# SYSTEM PERFORMANCE AND LAYERED ANALYSIS TOOL

John C. Hyland
Cheryl M. Smith

Naval Surface Warfare Center-Panama City Division
110 Vernon Ave.
Panama City, FL  32407-5000

## ABSTRACT

Naval Surface Warfare Center Panama City Division (NSWC-PCD) has developed a System Performance and Layered Analysis Tool (SPLAT) using MATLAB.  The overall goal is to detect terrorist threats, particularly in an open crowded area, in a timely manner.  Given a sensor configuration and a scenario specification, it combines a layered set of threat detection sensors to determine overall system performance in terms of probability of detection, probability of false alarm, and cost. SPLAT avoids overly optimistic performance estimates inherent when a series of closely spaced detection events are modeled as discrete, independent Bernoulli trials.  SPLAT describes all sensors using multi-dimensional lookup tables, thereby circumventing the need to mathematically model complex sensor performance functions. This methodology is sufficiently general that it can be applied to a broad class of problems where multiple stationary sensors attempt to detect a moving target.

## 1    INTRODUCTION

Throughout the world terrorism shows no signs of diminishing.  Significant terrorist attacks over the last decade have ranged from coordinated team efforts like the 9-11, Twin Towers attacks and the 2008 Mumbai, India attacks; to multiple, synchronized single-terrorist cells like the London subway bombing and the Tokyo subway Sarin gas attack; to lone wolf attacks like the Fort Hood shooter.  Reaction to this threat has led to increased deployment of threat detection systems at airports (Chamberlain 2002; Duncan 2002) and densely populated events as well accelerated research and development of new threat detection systems.

Potential terrorists have a myriad of weapons to choose such as knives, handguns, small assault weapons (Tuttle and Stohl 2009), explosives, dirty bombs (Reel 2002), etc.  In general, particular threat detection systems focus on detecting a specific weapon characteristic.  For instance, a munitions or explosives detection sensor would detect trace explosive chemicals while a radiation detection system would detect radioactive material.  Unfortunately, no one system will detect every possible weapon.  However, combining multiple detection sensors to cover a broad spectrum of weapons should greatly improve the overall probability of detecting a weapon on a terrorist.  The goal of such a system is the timely identification of potential terrorist threats, particularly in an open crowded area.

Preliminary research shows that combining detection sensors into a layered system-of-systems looks promising (NRC 2004).  Figure 1 illustrates the concept with a three-layer detection system concept complete with hypothetical sensor coverage patterns, a likely threat, and a threat trajectory through the layers. Note that the depicted sensors do not represent any particular sensors.  They merely represent a broad range of coverage patterns, automated sensors and man-in-the-loop sensors.  Here Layer 1 acts as a long-range cueing sensor that detects suspicious targets and designates them for tracking and further scrutiny by subsequent layers.  Layer 2 then tracks the designated targets and further scans them for biometric and

physical features indicating that they may be carrying weapons. All suspicious targets identified by Layer 2 are then passed off to Layer 3 which either confirms or denies the presence of weapons on the target. This particular approach makes sense when Layer 1 has a scan rate fast enough to scrutinize all of the candidate targets and cull them for subsequent layers with potentially slower scan rates. Of course, selecting the specific fusion rules to combine detection information within and across layers depends on specific sensor scan rates. And culling candidate targets may not be necessary when all sensors have high scan rates.
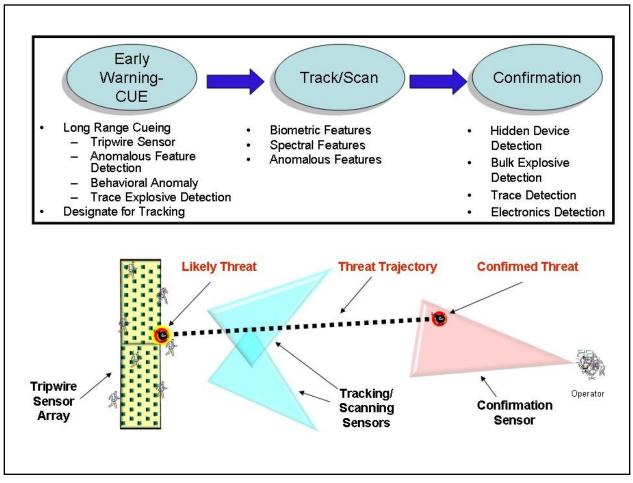


