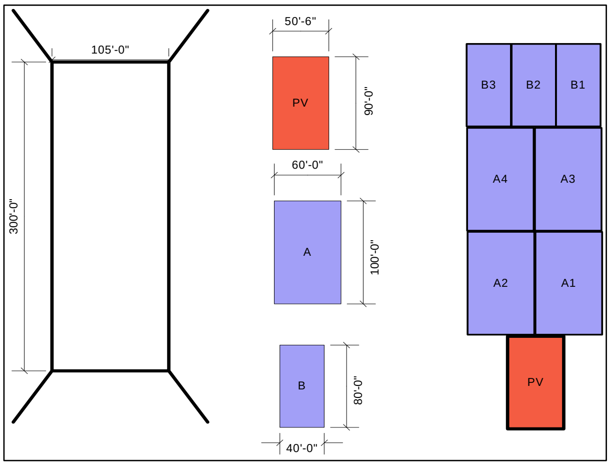

Figure 3: Chamber Parking – Step #1

2. The power vessel is placed in a new level on the strip – L1. L1 90' long and has an available width of 105'-50.6'.

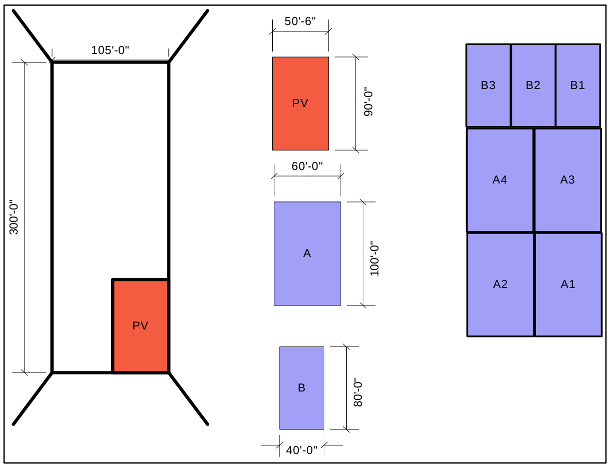

Figure 4: Chamber Packing – Step #2

3. Barge A1 is examined. It cannot fit on L1 due to length so a new level is created on the strip – L2. L2 is 100' long and has an available width of 105'-60'. The strip now has 300'-90'-100' of available length.

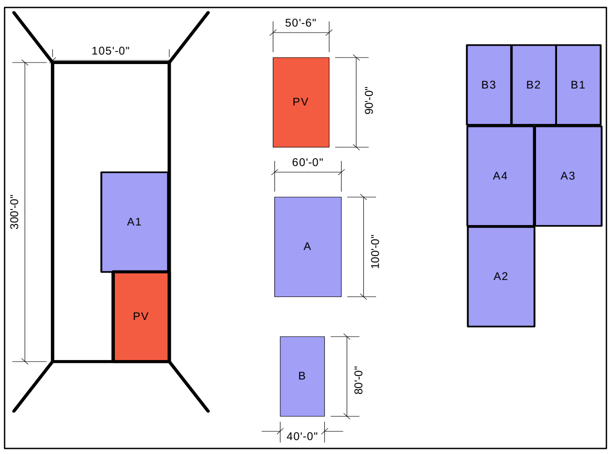

Figure 5: Chamber Packing – Step #3

4. Barge A2 is examined. It cannot fit on L1 due to length. It does not fit on L2 as the available width is insufficient. A new level L3 is created and A2 is placed on it. L3 is 100' long and has an available width of 105' – 60'. The strip now has 300' – 90' -100' -100' (10' available).

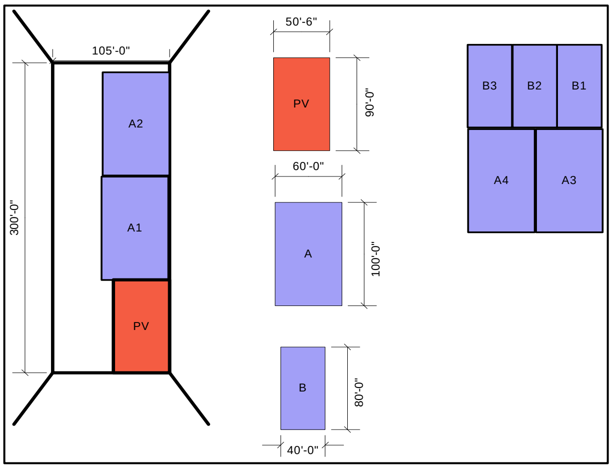

Figure 6: Chamber Packing – Step #4

5. Barge A3 is examined. It cannot fit on any of the existing levels and a new level would exceed the maximum strip length. The strip is cleared and the cut count incremented to 1. L1 is created on the strip with a length of 100' and an available width of 45'. The power vessel is not reintroduced as assistance does not require it.

