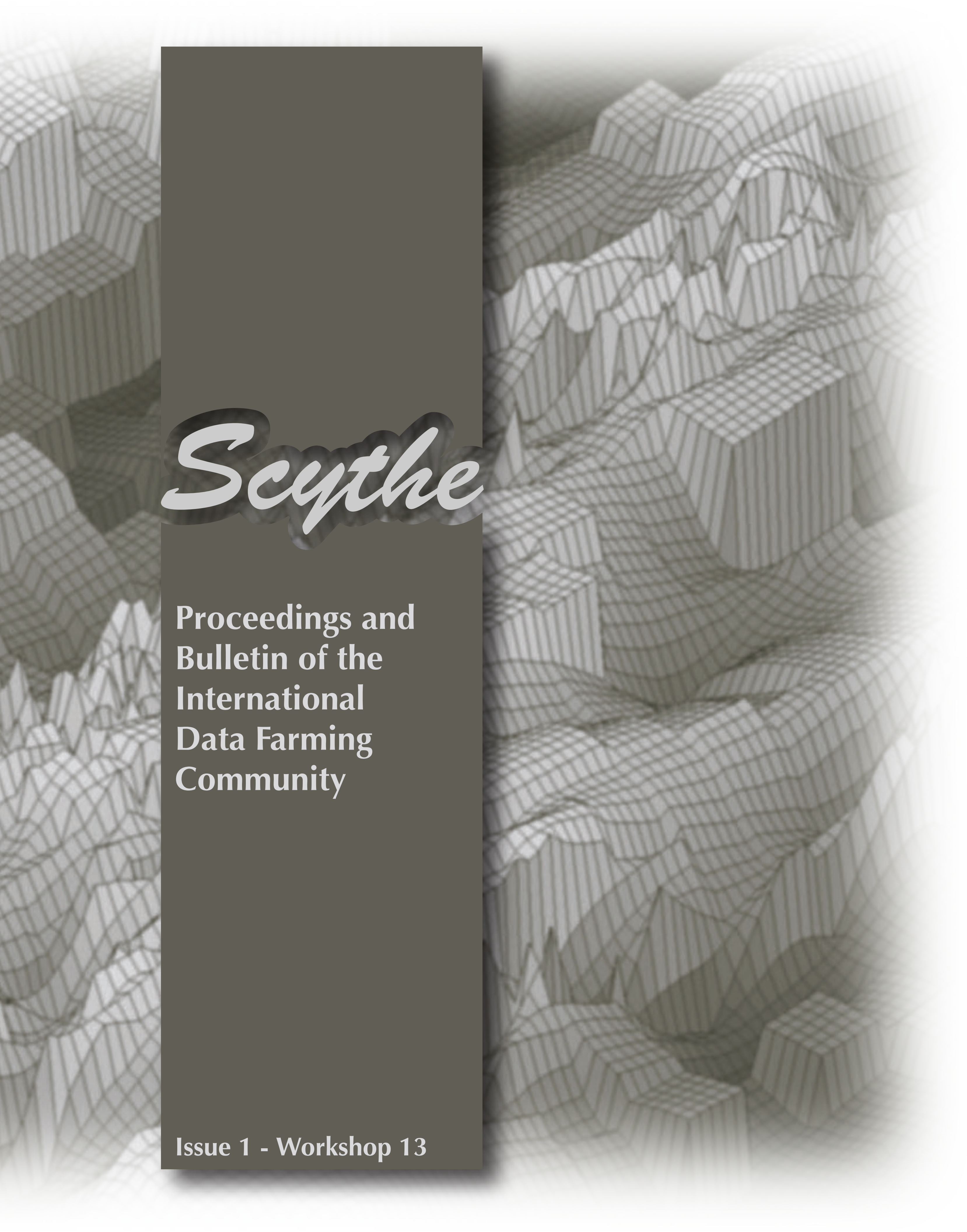


Scythe

Proceedings and Bulletin of the International Data Farming Community

Issue 1 - Workshop 13

Proceedings and Bulletin of the International Data Farming Community

Table of Contents

Scythe

Proceedings and Bulletin of the International Data Farming Community

It is appropriate that the first publication supporting the International Data Farming Workshop is named after a farming implement. In farming, a scythe is used to clear and harvest. We hope that the "Scythe" will perform a similar role for our *data* farming community by being a tool to help prepare for our data farming efforts and harvest the results. The Scythe is provided to all attendees of the Workshops. Electronic copies may be obtained from harvest.nps.edu Please contact the editors for additional paper copies.

Please let us know what you think of this experimental first issue. Articles, ideas for articles and material, and any commentary are always appreciated.

Bulletin Editors

Ted Meyer................. tedmeyer@mac.com
Gary Horne...... datafarming@verizon.net

International Data Farming Community
Workshop 13 Program Committee

Klaus-Peter Schwierz
klaus-peter.schwierz@eads.com .... Germany
Mink Spaans
mink.spaans@tno.nl .... Netherlands
Felix Martinez
felixtiburcio@hotmail.com .... Mexico
Dave Ang
achoonki@dso.org.sg .... Singapore
Ole Jakob Sendstad
ole-jakob.sendstad@ffi.no .... Norway
Gary Horne
datafarming@verizon.net .... United States
Susan Sanchez
ssanchez@nps.edu......... Naval Postgraduate School, USA
Tom Lucas
twlucas@nps.edu........... Naval Postgraduate School, USA

International Data Farming Community
Overview

The International Data Farming Community is a consortium of researchers interested in the study of *Data Farming*, its methodologies, applications, tools, and evolution.

The primary venue for the Community is the bi-annual International Data Farming Workshops, where researchers participate in team-oriented model development, experimental design, and analysis using high performance computing resources... that is, Data Farming.

Scythe, *Proceedings and Bulletin of the International Data Farming Community,* Issue 1, Workshop 13
Publication date: January 2007

The Harvest from International Data Farming Workshop 13

What Was In the Black Bag

International Data Farming Workshop 13 took place from 12-17 November 2006 in The Netherlands. Thirty-six participants from six countries worked in nine different teams exploring questions using Data Farming methods. Of course, the numbers themselves can't quite quantify the significance of the workshop. Workshop 13 was actually the first not "sponsored" by Project Albert, which ended in September. But many of the ideas and methods developed during Project Albert are being carried forth as we move into the future. And our first and foremost characteristic remains central as our work continues to be driven by our Questions and our desire to support decision makers who need the answers to those questions in the future. I believe Workshop 13, with the theme "Harvesting Understanding," was a great start to that future and this publication shows the results of our Data Faming efforts. Many thanks to the team leaders and participants in IDFW 13 for the work they have accomplished, which I will briefly summarize here and will also be described in detail in this publication, The Scythe.