Figure 1: Multi-Layered Target Detection System Concept

NSWC-PCD has developed SPLAT using MATLAB to evaluate candidate sensor and layer configurations. Given a configuration, it calculates performance in terms of probability of detection, probability of false alarm, and cost. The tool will be used to help select detection sensors and to compare various sensor and layer configurations. SPLAT describes all sensors using multi-dimensional lookup tables, thereby circumventing the need to mathematically model complex sensor performance functions as part of the analysis. As long as a sensor performance can be characterized using as lookup table, SPLAT can evaluate the sensor. The lookup tables can be determined from experimentation, agent-based methods, high level simulation, data farming, or highly complex physics-based models. Even hypothetical lookup tables can be used. Utilizing this generalized sensor performance description gives SPLAT tremendous flexibility in that it can evaluate detection sensors at any stage of development.

## 2    TECHNICAL CHALLENGES

The purpose of SPLAT is to analyze candidate threat detection sensor configurations and to quantify their performances in terms of probability of detection, probability of false alarm and cost. These configurations will be designed to provide the timely identification of potential terrorist threats operating in an open crowded area. The performance evaluations will help dictate final sensor selection and the overall system layer configuration. However, SPLAT must overcome significant technical challenges before any analysis can take place.

First, detailed sensor performance for many of the candidate sensors does not exist. Second, sensor performances will be a function of numerous target (terrorist), weapon and environmental conditions; in general, we may not know all of these variables. Third, the number and range of sensor variables makes evaluating sensor performance over all possible variable mixture levels virtually impossible. Fourth, SPLAT needs to be capable of evaluating different sensor deployment patterns like number of and type of sensors at each layer and their precise deployment locations. And finally, SPLAT must account for non-independent sensor detection opportunities.

## 3    LAYER GEOMETRY AND LAYER INDEPEDENCE

### 3.1    Layer Cuboid Structure and Threat Movement through Layers

A cuboid in three-dimensional Cartesian space encompasses each layer. The cuboid is subdivided into voxels which are square in the *x-y* plane and are 2-feet by 2-feet by 1-foot. Note that a fundamental modeling assumption is that the voxels must be both sufficiently large (in the *x-y* plane) to encompass a target and sufficiently small that sensor performance does not vary significantly across adjacent voxels.

Note that all the performance analysis calculations take place at the voxel level. Furthermore, the analysis assumes that all target trajectories progress across the layer from $x = 0$ to the maximum *x*-value which occurs at the layer boundary. In other words, targets may not move backwards. In practice, a target moving backwards would merely create a longer target trajectory thereby presenting more opportunities for the sensors to detect a target. Although SPLAT cannot directly model this backwards motion, the SPLAT probability calculations would still work. However, since they would be based on a shorter trajectory, they would conservatively underestimate the actual probabilities of detection.

### 3.2    Detection Opportunities and Independent Layers

NSWC-PCD's extensive experience with mine-hunting systems has demonstrated that the typical approach of modeling detection opportunities as independent Bernoulli trials tends to be overly optimistic. This unrealistically inflates overall probability of detection. For instance, consider a potential threat standing still in a voxel and let Sensor 1 have a probability of detection for that voxel equal to 0.5. Furthermore, let the entire detection process for Sensor 1 take two seconds. The sensor, then, would have 10 detection opportunities if the threat were stationary for twenty seconds. Under the independent Bernoulli trial assumption the overall probability of detecting a threat at least once over twenty seconds would be $1.0 - (0.5)^{10}$ which equals 0.99902. As the time the target stays in the voxel increases, the probability of at least one detection would asymptotically approach 1.0. This analysis is inherently flawed in that each detection opportunity is not independent from each other.

