

# **ASSESSMENT TOOL FOR NUCLEAR WEAPON ACQUISITION PATHWAYS**

William S. Charlton and D. Grant Ford

Nuclear Engineering Department

3133 TAMU

Texas A&M University

College Station, TX 77843-3133

## **ABSTRACT**

An assessment tool has been developed at Texas A&M University for predicting weapons useable material acquisition based on an acquisition network simulation. The network has been designed to include all of the material, facilities, and expertise that must be obtained to acquire Special Nuclear Material (SNM), each of which are represented by a unique node. Using various historical cases and open source expert opinion the resources required to successfully obtain the goal of every node within the network was determined. A visual representation of the network was created within Microsoft Visio. This program utilizes Visual Basic for Applications (VBA) to run its default macros, and enables the user to include any additional coding. Using this feature a tool has been developed to predict that most likely pathway(s) that a predefined organization is most likely to take. The methodology uses the resources available to the organization, along with any of the nodes the organization may already have access to, to determine which path the organization is most likely to find the most attractive.

## **INTRODUCTION**

There are two types of nuclear proliferation: horizontal and vertical. Vertical proliferation refers to countries that already have weapons producing more and/or better weapons. Horizontal proliferation refers to countries/organizations developing a nuclear weapons capability where there was none before. While vertical proliferation is a concern to the global community, there are numerous well understood efforts that can be employed to lessen its impact. The problem of horizontal proliferation is generally more complex. Every case of horizontal proliferation in recent history (e.g. Iraq, South Africa, North Korea, etc.) has been covert, and in many cases the method of proliferation was unexpected. A proliferator will not necessarily follow the route that we expect, but will make an intelligent assessment of the resources available to them and the probability of success without discovery before deciding where to spend those resources.

In this work, a tool was developed which allows an analyst to determine the most likely paths an organization will take in an attempt to acquire a nuclear weapons capability. This tool accounts for the resources (expertise, facilities, manpower, money, materials, etc.) available to the organization. It makes use of an extensive network model for the acquisition of the special nuclear material needed for the production of a nuclear weapon. The tool was developed in Microsoft Visio, it allows an analyst to examine how

SNM might be produced covertly, what type of resources are required to do so, and the best way any given set of resources might be used.

## NETWORK DEVELOPMENT

The relevant set of expertise, facilities, and materials need to be presented in a way that allows the assessment tool to assemble them into correct and coherent paths. This was done by displaying the various fields of expertise, facilities, and materials as nodes on a network. This allows the user to see all of the interconnectivity inherent in the various nuclear, chemical, and physical processes.

In order to develop a network consisting of every way special nuclear material may be attained the level of detail of the system had to be balanced with the information that was thought to be available for determining the individual parameters of every node included in the network. For example, the “Build Reprocessing Facility” node could have been broken into many different components ranging from obtaining each individual chemical required to chopping the spent fuel to actually assembling all of the industrial components required for such a facility; however, assigning likelihoods and resources required for completion and times of completion for most of these steps would have been cumbersome. Thus, a more general approach was taken. It should be noted however that any additions or greater level of detail that may be desired in the future would be relatively easy to add if the required data was available.

The assumption was made that in order to produce any material, a modern organization would follow a series of steps: develop the basic skills necessary to facilitate producing the material of interest, build the required facilities, developing more refined skills along the way (these may be included in the building of the facility or, if necessary, denoted as a separate skill set), and operating the facility to produce the material.

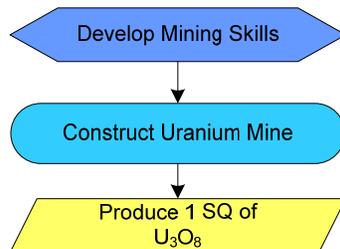


Figure 1. Example of nodes used within the network.

Figure 1 shows the basic structure of a very small subset of nodes in the network. This figure contains every type of node in the network: Skills, Facilities, and Materials. Keeping the network manageable necessitated keeping the network simple. Thus, the types of nodes introduced were kept to only those absolutely necessary for completeness. There are 5 basic types of nodes in the network: (1) Skills, (2) Facilities, (3) Materials produced in an indigenous facility, (4) Material acquired through theft, and (5) Material purchased on the black market. Each type of node will require a specific set of resource types.