Team 1 at International Data Farming Workshop 13 used the simulation system "PAX" to gain insight into specific aspects of peace support operations. The model PAX was developed by EADS under contract of the German Bundeswehr to model peace support operations with the focus on individual civilians and emerging group behavior. The IDFW 13 scenario addresses crowd and riot control operations in a post civil-war city. simulated and analyzed. Using PAX in the Data Farming Environment, the team was able to issue some recommendations on the behavior of the military and police forces to avoid escalation.

The German Federal Office of Defense Technology and Procurement has been analyzing the influence of networked sensors and effectors on military capabilities. **Team 2** used a convoy protection scenario in MANA as part of an overall exploration of Peace Support Operations in an urban environment. The technical effects of special sensors and effectors were examined. Variations were investigated in the technical representation of UAV/UGV speeds, communication and sensors, in scenario details and in a variation of protection and equipment. The workshop ended

with a convoy scenario that worked fairly well, but with many open questions

Team 3 asked: What effect does the communication aspect have on operations in urban terrain? They used ITSim to begin to evaluate the importance of reliability and delay times in a communication network. Furthermore, the effect of jamming the Red communication for force protection was also investigated. During IDFW13 a vignette was set up and tested within ITSim and based on the finished vignette, data farming experiments will be conducted during IDFW14.

The Command and Control Research Program has sponsored the design and development of a software environment for conducting human-in-the-loop experiments focused on information- and social-domain phenomena. Over the course of several Workshops, a team has strived to create and improve a simulation version of these experiments. At IDFW 13, **Team 4** used NetLogo to study how an organization's structure impacts a group's ability to share information in order to solve a simple cognitive task. Through data farming, they were able to explore the experimental factors postulated to influence the information sharing and shared awareness-building processes.

The intent of **Team 5** was to leverage the agent-based distillation MANA and Data Farming to examine the employment of ground swarm robotics to accomplish tasks such as IED detection. The team expanded the design of experiment from previous wok so as to gain more insights and fidelity on quadratic effects and interactions of various factors. In addition, swarm robots getting trapped by terrain hazards while moving within the area of operations are modeled and insights on the impact of such hazardous terrain on swarm robustness and effectiveness are obtained.

Team 6's work contributed to insight on the following question: What might matter and what doesn't appear to matter in the employment of systems before actual testing takes place? Through modeling and Data Farming of the scenario developed, the goal was to better understand what might be tested. The plan for the workshop was to utilize MANA, Pythagoras, and other models and use data farming methodology to understand the possibility space for important variables in the systems under test. The work at IDFW 13 was intended to be part of a continuing process to assist in systems of systems test planning in particular and in the overall development of a Joint Test and Evaluation Methodology.

The objective of **Team 7** was to determine the performance of various proposed company level urban fighting force structures to open and clear an axis through a built up area. The Auto Red Teaming framework developed by DSO National Laboratories in Singapore was used to identify the key parameters that would affect the outcome of the urban war fighting scenario. The findings from Team 7 at IDFW 13 highlighted some of the key issues for a force to fight in built up areas and can perhaps provide a useful basis for the future studies to be conducted.

The ultimate aim of **Team 8** is to enable the representation of Combat Identification characteristics and behavior within a constructive simulation in order to enable the exploration of the benefit of system interventions based

on Situational Awareness, Target Identification, Human Factors and TTPs in terms of increasing combat effectiveness and reducing fratricide levels. During the workshop, Team 8 was able to refine the process model, focusing on both the specification of the information models in the system and the agents, and on the choice of a general implementation method for the simulation.

Team 9 attacked the question: "Can agent-based simulation provide calibrated command agents for co-adaptive synthetic environment research?" During the workshop, this team developed two scenarios, one in NetLogo and one in Pythagoras to explore the problem space by using a command agent to replicate a human commander's simple battlefield decisions such as retreat, attack, and commit the reserve.

In addition to the 9 teams working throughout IDFW 13, we had many plenary presentations that started with Col Eileen Bjorkman's keynote address. They are summarized here in The Scythe and are included on the CD in the back cover.

Many thanks to our intrepid Dutch hosts for their efforts that allowed IDFW 13 to take place, and in particular thanks to Mink Spaans. It was a challenge to hold this workshop in the transition from Project Albert, but I think all who participated would agree that our hosts enabled us to launch into the future with integrity, audacity, and humility.

Our Data Farming community is again fortunate to have a wonderful venue for our next workshop. International Data Farming Workshop 14 will be held in Monterey, California, USA, the home of the SEED Center for Data Farming at the Naval Postgraduate School. It will start with the opening dinner on Sunday 25 March 2007 and continue through the week with the closing session on Friday 30 March. There will also be an opportunity for question teams to gather the Thursday and Friday of the previous week at the Naval Postgraduate School to get a jump start on the workshop if they desire.

Whatever your plans, please enjoy the remainder of this publication which includes articles on NetLogo, Mana and Pythagoras. Please feel free to contact me at datafarming@verizon.net with questions, comments, or discussion at any time on any topic. Finally, allow me conclude by saying Thank You for a job well done to Ted Meyer for collecting and publishing the work in this issue of The Scythe.

Best Regards,
Gary Horne

"Imagination is more important than knowledge" ~ *A. Einstein*

"All models are wrong, some are useful." ~ *G. Box*

"Anything worth doing is worth doing excessively."

"We must know. We will know." ~ *D. Hilbert*

"The beginning of knowledge is the discovery of something we do not understand." ~ *F. Herbert*

"Now what?"

"It needs more cowbell." ~ C. Walken on SNL

"Pain is inevitable, suffering is optional."