In general, if a detection model for a sensor were capable of including each and every variable that affected its performance, the outcome of a detection opportunity would be not be random at all; it would be completely deterministic. In practice, however, we can only capture a subset of all of the variables that affect sensor performance. Furthermore, we cannot measure this subset with absolute precision. Many variable measurements include random noise. Given an imprecise measurement subset, however, we can still measure a sensor's performance for a random target. Repeating this process over and over is a standard methodology to determine a sensor's probability of detection for a truly random target. However,

the idea of a random target explicitly means that both the excluded variables and any random noise components on the variable measurements must be truly random in nature.

The preceding example analysis failed to capture the point that although the excluded variables were in fact random for the first detection opportunity, they were no longer random for any subsequent detection opportunity. In fact, there was likely very little change in the excluded variables from one detection opportunity to the next since the time interval was only two seconds. Whether or not the random noise on the measured variables is truly random from one detection opportunity to the next depends on the nature of the measurements and how these measurements are taken. Therefore, as a target moves through a layer presenting detection opportunities, many of the variables (both included and excluded) affecting sensor performance are highly correlated.

SPLAT analysis employs the conservative approach of using the maximum probability of detection along the target trajectory through a layer as a single, discrete probability of detection opportunity. Although this nonlinear approach makes the mathematical analysis extremely complex, it provides a solution without knowing joint sensor probability of detection. And even though it underestimates probability of detection, it does not over estimate it. It is a conservative and practical solution that accounts for correlated detection opportunities.

SPLAT also assumes that for a random candidate target moving along a random trajectory, the detection process for sensors in each detection layer operate independently from sensors in other detection layers. For instance, consider a target carrying a loaded pistol; Magnetic Sensor 1 in Layer 1 detects magnetic objects while Chemical Sensor 2 in Layer 2 detects gunpowder. Clearly, the underlying physics involved in magnetic and chemical sensing make Sensor 1 and Sensor 2 independent. However, note that in the Figure 1 example, Layer 1 sensors identify specific targets for tracking and scanning by subsequent layers. This scenario creates a conditional target detection dependence between layers; such targets are no longer random and in general, under these conditions sensor independence is no longer valid.

## 4 VARIABLE OVERLOAD AND MAPPING SENSOR PERFORMANCE

### 4.1 SPLAT Coordinate System Variables

For flexibility, SPLAT employs a general sensor approach in which all sensor performances consist of multi-dimensional lookup tables as previously described. At a minimum, these lookup tables must include $x$, $y$, and $z$ so the probability of detection can be calculated at the voxel level. In addition to the $x$, $y$, and $z$ indices, when necessary, the SPLAT coordinate system also accommodates $\theta$, orientation to target. This enables SPLAT to properly account for a sensor's angular position relative to the threat. And with the addition of scenario variable $\beta$, position of weapon on the target, SPLAT can analyze a target carrying a hand gun in their front pocket differently than a target carrying a hand gun in their back pocket. The presence of any such sensor angular performance variation is sensor specific. In summary, the SPLAT coordinate system includes four variables: $x$, $y$, $z$, and $\theta$.

### 4.2 Other Inputs and Variable Overload

Numerous other inputs affect SPLAT. These inputs describe environment, threat, atmospheric conditions, weapon type, threat movement, layer sizes and configurations, and fusion rules to combine the layer detections. And besides the four SPLAT coordinate system variables, each sensor will have its own lookup table variables. For example, the lookup table for an optical sensor might contain illumination while the lookup table for an imaging sensor may contain weapon cross-sectional area. Figure 2 depicts the situation. The massive number of potential variables feeding into SPLAT is potentially overwhelming.