The network considers metallic HEU and metallic Pu production with a large array of possible starting points and paths to that end. All viable enrichment and

reprocessing technologies are included as well as mining, chemical conversion, and machining. Also, the network accounts for the presence of a nuclear black market and includes the possibility of purchase or theft of any of the materials. This network then defines all of the possible paths a proliferator might choose.

## NODE DEVELOPMENT

The network must remain manageable but not by sacrificing too much detail. The solution to this dilemma was for the network to consist of nodes which had properties that included many of the details needed to make the network simulate reality and produce valid results. Some examples of the properties for each node are total dollar cost, requirements for various types of workers (construction workers, specialists, special forces, etc), and land requirement (footprint). These properties are available to the user for modification through a default set is provided. This provides the capability to incorporate specific intelligence into small, localized sections of the network.

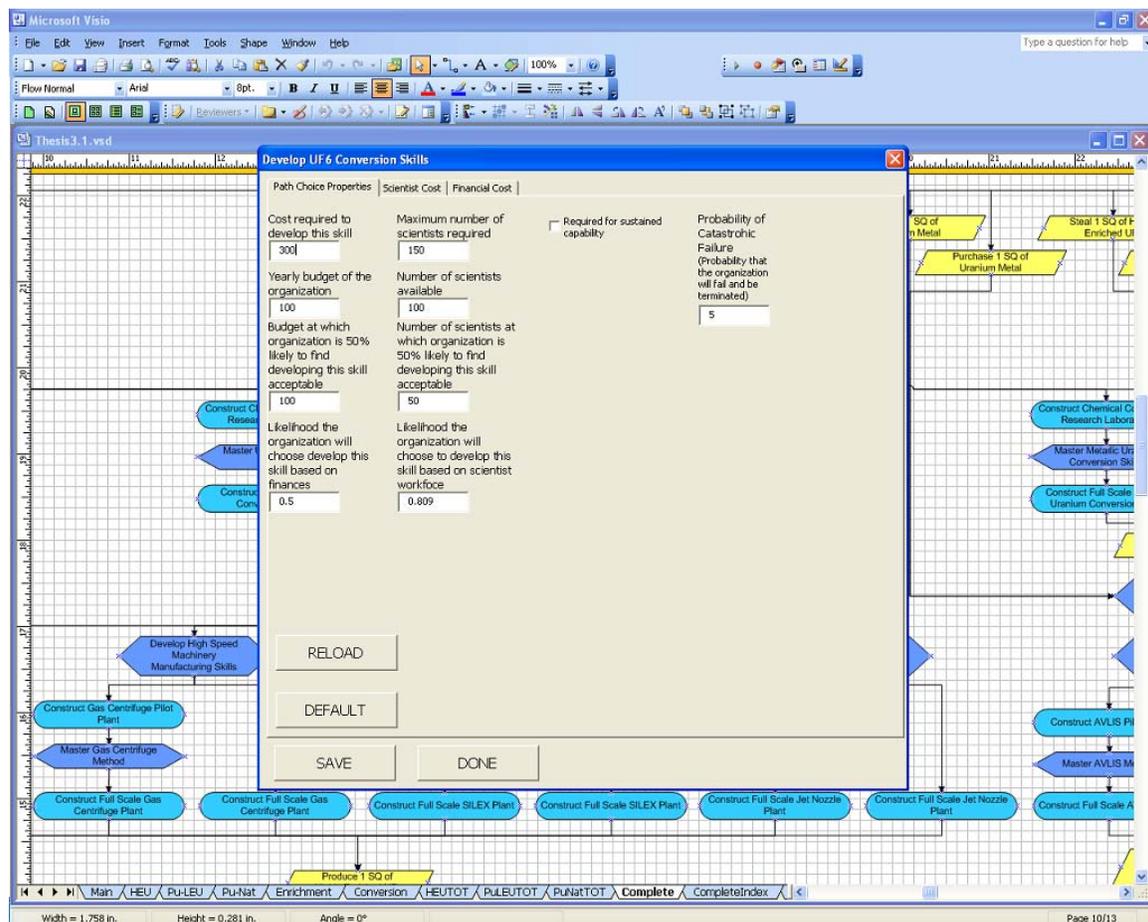


Figure 2. Illustrates part of the network in the background and the individual node features that may be customized in the foreground.