Project Albert

History & Summary of Methods

Project Albert was a congressionally funded modeling and simulation initiative of the United States Marine Corps from 1998 to 2006. The basic idea of Project Albert was to pair simple, efficient, abstract distillation models with high performance computing to do explore large design spaces. The techniques developed within Project Albert fall under the overall methodology known as Data Farming. When distillation models and high performance computing are combined with efficient experimental designs, a huge sample space can be explored very rapidly. And when rapid prototyping capabilities and collaborative environments are introduced into the Data Farming process, progress on questions, even long-standing and difficult questions involving many interacting variables, is possible.

Project Albert used what are referred to as agent-based distillation models. These are a type of computer simulation which attempts to model the critical factors of interest in combat without explicitly modeling all of the physical details. Some of the models used in Project Albert were ISAAC, MANA, PAX, and Pythagoras, all agent-based models, although the methods developed can be applied using any type of simulation model. But agent-based models are small and abstract and can easily be run many times to test a variety of parameter values and get an idea of the landscape of possibilities. The term distillation is added, because the intent is to distill the question at hand down into as simple a representation as possible. Also, models used in Project Albert were specifically developed and used because the capability to rapidly prototype scenarios is very important in the process.

Data Farming combines the rapid prototyping of agent-based distillations with the exploratory power of high performance computing to rapidly generate insight into military questions. Data Farming focuses on a more complete landscape of possible system responses, rather than attempting to pinpoint an answer. This "big picture" solution landscape is an invaluable aid to the decision maker in light of the complex nature of the modern battlespace. And while there is no such thing as an optimal decision in a system where the enemy has a vote, Data Farming allows the decision maker to more fully understand the landscape of possibilities and thereby make more informed decisions. Data Farming also allows for the discovery of outliers that may lead to findings that allow decision makers to no longer be surprised by surprise.

The simulations that defense analyst use are often large and complex. And even the smaller more abstract agent-based distillations referred to above can have many parameters that are potentially significant and that could take on many values. In addition, response surfaces can be highly non-linear. Thus, even with high performance computing and the small models used in Data Farming, gridded designs where every value is simulated are unwieldy. Thus, using efficient experimental designs is essential and work in this area has been performed at the Naval Postgraduate School and other places. Techniques such as nearly orthogonal Latin Hypercubes have been used to examine many variables efficiently and analyze the many effects and interactions that are important to answering the questions at hand.

Throughout the years Project Albert grew to include many entities outside the USMC. In addition to participation by the US Air Force, Army, and Navy, and other US Department of Defense agencies, international participation included Australia, Canada, Germany, Mexico, Netherlands, New Zealand, Norway, Portugal, Singapore, South Korea, Sweden, and the UK. The central theme across all the years, all the nations, and all of the workshops has been the Questions that traditional methods have not been able to answer adequately where Data Farming offers promise. Project Albert ended in September 2006, but of course the many questions that motivated the work continue. And the SEED (Simulation Experiments & Efficient Designs) Center for Data Farming at the Naval Postgraduate School continues as the US governmental home for the methods developed and the application of Data Farming.

Team 1: Peace Support Operations with the PAX Model

TEAM 1 MEMBERS

J. HARTMANN - Lead
A. GRÜNAUER
M. LINDE
Bundeswehr, Deutschland

G. SCHWARZ - Contact[1]
G. WAGNER
EADS, Deutschland

F. MARTINEZ
Mexico

TEAM PROPOSAL

The Peace Support Operations Team at International Data Farming Workshop 13 will use the simulation system "PAX" to gain insight into specific aspects of peace support operations.

The model PAX was developed by EADS under contract of the German Bundeswehr to model peace support operations with the focus on individual civilians and emerging group behavior. The heart of PAX is the modeling of collective aggression on the civilian side. Civilian agents are characterized by several personality factors and internal processes that generate a situation-dependent behavior driven by their motivational and emotional state.

This modeling is persistently based on empirical findings from the psychological research on aggression.

The IDFW 13 scenario addresses CRC[2] operations in a post civil-war city. Within the scenario, different – potentially violent – civilian groups will be modeled and the effects of different approaches of the security forces will be simulated and analyzed.

For IDFW 13, the Peace Support Operations Team has the following goals:

1. Review and face validate the upgrades made to the PSO model PAX between PAIW 12 and IDFW 13, especially the implementation of extended possibilities for the setup of scenarios.
2. Develop and test a potentially violent CRC scenario with different civilian groups. Develop and test two alternative scenario versions with different approaches.
3. Conduct experiments with different designs (both NOLH and gridded.
4. Identify needs for further work.
5. Gain insight into other models (participation in plenary sessions).
6. Provide information about the simulation model PAX (plenary session briefing).

TEAM ACTIVITIES

PAX is able to show dependencies between the soldiers' behavior and the escalation of violence, which may occur between soldiers and civilians as well as between different civilian groups. Furthermore, PAX allows for the investigation of many other measures of effectiveness (MOEs), such as the level of escalation or the number of civilians and / or soldiers who get injured or killed, to give an example.

In advance of IDFW13, LTC Hartmann distributed among the team members a scenario proposal which was taken as a basis for further discussions. The PSO team was instructed to act as an operation analysis cell and to elaborate recommendations for the given problems. To be able to do so, the described situation was modeled and simulated in PAX.

The first day of IDFW13 was used to set up the scenario in PAX. In the following days single simulation runs as well as extensive experiments were accomplished and analyzed.

Scenario Overview

The IDFW 13 scenario is based on the following background events: The newly-elected president of an interim government in a post war country (DANUBIA) just recently introduced his cabinet. However, a part of the population