### 4.3 Scenario Probability Density Function

A fundamental problem of having an abundance of input variables is that the analysis usually requires a specific value for each variable before it can calculate system performance. Often, though, the user may
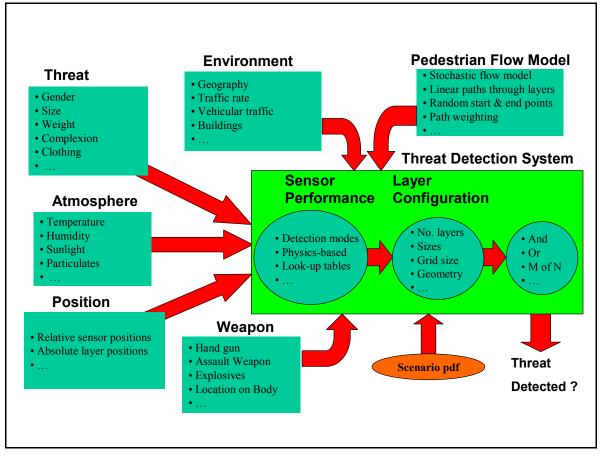
Figure 2: Inputs Affecting SPLAT

not care about system performance against a specific target like white males between 5' 8" and 6' 00" carrying a hand gun in their front pocket; the user just wants to know what is the aggregate system performance against a general threat. SPLAT provides the user with this flexibility.

SPLAT uses a scenario probability density function (pdf) to describe the probability of all possible input variable configurations. For instance, an atmospheric pdf describes how temperature, humidity, sunlight, particulates, etc. jointly vary. Other pdfs capture the relationship between threat characteristics and weapon characteristics. For many input variables the user has the option to choose a specific value, a range of values or to aggregate over that variable using the known scenario pdf.

Aggregate system performance is a useful measure. However, aggregating probability calculations prematurely can lead to disastrous results. Once aggregation removes a variable from the analysis, the analysis can no longer discern how changes in that variable affect performance. For instance, if two sensors were complimentary in nature in that one could detect all threats below 4 feet and the other could detect all threats above 4 feet, aggregating over height before the joint sensor performances were calculated would effectively destroy their complimentary nature. SPLAT has gone to extensive lengths to preserve any complimentary nature between sensors.

## 4.4 Mapping Sensor Performance into SPLAT Coordinates

Figure 3 illustrates how SPLAT maps sensor performance into Layer Coordinates. In this example $Pd(X|V1, V2)$ is the sensor performance as a function of relative position $X$ conditioned on two scenario

variables, *V1* and *V2*.    For this example, *X* represents the (*x, y*) position only.    The scenario indicates that  *V1* = 1.0  and that *V2*  is  unknown.  Therefore,  SPLAT will use the  *Pd(X|V=1.0,  V2*)  conditional
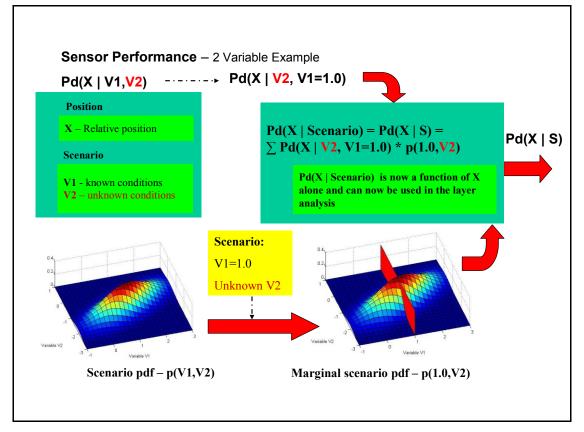