The ways in which each type of node actually uses these resources are somewhat more complicated. The nodes use the organizations available resources to determine two things at two different events in the code:

1. Determine the probability that the organization will find the node in question acceptable. This is done for every type of resource the node requires, and the total probability is simply their product. As was mentioned in the network development section, the probability of any path being chosen is the product of the total probability of the organization finding a node acceptable for every node in that path.
2. Determine the amount of time the organization will require to complete a given node based on its requirements and the resources available to the organization. The resources required have been determined from historic modeling and from expert elicitation. The time involved is important for the analysis because it gives, at least, a ball park figure for how long each node will take and thus how long the entire path will take, but it also directly determines how likely the organization is to fail. Each node has a static probability of catastrophic failure for a 1 week time frame. For every week that a node takes to complete a die is rolled to see if the organization catastrophically failed at that point. If so then that round of the simulation is over, if the organization successfully makes it through every week to completion then they move on to the next node.

## **NETWORK SIMULATION**

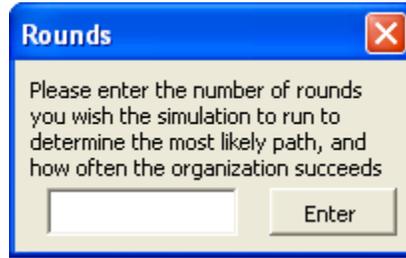
The network simulation determines the most likely path the proliferator will attempt in order to acquire a weapons capability. The network was constructed in Microsoft Visio and the network simulation is performed using Microsoft Visual Basic for Applications which can easily access all the information contained in the network model.

When the user launches the Visio File the code will begin by prompting them to enter all of the relevant information they have on the organization they wish to model. This initial form obtains all of the information the code needs to run its various analyses. The first three tabs are all meant to determine if the organization already has accomplished any of the nodes contained in the network. This will allow the organization to do three things. First, if any of the previously acquired nodes are a material then it is assumed the organization must begin with said material, so every path the model considers will begin with it. Second, if one of the nodes is not a material, the organization will be able to skip either the skill or facility associated with the pre-existing, non-material parameter. Third, the model will automatically weight the paths that contain the pre-existing parameters such that the organization is more likely to choose a path that utilizes them.

The fourth tab is very self-explanatory, and exists solely to give the user the option of stressing plutonium production vs. HEU production. The code does this simply by weighting all of the appropriate paths. The fifth tab, as described earlier, gives the methodology all of the information it needs to determine how the organization will perceive and complete any node it does not begin the simulation with.

The last piece of information the code prompts the user for is the number of rounds. This number is used to specify how many times the code will “roll the die” to determine which path the organization chooses to attempt. This number directly effects how long the code will take to run, if only a few rounds are specified then the code will

run in minutes, if 100,000 rounds are specified the code could take the better part of three days to complete.



**Figure 3. Form the user will see when prompted for the number of rounds.**

Once the user has entered all of the data required for the program to run they will be prompted to alter any of the nodes as they see fit before beginning the analysis. By double clicking any node of the network the user will have access to all of the information contained in any node. While the user will never be able to change how the program utilizes the parameters within each node, they will be able to change the values of every node. A simple form will be present during this phase so that when the user has completed browsing and/or altering the network they can signal the program to begin the analysis.

### **Assigning Path Probabilities**

Since the probability of any path being chosen is a function of the probabilities of each node within the path, it is necessary to explain how the details of the probabilities of each node are determined. In the previous section on Node Development all of the resources required for each type of node were outlined and explained in detail. Those same resources are used when determining the probabilities associated with each node. It should be noted that while each node will have an effective probability of being chosen assigned to it, the number stored in the node itself is only meaningful relative to the number stored in every other node. This is because once the code calculates this number for all the nodes, it will assign a similar number to every path equal to the product of the numbers stored in each node within said path. Once each path has a number calculated for it their sum will be determined and each path will then be assigned its probability based on its original numbers fraction of the total sum. Once again, all of the original data is stored within the nodes, and is manipulated to calculate how it affects the network as whole.

The probabilities contained within the nodes were designed in such a way that they can be easily manipulated by the user, should they see fit. All of the nodes have either two or three resources that are required for both completion and whose details determine their probability of being found “acceptable”. The code uses a simple equation to determine the probability that the organization will choose the node in question based on any given resource.

$$P = \frac{\exp(\alpha x) - 1}{\exp(\alpha x_{max}) - 1} \quad (1)$$

where  $\alpha$  is constant,  $x$  is the amount of resource available, and  $x_{max}$  is the maximum amount of the resource allowed. The user is allowed to directly vary  $x$  and  $x_{max}$  through a

form and may indirectly vary  $\alpha$  by adjusting where the organization is likely to find the node 50% acceptable based on the resource of interest. The code solves for alpha using a simple iteration method, as alpha cannot be solved for directly.