[1] For more information contact: Gunther Schwarz, gunther.schwarz@eads.com

[2] Crowd and Riot Control

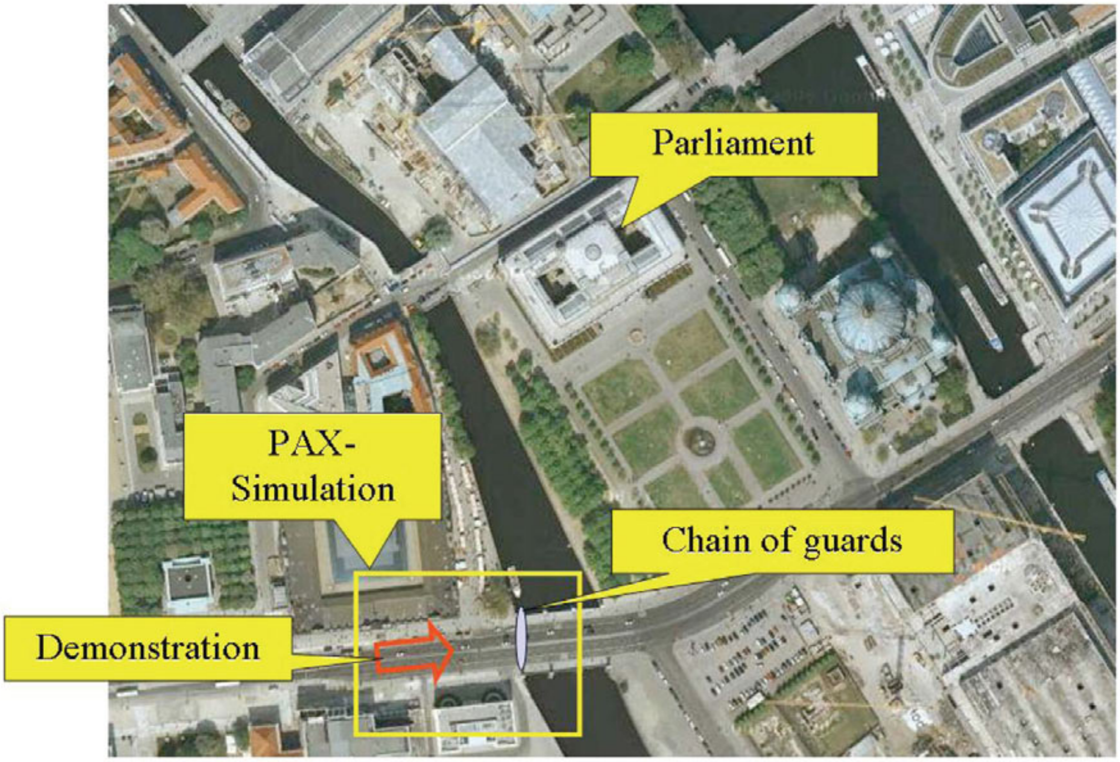


Figure 1: Aerial view of the real-world scenario

(20%), the Paxians, feels disadvantaged and inadequately represented. Only a few days ago, an initially peaceful demonstration of this group led to violent riots during a mass rally of the governing party. Messages of the local news service suggest the possibility of further demonstrations within the governmental district. The security assistance forces DAPFOR are instructed to protect the governmental buildings and the representatives of the Danubian parliament. This task is intended to be transferred to the newly established Danubian police forces. In the concrete situation the question arises whether the DAPFOR on the one hand or the Danubian police forces on the other should undertake the task of blocking the access roads to the governmental district. Simulating the situation with PAX shall help to assess which alternative is best to avoid the evolvement of collective aggression and to prevent the opponents' violence as early as possible. Thus, several objectives are to be pursued: while preventing the demonstrators from passing, the chain of guards own losses are to be minimized and violence should be avoided whenever and wherever possible.

Within the scenario, two different civilian groups are modeled, both unsatisfied with the current political situation and trying to demonstrate in front of the Parliament. The majority group is quite peaceful whereas a small minority will not flinch from using force. The military forces try to hold the civilians back. The effects of various tactics, techniques and procedures of these forces will be simulated and analyzed.

The situation in the whole area is expected to be initially calm, but with potential for escalation due to some

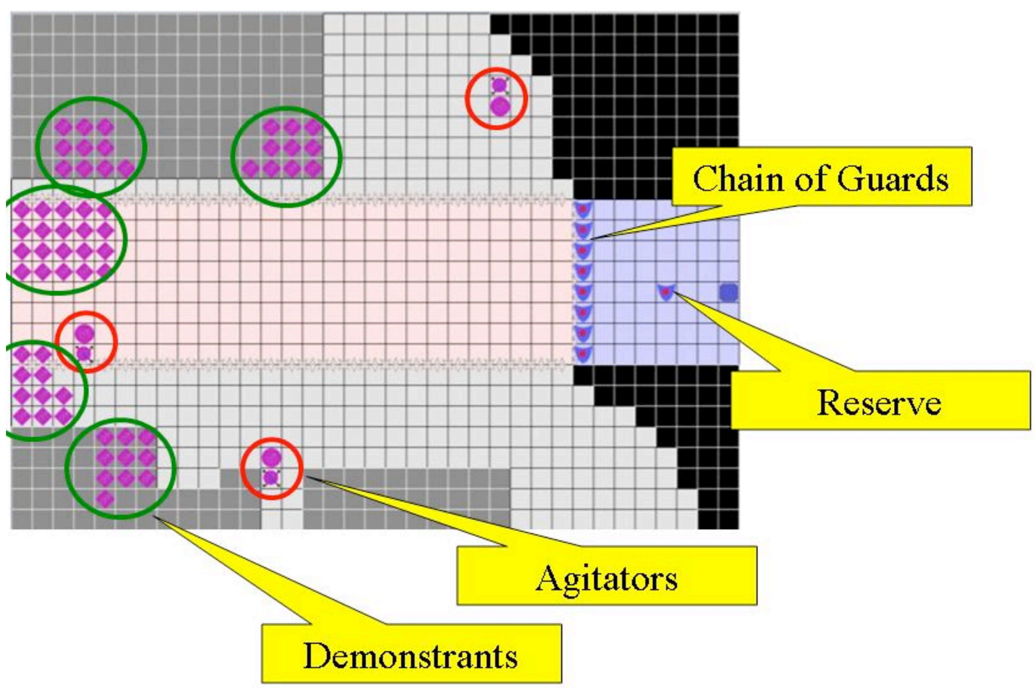


Figure 2: Peace Support Operations Team Basecase Scenario

aggressive agitators in the area and the still present experiences the civilian population has made during the recent war. Thus, the team hopes to get insights into the evolvement of collective aggression throughout the course of the simulation.

Figure 1 shows an aerial view of the real-world scenario.

An equivalent base case scenario developed by the PAX team members is shown in Figure 2. A chain of guards is controlling the access road to the governmental district where civilian demonstrations and counter-actions are expected.

Civilian group "Demonstrators" – identified by a diamond shape – consists of men who want to demonstrate peacefully against the newly established cabinet.