Figure 3:  Calculating Conditional-Marginal Sensor Performance in Sensor Coordinates

sensor performance values.  Since *V1* = 1.0, SPLAT calculates the marginal scenario pdf *p(V1=1.0, V2*). Now  SPLAT  calculates  *Pd*(*X*|*Scenario*),  the  conditioned-aggregate  sensor  performance  conditioned on the scenario by the summing the conditional sensor performance values *Pd*(*X*|*V=1.0, V2*) over all *V2* and weighting each by the marginal scenario pdf *p(V1=1.0, V2*).  Now, *Pd*(*X*|*Scenario*) = *Pd*(*X*|*S*) is only a function of *X*, the relative sensor coordinates (Feller1966).

A simple coordinate transform maps *Pd*(*X*|*S*) to Layer coordinates.  Figure 4 shows a hypothetical plot of sensor performance *Pd*(*L*) after this coordinate transform.     Similarly, *L* represents (*x, y*) position in Layer Coordinates.  To complete calculating expected sensor performance we now need a pedestrian flow model that probabilistically describes movement through the layer.  For this example, we will gener-ate a simple pedestrian flow pdf by enumerating all possible straight-line paths through the layer.  For ex-ample, the blue lines on the *x-y* grid located in the bottom left quadrant of Figure 4 show all of the possi-ble paths through the layer given that the starting point is 10.  Continuing with this, we create *p(path(j,k))*, the joint probability of starting at point *j* and ending at point *k*.  Since *p(path(j,k))* is a joint probability density function, reasonable care must be taken to ensure that it behaves like a pdf.

Given a path, SPLAT calculates the maximum *pd* along that path.  Now by forming a weighted sum of these maximum *pds* over all possible paths, SPLAT completes the performance calculation.  The weights are the *p(path(j,k))* values.  The example in Figure 4 shows the last steps in this process as well as a graph of maximum probability of detection given starting point and the overall mean maximum *pd*.

Note that by cleverly transforming and combining probability of detection lookup tables and joint pdfs using conditional probability, marginal probability, and the law of total probability, SPLAT is a dis-

crete, analytic solution to the system performance calculation. SPLAT contains no Monte Carlo analysis or approximations. Effectively, SPLAT is a massive probability calculator.
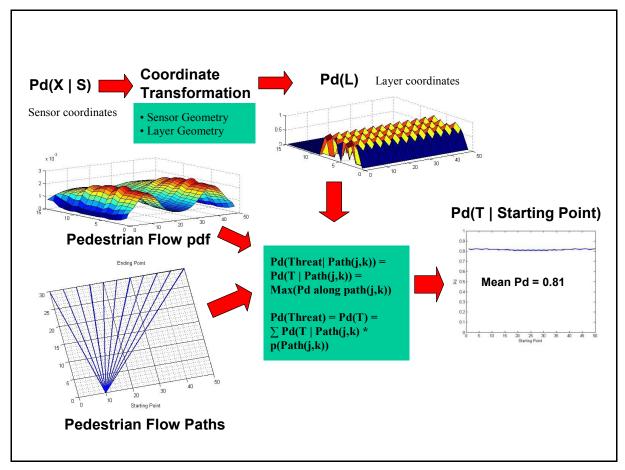


Figure 4: Enumerating Paths and Calculating Overall Performance

## 5    RESULTS

### 5.1    Graphical User Interface (GUI)

Early in the development of SPLAT it became clear that layer design and sensor placement requires some visual feedback to approximate the area coverage for each individual sensor. The large number of environmental and target description variables and options make it difficult to describe the input scenario. Also, the layer areas need gateways, barriers, and defined traffic flow patterns. Hence, a graphical user interface was developed to facilitate the design of the detection layers and the choice of scenario variables. The GUI also provides an interactive environment to run SPLAT with step-by-step visual feedback for the quality of the resultant detection layer.