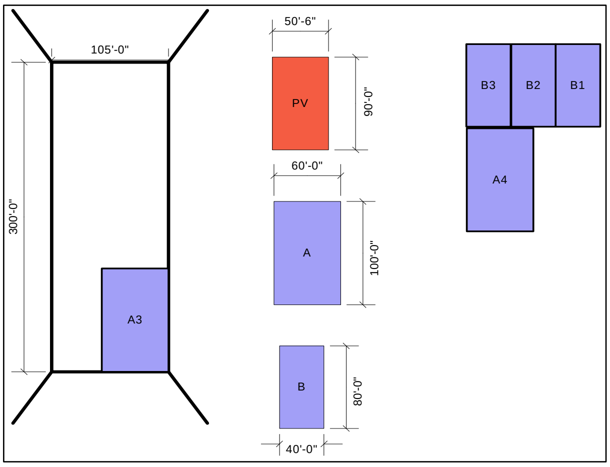

Figure 7: Chamber Packing – Step #5

6. Barge A4 is examined. It cannot fit on L1 as there is insufficient remaining width. L2 is created with a length of 100' and a remaining width of 45'. The strip now has 100' of available length.

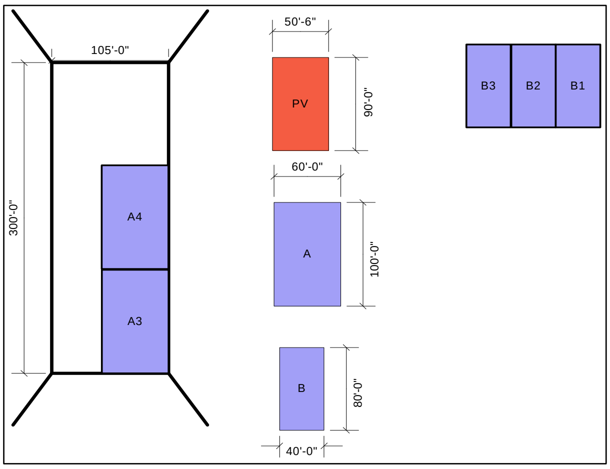

Figure 8: Chamber Packing – Step #6

7. Barge B1 is examined. It fits on L1 leaving L1 with 45' – 40' of available width.

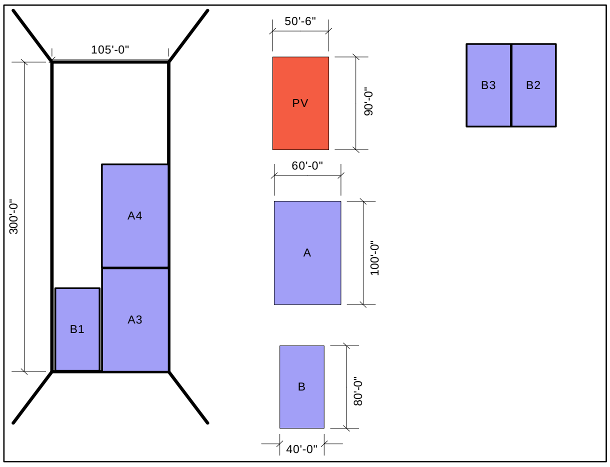

Figure 9: Chamber Packing – Step #7

8. Barge B2 is examined. It does not fit on L1 due to insufficient width but does fit on L2 leaving L2 with 5' of available width.

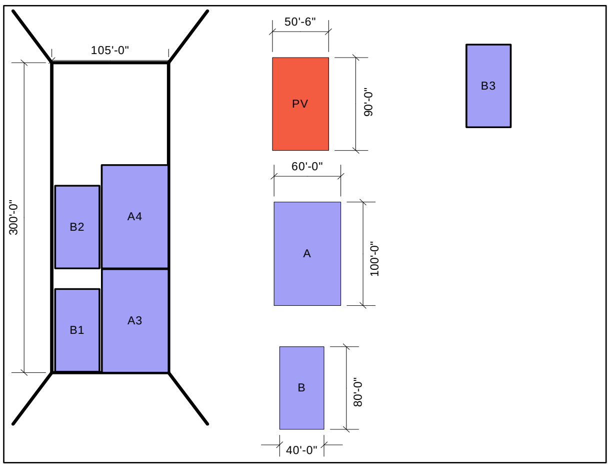

Figure 10: Chamber Packing – Step #8

9. Barge B3 is examined. It does not fit on either L1 or L2 due to insufficient width. L3 is created with a length of 80’ and an available width of 105’ – 40’. The strip has 300’ – 100’ -100’ -80’ of available length.