The intention of civilian group "Agitators" is primarily to pick a fight and especially to interfere with the military operation. Members of this group are symbolized by a hexagon and are partly equipped with throwing or impact weapons while the remaining members may throw stones at the utmost. Small groups of civilians of this group may occur during the scenario as a kind of a "planned" attack towards the soldiers.[1]

PAX Setup

During the first day, the PAX team became acquainted with the improved scenario editor of PAX version 2.10, allowing users to create new PAX scenarios in a comparatively short time. The newly designed agent editor was used to create new agent classes for the different civilian and military agent types. Instances of these agent classes could then be used for the concrete scenario setup.

Four alternative scenario versions were developed, investigating

- the deployment of both, the Danubian Peace Forces DAPFOR (case A) and the local Danubian police forces (case B) but also analyzing
- two different Demonstrator-Agitator ratios on the civilian side (case 1: 50 demonstrators and 28 agitators; case 2: 60 demonstrators and 18 agitators).

The rather peaceful Demonstrators were led by one leader and characterized by the following attributes:

- Low values of fear, anger and readiness for aggression
- High willingness for cooperation
- High motivation to demonstrate and to pass the chain of guards
- Only equipped with throwing weapons (e.g. stones)

The Agitators are divided into three smaller groups, each group having its own civilian leader. The Agitators' behavior was determined by the following characteristics:

- Low values of fear and willingness for cooperation
- High values of anger and readiness for aggression
- Medium motivation to demonstrate and to pass the chain of guards
- Equipped with throwing weapons and impact weapons

The two different mission approaches A and B could be modeled by parameterizing the military agents in different ways:

- DAPFOR (A): higher protection and authority, but only passive reserve
- Danubian Police (B): less trained and experienced and more "hated" by the civilians, but active reserve that behaves according to a "Zero Tolerance" rule set.

Using the NOLH Design

Having reviewed the upgrades made to PAX between PAIW12 and IDFW13 (especially regarding the Scenario and Agent Editors) PAX parameters of main interest and presumable importance in the scenario were identified. These 11 parameters were used as input for the Nearly Orthogonal Latin Hypercube (NOLH) design and four different experiments (representing the four possible combinations of cases A, B, 1 and 2) were set up in order to analyze the consequences of the two different mission approaches (A and B) on the overall escalation of violence and on the mission success.

The used NOLH Design considered the following parameter variations:

- Threshold for calling reinforcement by the chain of guards
- Increase of the soldiers' stress level in case of getting wounded
- Threshold for intervention of the police's active reserve
- Agitators' anger, readiness for aggression and dog factor[2]
- Demonstrators' fear, anger, readiness for aggression and dog factor

[1] Modeling detail: Planned actions are not yet built into PAX so that these groups will only be set to appear in the scenario at a predefined point in time, from which on they will act according to their given motivations.

[2] The dog factor describes the soldiers' or police's threatening effect on the civilians. A higher dog factor leads to a higher increase of fear whenever a soldier defends himself against a civilian's attack.

Thus, 11 PAX parameters were examined in more detail, resulting in a total number of 33 design points according to the NOLH design.

To be able to measure the mission success of the different military forces (DAPFOR or Danubian police) the PAX team evaluated the following measures of effectiveness:

- Overall escalation (resulting from aggressive actions performed by civilians and soldiers)
- Escalation performed by different groups
- Number of wounded or killed civilians and soldiers

For each of the 33 parameter constellations 50 replicates were submitted to the German 128 node cluster in Immenstaad. In advance of analyzing the experiments' results, the PAX team members wrote down their expectations regarding the overall aggregated escalation for the two mission approaches (see Figure 3).

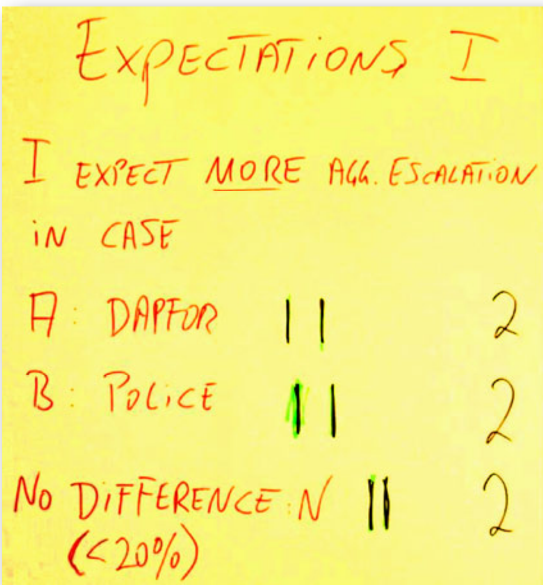


Figure 3: Team members' expectations for NOLH experiment

These expectations were than compared to the actual results, which are shown in Table 1.

	Ratio 1: 50 demonstrators 28 agitators	Ratio 2: 60 demonstrators 18 agitators
DAPFOR (A)	5699	5666
Danubian Police (B)	7749	7163

Table 1: The overall aggregated escalations caused by aggressive actions performed by civilians and soldiers

Thus, the Danubian Police seems to perform better with regard to the prevention of escalation. When looking at the experiments' results in more detail using statistical software (JMP 5.1), the soldiers' thresholds for intervention, the civilians' respect for the security forces and the civilians' readiness for aggression turned out to be the most important factors regarding the evolvement of collective aggression. Since the first two factors characterize the different mission approaches (the Danubian Police reserve intervenes much faster than the DAPFOR reserve, who waits to be called), it was decided to set up another gridded design experiment, varying both the Demonstrators' and the Agitators' readiness for aggression taking into account three different sets of RoEs for the soldiers.

Using the Gridded Design

In the gridded design, the agitators' readiness for aggression was varied in the range of 50 to 100 (maximum value) and the Demonstrators' readiness for aggression was set to values in the range of 20 to 70. The soldiers' behavior was determined by three different rule sets:

- Gandhi: the soldiers try to deescalate whenever possible
- PSO Manual: all types of actions (deescalating, threatening and defending) are performed by the soldiers
- Zero Tolerance: no deescalating actions are performed by the soldiers

Again, the team wrote down its expectations for case 2 (60 demonstrators and 18 agitators).

Figure 4 shows, for instance, that all team members were convinced that DAPFOR would perform best in the case of high values for both, the demonstrators' and the agitators' readiness for aggression. However, the results of the gridded experiment showed that the opposite was true. The Danubian Police was more effective than the DAPFOR soldiers in avoiding escalation.