As has been stated above, the numbers that appear in the likelihood boxes inside every node are what determine how attractive the organization finds it, and therefore any path that the node is a part of. These numbers are therefore very important in the methodology that determines which path the organization will take. As it stands the resources at which the organization is 50% likely to find this node acceptable based on each resource required will be set at 1/3 the actual amount of resources required based on the research done. This is to ensure that organizations will still find the nodes reasonably attractive even if it may take them slightly longer to produce the node, and if they have capabilities in excess of what is required then the probability that they will find the node attractive will be at or close to 1. Once again, in the event that the user thinks the numbers assigned to the nodes are incorrect, they are free to change them. It should also be noted again that the facility nodes are always capable of making a paths probability of being chosen zero. If the organization does not have the land available to house all of the facilities contained in a path, then that paths probability of being chosen will be zero, regardless of the attractiveness of any of the other nodes on the path.

## **Choosing the Paths**

Concerning the paths available to the organization, at this point the code has done 3 things:

1. Assigned likelihoods to each path
2. Taken the sum of all the likelihoods of all available paths
3. Assigned a probability of being chosen to each path based on it's likelihood / sum of all likelihoods

Now the code sees, based on the above calculations, each path as occupying a region of numbers between 0 and 1, the width of the region a path occupies determines how likely it is to be chosen (i.e. the most likely paths occupy the widest regions). The code now simply picks a random number between 0 and 1, and whichever paths region it fall in is chosen and stored in a matrix for later analysis. The program will repeat this process until it has chosen a number of paths equal to the number of rounds the user entered earlier.

Once the code is finished choosing the paths that will be analyzed it goes back over the matrix it just created to tally how many times each path was chosen. This is done for two main reasons, to provide the user with the most traversed path, and to do a basic statistics check. The code already knows how many times each path should have been chosen based on the previous analysis and the number of rounds the user determined the code should run, and as long as the actual number of times each path was chosen is relatively close to the theoretical number, the results can be trusted. These results will all be displayed to the user once the code is completely finished running its analyses.

## **Results Displayed**

The results that will be displayed are many and varied. The code is capable of directly giving the user the following information:

1. How many times the organization succeeded at producing a pit of SNM
2. How many times the organization failed
3. The Nodes that were attempted most often by the organization
4. The Paths that were attempted most often by the organization that succeeded
5. The paths that were attempted most often by the organization including both successes and failures
6. The entire path for every attempt the organization made (including where they failed if unsuccessful)

All of this information will be available to the user on-screen as soon as the code is finished running as well as being stored in text files for future reference. What these results will allow the analyst to see is how a unique set of resources will affect the most likely paths an organization is going to take along with its chance of success at completing those paths. This feature can not be underestimated as never before to this author's knowledge has this automated ability been available. As stated above many analysts are capable of determining what resources are required for obtaining a specific skill set, facility, or material, but how those resources affect the series of choices that must be made when developing an entire program for obtaining SNM quickly becomes a daunting problem when one considers just how many pieces must be considered. This code and the results it provides give the analyst the ability to see how a resource set will affect an entire weapons program, rather than just 1 piece of it at a time.

## **CONCLUSIONS**

The results of this work show that it is possible to determine how an organization is most likely to behave given an accurate description of the resources available to them. The code is capable of performing a rational assessment of which pathways would suite the organization best given the resources available to them. This analysis will hopefully provide an intelligence expert with more knowledge and a better understanding of the problem than was previously available. While this code was never intended to replace the human analyst, it does succeed at ensuring that he will have more information available to him when making his decisions than before.

## **ACKNOWLEDGEMENTS**

We would like to acknowledge Brian Bluhm, Ariane Eberhardt, and Lance Eaton of Los Alamos National Laboratory for their assistance in developing this tool. The work done by Corey Freeman of Texas A&M University in obtaining open source information on the resources required for the nodes was also valuable. This work was supported by Los Alamos National Laboratory and the Nuclear Security Science and Policy Institute at Texas A&M University.