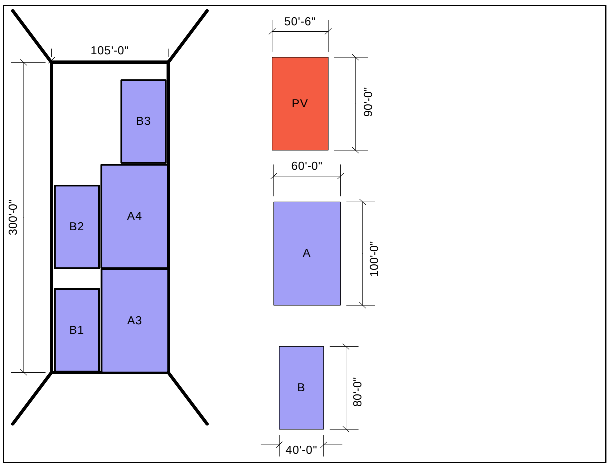

Figure 11: Chamber Packing – Step #9

The barge processing list is now empty. The cut count is incremented as the strip is not empty. This yields a cut count of 2 in our example. If assistance had not been available at the chamber, the power vessel would have been inserted each time a new strip (cut) had been created.

7.3. Implementation Details

The process of cut counting is not trivial in terms of computational resources. In a simulation consisting of hundreds of iterations over a 50 year horizon in which each year could have tens of thousands of lockages, this represents a tremendous amount of time expended. In consultation with USACE field experts, several assumptions have been made that allow for potentially huge savings in computational resources through the reuse of previous results. The first assumption is that the specific type of barge does not affect how it is packed in the chamber. This assumption can be reduced to “a barge that is X feet long by Y feet wide is a barge X feet long by Y feet wide”. The second simplifying assumption is that cut count does not vary by particular chamber or location for chambers of a given dimension.

The data structure employed for this is a tiered dictionary of cut counts based on a chamber dimension signature and a tow composition signature based on barge dimensions. This dictionary is persistent across simulations thus eliminating the need to count cuts unless a previously unseen combination of chamber dimensions and tow is encountered at which point the information is determined and never recalculated.

The chamber signature is a string used to identify chambers of equivalent dimension and assistance availability. In order to be more open to inspection, this string is quite plainly human readable. The components for the chamber signature are length, width, and assistance with each element separated by an ‘x’ and the entire triplet enclosed in parens. An example for a chamber that is 100.5 feet long by 56 feet wide without assistance would be: (100.5x56xNOASSIST). Thus, the signatures of dimensionally equivalent chambers are identical.

8. RELIABILITY

Reliability is an important issue, as many of the locks on US waterways are more than 70 years old, well beyond their initial design life. Reliability considerations are handled within BasinSym using the concepts of component states and state transitions. Under this approach, at minimum one component is assigned to each chamber. The component can be in one of a number of states, with transitions between the states defined probabilistically, based on component state transition functions (Males 2004).

A lock is represented as a hierarchy, consisting of a lock, composed of one or more chambers. Each chamber is composed of one or more components. There is no particular physical definition or behavior associated with a component—it is an abstract general concept. A component is simply something that can fail and whose current status participates in determining the overall performance of the lock. This concept allows the modeling effort to focus on the specific components of interest. Each component has an associated value of “age” and operating “cycles,” that is incremented as the simulation proceeds and that can be reset by a component repair or rehabilitation iteration.

In order to make data handling and modeling feasible, the concept of component states is used. Each component can occupy, at any given time, one of a set of user-defined states specific to that component. A miter gate at the entrance/exit of a lock chamber, for example, might be in one of three possible states—excellent, poor and non-operational. A guidewall might be in one of four possible states—very good, medium, poor and highly degraded. Each component can transition between its available states. The probability of moving to another state is a function of the current state. This is generally referred to as a Markov process.

The driving force for moving a component from one state to another state can be either the passage of time or the cycling of the lock due to the passage of a vessel. Thus, over time, components can fail due to age, stresses associated with repeated opening and closing, or a barge can collide with the gate, causing major or minor damage.

8.1. State Change Probabilities

In order to define probabilities associated with a change in state of a component, *state transition probability curves* are associated with the component for each of the event types and failure modes that are expected. These curves are typically referred to as PUP (probability of unacceptable performance) functions which may be defined as either age or cycle based. An example of a cycle based PUP function is shown in figure 12 below.