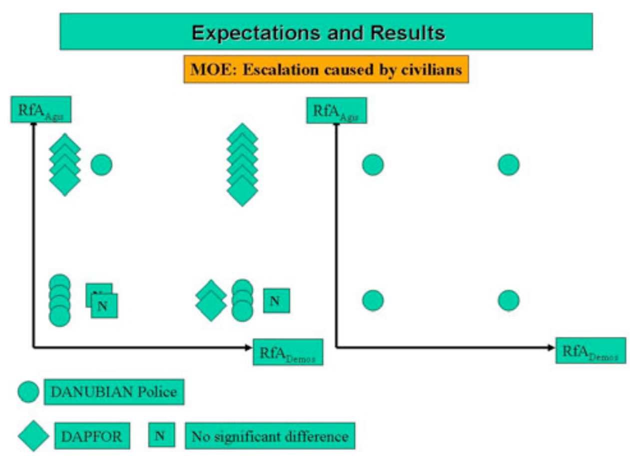


Figure 4: The team members' expectations for the gridded experiment compared to the actual results

In the analysis of the fitness landscapes resulting from the gridded experiment performed on the German 128 node cluster, some of the expectations of the group were not met. The results were examined in more detail looking at the course of action of relevant single PAX runs.

At first glance, the Danubian Police with the reserve acting very proactively even in case of quite low escalation seems to be the best way to minimize escalation. However, when looking at other MOEs, some negative aspects could be identified. The reserve using high show of force (making civilians more fearful) and intervening immediately may help to keep the overall escalation low, but this also leads to quite a high number of wounded civilians (see Figure 5). By contrast, in the case of deploying the DAPFOR soldiers no civilians were wounded (see Figure 6). Therefore another experiment was submitted, analyzing the effect of the Danubian reserve's proactive behavior by specifying a very high value for the reserve's threshold for intervention (i.e. the reserve only intervenes when being called by the chain of guards). Applying this parameter constellation, no civilians got wounded and with regard to the escalation expected there was no significant difference to the DAPFOR experiment.

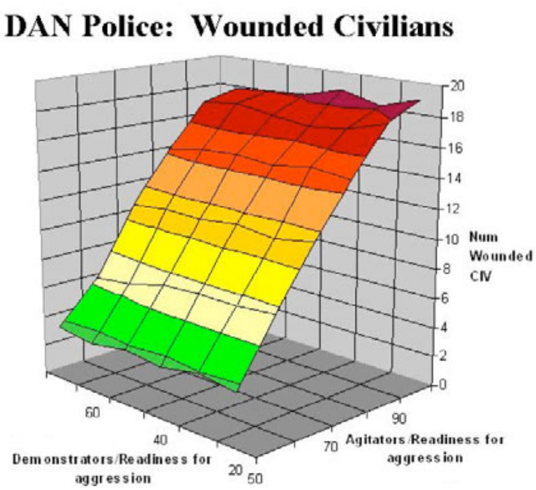


Figure 5: Number of wounded civilians in case of experiment B (Danubian Police), ratio 1 (50 demonstrators, 28 agitators)

Figure 6: Number of wounded civilians in the case of experiment A (DAPFOR), ratio 1 (50 demonstrators and 28 agitators)

Thus, with the help of the PAX simulation, the PSO team was able to issue some recommendations for the commander on-site. The task of securing the government district can be transferred to the Danubian Police without hesitation since no significant difference with regard to the evolvement of escalation is to be expected. However, to avoid negative side effects, the Danubian reserve should behave rather passively.

SUMMARY

The new PAX features allow for more realistic scenarios which are easy to set up. The different experimental designs (NOLH and gridded design) help to figure out important parameters. It is possible to measure the influence of parameters the peace keeping forces can actively change and to gain insight into the different possible outcomes of an operation depending on these parameters. Suggestions for improvements could be found and working in a multinational group helped broaden the view. The PAX team attended all plenary sessions which gave insight into other ongoing work.

"Not everything that can be counted counts, and not everything that counts can be counted." ~ Albert Einstein

"About comets we could talk a lot since we know so little." ~ J. P. Hebel (19th Century)

"The reall tragedy of the poor is the poverty of their aspirations." ~ Adam Smith

"Sailors know that they never reach the stars. Still, they navigate by them."

"The only opponent is within." ~ M. Ueshiba

"A weak man has doubts before decision, a strong man has them afterward."

Team 2: Simulation of Technical Aspects in Network-Centric Operations: Results

TEAM 2 MEMBERS

B. ERDMANN – Lead, Contact*
Karsten Haymann
Roger Musselman
Dr. Klaus-P. Schwierz
EADS Deutschland GmbH

TEAM PROPOSAL

The German Federal Office of Defense Technology and Procurement, has been analyzing the influence of networked sensors and effectors on military capabilities. Background for the actual technical evaluations of sensors, effectors and the connecting network is the scenario vignette: Convoy Protection as part of an over all scenario PSO in an urban environment.

The convoy transports fuel, ammunition and food in an urban environment and is protected and supported by:

- Two check points as flank protections
- UAV, UGV
- 4 AWC (Wiesel)
- 3 LIV (Fuchs)

Evaluation of the Ground Picture

There is an asymmetric threat: A local burning obstacle brings the convoy to a stop. Mobile barriers are used in an ambush, with snipers and bazooka shots that are looking for an opportunity to intercept the convoy.

The convoy leader can react on information from UAV and UGV on possible trafficability of the pre planned route. Detours are possible.

The MOÉs are:

- RED casualties and
- BLUE casualties.

The technical effects of special sensors and effectors at the convoy and his NCO capability are examined. Variations are investigated in the technical representation of UAV / UGV speeds, communication and sensors, in scenario details and in a variation of protection and equipment and these will be interpreted in the following three step approach:

1. using existing equipment (sensors, effectors),
2. using equipment under development (sensors, effectors),
3. using future equipment (sensors, effectors).

Results

The basic implementation of the scenario in MANA is challenging because of the high level of detail and clearly we reach the limitations of the tool MANA. Especially the implementation of the convoy as a sequence of vehicles was really time consuming. In defining the "working point" of the scenario in various cases secondary and third order effects in MANA lead to completely unpredictable model behavior.

Sequentially, in 5 Data Farming activities on the 128 / 32 node clusters at EADS, we found a parameter set corresponding to our question base.