### 5.2    Scenario Definition

Figure 5 shows the scenario definition screen. Scenario variables are selected from list-boxes that allow one or more choices in each category. The size of each layer is defined in an editable table. The general commands in the left-hand column allow the user to navigate through the program and to save his results.

The scenario variables are a function of the available sensors. All variables in a sensor model must be included in the scenario. Some extra variables are included to help the user describe the targets of interest

such as gender, height, weapon, location, etc. If the user either has no information about a variable, or would just rather ignore it, the results will be aggregated over all values of that variable.



Figure 5: Scenario Input Screen

Once the scenario is defined, the sensor performance for each available sensor against the described targets under the defined conditions is calculated. Thus, the sensor performance goes from being a function of the target type, position, orientation, and the environment to just a function of the target's position and orientation.

## 5.3    Layer and Sensor Definition

Figure 6 shows the sensor selection screen. Sensors are added by clicking on the layer axes in the center of the screen. Slider bars at the bottom of the sensor placement panel adjust the pitch and height of the sensor. Beside the sensor selection buttons, the user may scale the performance of each sensor type. The figure shows that the bottom layer is covered by six sensors; the two sensors on the lower corners on the images have a rectangular coverage pattern (in green) while the other four sensors have flashlight coverage patterns. The images on the right show the resultant probability of detection per cell at a specified height. Probability is scaled using a gray scale. Red blocks show the positions of barriers. And the red path drawn on the images shows the path of minimal detection probability. That is, it is the best path that a knowledgeable target could take to avoid detection.

## 5.4    Path Flow Models

The flow of pedestrian traffic through the layer determines whether or not a subject will enter into a particular sensor's field of view. Given that a subject comes into the sensor's detection area, his path deter-

mines his range to the sensor and his orientation relative to the sensor. All these factors can influence the probability of detection.
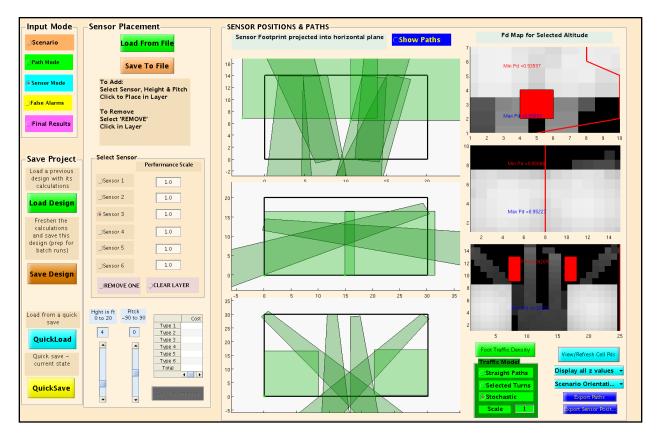


Figure 6: Sensor Placement Screen

To calculate the expected performance for a given layer design, SPLAT has multiple path flow models available. The straight-line path model, in which subjects enter the layer and follow a straight-line path to an exit, provides a lower bound for the detection rate based on random behavior. Subjects spend a minimal amount of time in the layer and have a fixed heading. This model executes quickly and allows for the enumeration of all paths, calculating the maximum probability of detection over each path individually. Unfortunately, in the presence of barriers many paths must be eliminated. And sometimes there is no route to an exit at all. To solve this problem, the limited turns model allows turns at specific layer row locations. Although all possible paths are still enumerated, the computation time increases from on the order of $K^2$ to $K^{N+2}$ where $N$ is the number of turn layers. Also, this solution is difficult to use as the choice of the turning point rows can greatly modify the outcome.

A stochastic traffic flow model using a modified Brownian motion provides the desired flow around barriers and through the layers. Probability distributions for the start and end points are determined by the gate positions. The path between each start and end is modeled using a Brownian bridge (Karlin and Taylor, 1981). Rescaling the Gaussian distribution accommodates movement around barriers. Figure 7 shows the selected layer design and the resultant path flow using the stochastic model. Here the green areas indicate gates and the red areas indicate barriers. The blue areas indicate high density foot traffic areas. The lower layer has no gates so traffic can enter and leave anywhere along the layer boundaries.