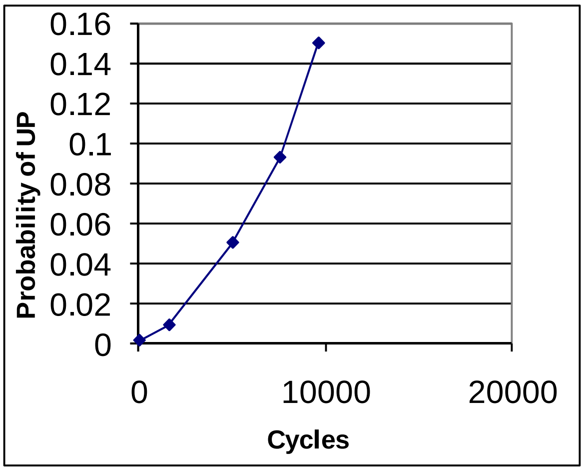

Figure 4: Cycle Based PUP Function

Recall that, for each component, the age and number of cycles are known (and continuously updated during the simulation). Thus, at any given time, when an event takes place that may cause a state transition, it is possible to determine, by curve lookup, the associated current probability of that state transition.

8.2. Failure Modes

For each component state, it is possible to have multiple failure modes, with different probabilities of occurrence. When an event that can trigger a state transition takes place, the current values of the point probabilities of each failure mode are determined from the associated lookups into the PUP functions (based on current component age or cycles). These probabilities are arrayed cumulatively on a probability line, a random number is generated, and the determination is made if there is a failure and, if so, which failure mode should be selected.

For each failure mode, a repair cost and duration are defined. The repair for a given failure can involve changes to more than the component that underwent failure. For example, if a gate fails, then the electrical system component can also be replaced. The user defines all the component repairs associated with each failure mode. Each such component repair involves setting the current state for that component and revising the current age and cycle.

If the particular failure mode is activated, then the chamber is out of service for the duration period and the associated cost is added to the economic analysis. At the end of the repair, the component states, ages and cycles are re-set to the user-defined values. The post-repair state can be the same as the current state or it can be different. For example, a minor repair can simply consume time and dollar resources, without making a significant enough performance change to move to a different component state.

The resetting of age and cycles associated with a repair can be either absolute or relative. If the values for post-repair age and cycle are positive, then the component's current values are set to these values. If, however, either of the values is negative, then the change is made relative to the current value. For example, if the current age of a component that is being repaired is 25 years, and the user entered value post-repair is 10, then the age is reset to 10 years. However, if the entered value is –5, then the post-repair age becomes 20 (25-5).

8.3. Scheduled Rehabilitation

A rehabilitation (rehab) is a scheduled outage at the end of which component state changes (or age/cycle changes) take place. Rehab events take place at user-defined times, with specified costs and outage durations and associated component rehabs, exactly analogous to component repairs in that a new post-rehab state, age and cycle are specified. Each set of rehab events is grouped into a rehab plan that can be activated when the simulation is run. Thus, multiple rehab plans (alternatives) consisting of different combinations of component rehabs can be stored and tested.

This approach is implemented for each lock/chamber, handling the level of detail through data. That is, at minimum each chamber at a lock is represented by a single component. Some locks, e.g., a lock at which a rehab study is being done, would be represented by more components, with more states and failure modes. A “simple” component for another lock might have two states, a single failure mode and a single associated repair.

8.4. Performance Penalties

The concept of a performance penalty is used to relate the state of the components to the performance (in terms of time spent serving vessels) of a chamber. At any given time, the total performance penalty that is added to the lockage time is calculated as the sum of the individual state-based performance penalties, given the state that each component is occupying. The performance penalty is probabilistic and associated with a particular stage of locking (e.g., approach, entry, chambering or exit). Thus, a component representing a set of valves would apply the performance penalty to the chambering or chamber turnback stage, as would the gate performance penalties.

9. DATA DEVELOPMENT

Data for development of the shipment lists comes primarily from the Corps Lock Performance Monitoring System (LPMS) and the Operations and Maintenance of Navigation Information (OMNI) system, corporate data warehouses that store information on every lock transit under Corps jurisdiction. As part of the NaSS suite of tools, a data extraction and analysis tool was developed for the LPMS / OMNI data stores and subsequently named the Data Analysis Pre-Processing (DAPP) module. The DAPP takes as input a database schema from the external data warehouses and provides a set of tools for developing shipment lists for input into the simulation model based on historical movements, statistics and commodity demands. There are three types of shipment lists available, historical – for calibrating the model; statistical – for using historical movements to create synthetic movements through distributions; and commodity driven – for developing a synthetic fleet to satisfy a projected demand of commodities at the dock level. Once a particular shipment list is exported from the DAPP it is ready for use as the simulation driver within BasinSym.