We continued the work of PAIW12 where we looked at 16 parameters: Convoy Speed, Hit-probability and Combat Distance of Fighting Vehicles, UAV / UGV Speed, UAV / UGV Sensor Range, UAV / UGV Classification Probability, Network Reliability, Network Accuracy, Bazooka Range, Bazooka Hit-probability, Bazooka Reaction Time, Rifle Range, Rifle Hit-probability, Rifle Maximum Number of Targets per Timestep.

Unfortunately the results were almost all equal to zero. That means no casualties on the blue side happened.

With our interpretations of the datafarming results from PAIW12, we saw that we did not appropriately appreciate the cumulative effect of the UAV's and UGV's detection probability ((per time-step). Furthermore we saw that with the selected ranges for the weapon factors we gave a certain disadvantage to the red side. Therefore we studied with our first investigations on IDFW13, the effect of the following factors:

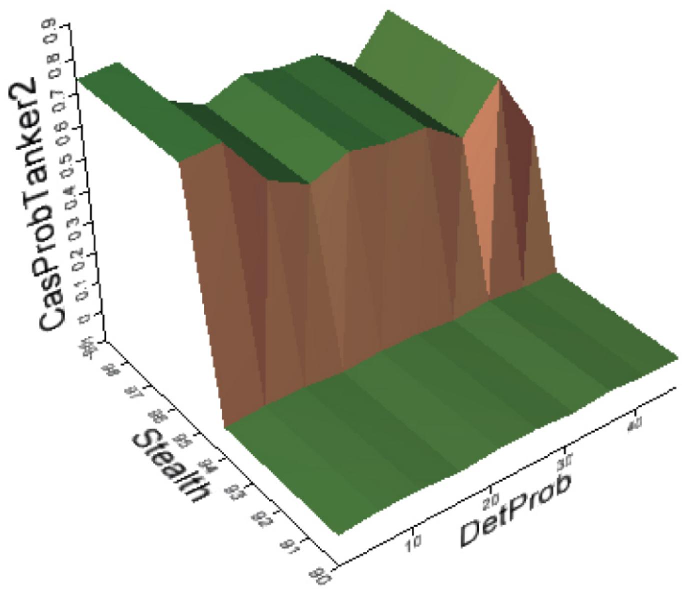
- UGV's detection probability
- Rifle's hit probability
- Rifle's maximum number of targets per time-step

* For more information contact: Bertram Erdmann, bertram.erdmann@eads.com, 804-765-1768, USA

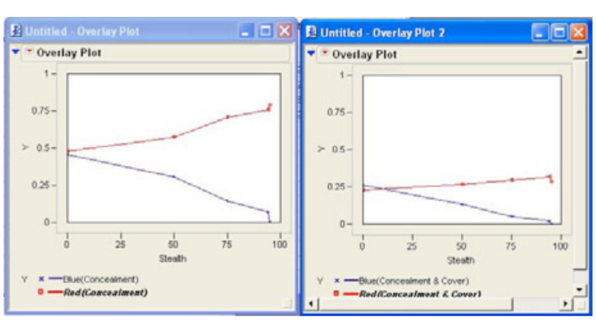
- Bazooka's hit probability
- AWC/LIV's hit probability

We found that the rifle's maximum number of targets per time-step is the most import factor (it already explains 70 % of the responses by itself). The following factors are as expected: the UGV's detection probability and the rifle's hit probability.

At the same time we had to observe that there were no casualties on the red side. We could explain this by the stealth factor of the red agents. Casualties only appeared when it was smaller than 95 %. This confirms a result we got during the revision of PAIW12.



Here we studied the probability of casualties of the second tanker by a gridded design for the factors stealth and UGV's detection probability. In the direction of the stealth factor there is a discontinuity in the surface and it is constant on the two branches. This behavior is strange and contradicts the results of a simple experiment we made with the default scenario of MANA where we added a terrain with the same attributes.



We made a thousand replications for each level of the stealth factor of the blue agent. The left part of the diagram shows the casualties of both agents with the terrain attribute concealment = 0.9 and cover = 0. For the right part the cover attribute was increased to 0.8. Obviously the curves are not constant.

This fact should be discussed with the developers of MANA.

To go on with our investigations we found a reasonable and practical workaround. By adding the "shot-at" - trigger state the blue escort can't open fire but return fire. And with the "taken shot" - trigger state the red side reduces the stealth factor from 95 % to 90 % when opening fire.

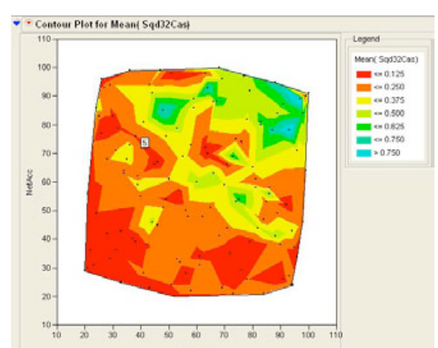
With these modifications and some changes in the design we carried out the data farming. In comparison to the design of PAIW12, besides some changes of ranges, we removed the concealment factor on the red side and added the hit-probability of the escort and the maximum number of target per timestep of the rifles. Instead of the 16 factors we had 17, and so we had to switch to the bigger NOLH - design with 129 excursions.

Again we worked with a hundred replications. Looking at the results we were especially interested in:

- the probability that the convoy takes the original route (to study the effects of the sensor and network parameters) =: MOE1
- the kill-probability of the first truck (food) (to study the effects of the escort parameters) =: MOE2
- the kill-probability of the last truck (gas) (to study the effects of the escort parameters) =: MOE3

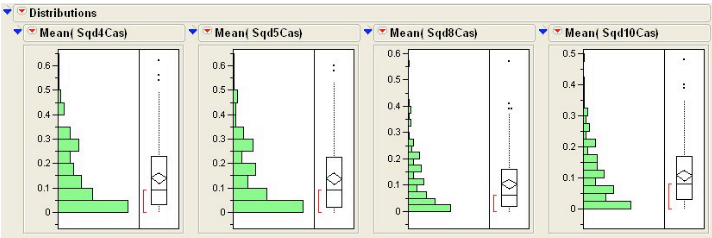
The regression tree for the probability that the convoy takes the original route (MOE1) shows that in the following order the network accuracy, the network reliability, the UGV's sensor range, the UGV's speed and the UAV's sensor range are the most important factors.