The gray-scale foot traffic density plots towards the right side of Figure 7 confirm the gate and barrier flow restrictions and demonstrate smooth traffic flow around the barriers. Additionally, the stochastic

model has been designed to accept a one-step Markov transition matrix to describe movement through the layer if a more sophisticated traffic model is available.
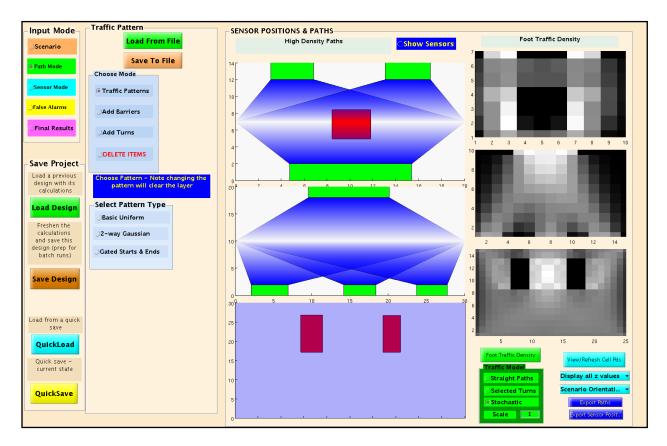


Figure 7:  Path Flow Definition Screen

## 5.5    Final Calculation

The goals of the system analysis are to 1) quantify the joint performance of the sensors as deployed in the layer configuration under the selected conditions and, 2) build detection rules to combine the sensor outputs from each layer to balance the required detection rate with an acceptable false alarm rate.  The final calculation from SPLAT provides the joint performance of the sensors in each layer.  It uses the sensor performance tables modified for the current scenario, the sensor positions in each layer, and the subject positions and orientations from the flow model to calculate the maximum expected probability of detection (*pd*) for a target passing through the layer for each sensor individually and combined with other sensors in the same layer.  The combined sensor performances include the joint expected performance for all sensor type pairs and all triples.  Figure 8 shows that for Layer 1, the expected maximum *pd*s are 0.7260 for Sensor 1, 0.9100 for Sensor 2, and 0.6611 for Sensors 1 and 3 jointly.

The performance table shown in Figure 8 provides all the values required to calculate the resultant overall performance from any desired combination of sensors.  Currently only values for probability of detection are displayed.  False alarm models for various sensors are under development (Hall,1992).

Note that by specifying a layer configuration and sensor positions, the user has created a spatial correlation between the sensors.  SPLAT's probability analysis carefully preserves these correlations.

Error bounds for the estimated *pd* for a given sensor configuration are provided by the minimum and maximum *pd* values displayed on the *Pd* Map images.  These account for errors due to the path modeling.  Sensor performance error analysis will be included as it becomes available.
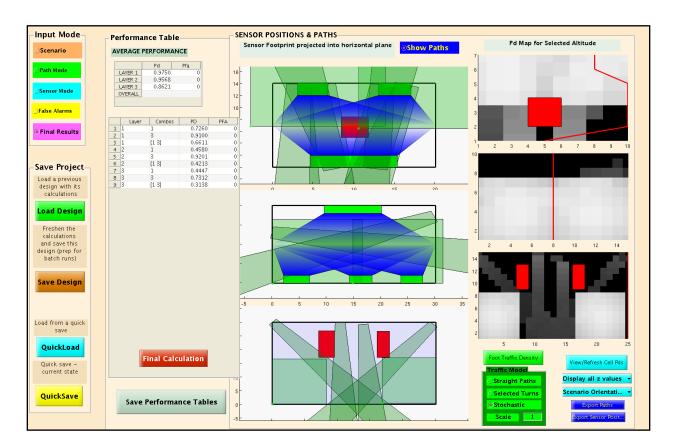
Figure 8: Performance Table