The DAPP includes utilities to produce distributions which feed the probabilistic data requirements of the BasinSym. The DAPP is able to quickly analyze a subset of traffic information for a set of locks and provide cumulative distribution functions to represent lock transit time based on a user defined set of parameters, such as vessel class, lockage type, number of cuts and date range.

10. CONCULSION

The NaSS suite of tools can be used to perform a basin-level analysis of inland waterway traffic and compare baseline conditions to a set of alternative projects. This information provides the basis of an economic analysis required by the USACE for evaluation of investment decisions. Detailed output logs are available once the simulation completes as well as .csv formatted outputs of transit times by reach, vessel class and lock.

The NaSS suite has been initially populated with historic vessel movement data from 2007 for the Ohio River basin, which includes over 2,500 specific shipments for the one year period. The NaSS suite was successfully used to analyze the potential benefits of conducting concurrent, as opposed to sequential, lock chamber closures on the Ohio River. The analysis showed concurrent lock closures would save about $1.3 million in delay cost at Byrd-Greenup Lock and Dam's compared to sequential closures at those locks. The analysis further showed concurrent closures at Myers-Smithland Lock and Dam's would save only about $126,000 due to the main and auxiliary chambers at Smithland being of identical size. The entire analysis was performed within a 2 week timeframe and the results were subsequently used to help set the 2009 lock and dam maintenance schedule on the Ohio River.

ACKNOWLEDGMENTS

The authors wish to express their sincerest gratitude to Dr. Richard Males of RMM Technical Services in Cincinnati, Ohio, USA who was responsible for the initial design document that served as the blueprint for this model. Mr. Mark Lisney of the US Army Corps of Engineers Institute of Water Resources serves as the project manager of the NASS suite and was an invaluable resource as a subject matter expert on inland waterway movements. Model development for NASS was carried out by Mr. William Woelbeling and Mr. Michael Dean of Camp, Dresser, McKee, Inc. in Carbondale, Illinois, USA.

REFERENCES

American Mariner LLC. 2008. *Inland Waterways*. Available from: http://www.americanmariner.net/11_inland_waterways.php [accessed 11 April 2009]

Cardiff University, 2008. School of Computer Science, Level and Shelf Algorithms. Available from: http://users.cs.cf.ac.uk/C.L.Mumford/heidi/Approaches.html [accessed 8 March 2008]

Hart, P. E.; Nilsson, N. J.; Raphael, B. (1968). "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". *IEEE Transactions on Systems Science and Cybernetics* SSC4 4 (2): 100–107.

Males, R., 2004. *LockSym Draft Documentation*, Available from US Army Corps Of Engineers Institute for Water Resources.

Waterborne Commerce Statistics Center, New Orleans, Louisiana, USA. 2009. *Final Waterborne Commerce Statistics For Calendar Year 2007.* Available from: http://www.iwr.usace.army.mil/ndc/wcsc/wcsc.htm [accessed 16 April 2009]

AUTHORS BIOGRAPHY

Keith D. Hofseth

Mr. Hofseth has served as an economist at the Corp of Engineers for over 24 years; conducting studies in flood control, deep draft navigation, inland navigation and hydropower. He co-authored the first guidance for major rehabilitation analysis and was a pioneer in bringing risk and uncertainty into the Corp's planning process. Mr. Hofseth started with the Corps in the Seattle district and later served as the Chief Economist for the Alaska District. He is currently the Technical Director of the Navigation Economic Technologies research program at the Institute for Water Resources and is a graduate of the Corps' Planners Associate Program. Mr. Hofseth holds a B.S. degree in Finance and Economics from the University of Montana and an M.A. degree in Applied Economics from Johns Hopkins University.

Cory M. Rogers

Mr. Rogers is a computer scientist currently serving as the National Practice Leader for Applications Development at Camp Dresser & McKee, Inc. He has 15 years of experience in the design and development of software applications to solve complex decision making problems. He has managed teams to develop successful software applications ranging from single-user desktop applications to web-enabled systems with a user base of thousands. Prior to joining Camp Dresser McKee, Inc, he served as the Director for Software Engineering for a small consulting firm and was a founding member of the Curriculum Advisory Board for the Information Systems Technology program at Southern Illinois University. He has worked on numerous decision support software models which are currently being used to perform crucial functions such as estimating flood damages caused by coastal storms, investigate the economic feasibility of improvements to the U.S. inland waterway and port infrastructure and manage financial risk at global food and beverage companies.