# 6    CONCLUSIONS AND FUTURE RESEARCH

## 6.1    Conclusions

Combining multiple threat detection sensors into a layered configuration should create a system capable of quickly identifying potential terrorist threats. The overall goal is to detect such threats, particularly in an open crowded area, in a timely manner. Naval Surface Warfare Center Panama City Division has developed a System Performance and Layered Analysis Tool using MATLAB. Given a sensor configuration, it calculates overall performance in terms of probability of detection, probability of false alarm, and cost. SPLAT describes all sensors using multi-dimensional lookup tables, thereby circumventing the need to mathematically model complex sensor performance functions. The lookup tables can be determined from experimentation, agent-based methods, high level simulation, data farming, or highly complex physics-based models. Even hypothetical lookup tables can be used. Utilizing this generalized sensor performance description gives SPLAT tremendous flexibility in that it can evaluate detection sensors at any stage of development.

SPLAT cleverly transforms and combines the probability of detection of lookup tables and joint variable probability density functions using marginal probability, conditional probability and the law of total probability; this creates a discrete, analytic solution to overall system performance. Furthermore, SPLAT avoids overly optimistic performance estimates inherent when detection events are modeled as discrete, independent Bernoulli trials. Instead, SPLAT uses the maximum probability of detection over a path as a

single detection opportunity, thereby capturing the fact that a series of closely spaced detection opportunities are usually highly correlated.

Additionally, SPLAT contains a powerful and flexible graphical user interface (GUI) that facilitates layer design, sensor placement and scenario definition. Furthermore, it permits detailed pedestrian traffic flow patterns through the layers which can include gated entry and exit areas, barriers, linear paths, piecewise liner paths, and Markov model paths.

The SPLAT methodology can be applied to the general problem of detecting a moving target with multiple stationary sensors. The only requirement is that the sensors be adequately described using the required lookup table format.

## 6.2    Future Research

Currently, SPLAT contains a small number of threat detection sensors. As more candidate sensors are selected for considerations, their performances will have to be characterized in the required lookup table format. And a reasonable plan to characterize them over represented levels of all significant variables will have to be developed.

Additional future research will include: development of false alarm models; development of cost approximations; implementation of fusion rules both within layers and between layers; implementing target tracking models; and developing sub-optimal automated sensor placement strategies.

## REFERENCES

Chamberlain, D. 2002. "National Security and Interception Procedures," *FAA NEWS*, January/February 2002

Duncan, P. A. 2002. "FAA Advises Travelers on Air Travelers on Airport/Airline Security Measures," *FAA NEWS*, January/February 2002

Tuttle, D. and R. Stohl. 2009. "Mumbai Attacks Demonstrate Terrorist Reliance on Small Arms," Center for Defense Information, Washington, DC. Accessed March 8, 2011. http://www.cdi.org/friendlyversion/printversion.cfm?documentID=4442

Reel, M. 2002. "Imagining A 'Dirty Bomb,'" *Washington Post*, June 12, 2002.

National Research Council of the National Academies. 2004. *Existing and Potential Standoff Explosives Detection Techniques*. Washington DC: The National Academies Press.

Hyland, J. C., C.M. Smith and J.R. Stack. 2009. "Multi-Sensor Coverage for Aspect Dependent Targets," In *Proceedings of IEEE CIAA 2009 Conference,* Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Karlin, S. and H.M. Taylor. 1981. *A Second Course in Stochastic Processes*. Orlando, Florida: Academic Press.

Hall, D.L. 1992. *Mathematical Techniques in Multisensor Data Fusion*. Norwood, MA: Artech House, Inc.

Feller, W. 1966. *An Introduction to Probability Theory and Its Applications, Vol. 2*. New York, NY: John Wiley and Sons, Inc.