For example the following picture shows MOE1 depending on network accuracy and reliability.



It shows the expected behavior that with low accuracy and reliability of the network the convoy leader doesn't get the information to stay on the original route.

In contrast to the results of PAIW12 the next picture shows the distributions of the kill-probabilities of the trucks which obviously are not equal to zero.



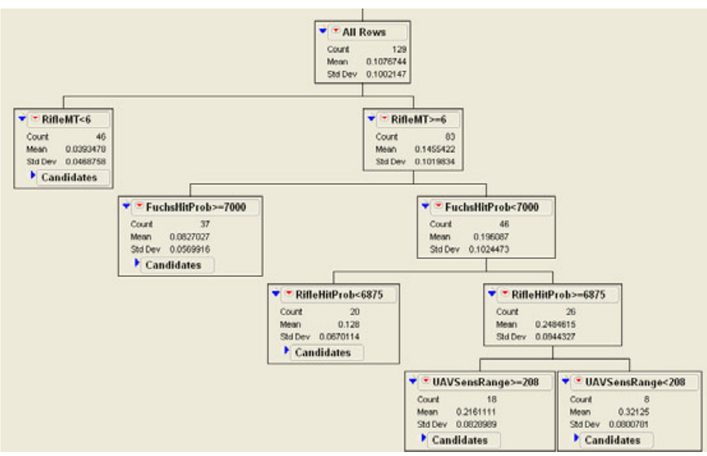
Quantiles

		Mean(Sqd4Cas)	Mean(Sqd5Cas)	Mean(Sqd8Cas)	Mean(Sqd10Cas)
100.0%	maximum	0.62000	0.60000	0.57000	0.48000
99.5%		0.62000	0.60000	0.57000	0.48000
97.5%		0.52750	0.51500	0.39000	0.38000
90.0%		0.33000	0.31000	0.24000	0.25000
75.0%	quartile	0.23000	0.22500	0.16000	0.17000
50.0%	median	0.09000	0.09000	0.06000	0.08000
25.0%	quartile	0.03000	0.02000	0.02000	0.03000
10.0%		0.00000	0.00000	0.00000	0.00000
2.5%		0.00000	0.00000	0.00000	0.00000
0.5%		0.00000	0.00000	0.00000	0.00000
0.0%	minimum	0.00000	0.00000	0.00000	0.00000

Moments

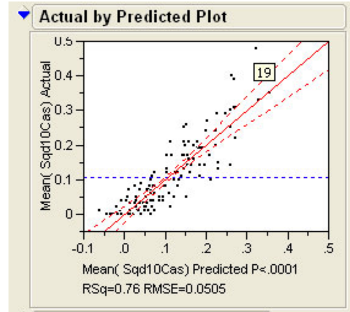
	Mean(Sqd4Cas)	Mean(Sqd5Cas)	Mean(Sqd8Cas)	Mean(Sqd10Cas)
Mean	0.1394574	0.1365116	0.1032558	0.1076744
Std Dev	0.1388193	0.1379688	0.1064495	0.1002147
Std Err Mean	0.0122224	0.0121475	0.0093724	0.0088234
upper 95% Mean	0.1636414	0.1605475	0.1218006	0.125133
lower 95% Mean	0.1152733	0.1124758	0.084711	0.0902158
N	129	129	129	129

So we could go on with first analysis. The regression tree neglecting the sensor and network parameters showed that the rifle's maximum number of targets per timestep and it's hit-probability followed by the hit-probability of the escort are the most important factors as far as MOE2 and MOE3 are concerned.



The upper regression tree for MOE3 was built taking all factors into account. It shows that for MOE3 the weapon parameters are more important than the sensor and network parameters.

By a regression model we studied interactions between factors. The model that explains the results to 75 % came up with some minor interactions :

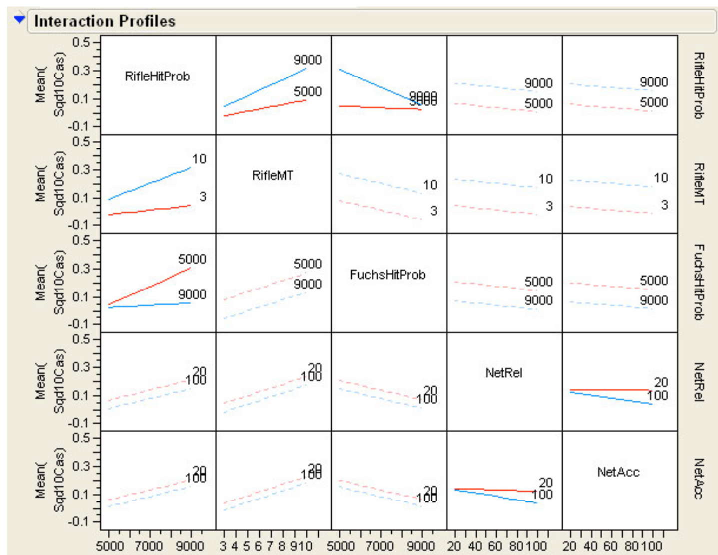


Summary of Fit

RSquare	0.761627
RSquare Adj	0.745736
Root Mean Square Error	0.050533
Mean of Response	0.107674
Observations (or Sum Wgts)	129

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	8	0.9790734	0.122384	47.9266
Error	120	0.3064289	0.002554	Prob > F
C. Total	128	1.2855023		<.0001



The workshop ended with a convoy scenario that worked fairly well. There are still open questions:

- Why wasn't there an action of the bazooka?
- Why was the bazooka not killed?
- How will the results change depending on the behavior on the red side?
- What are the effects of the sensors of the escort (without UGV an UAV)?
- What will be the effect of a mine road block?
- Can the upper questions be answered by MANA?

Above all we have to have a closer look at the results of the datafarming session at IDFW13!

Team 3: Communication Aspects in Urban and Sub-Urban Operations

TEAM 3 MEMBERS

CPT T. DOLL – Lead, Contact[1]
German Armed Forces, IT-Amt, Koblenz, Germany

P. HÜGELMEYER
B. KLAASSEN, PhD
T. ZÖLLER, PhD
Fraunhofer IAIS, Sankt Augustin, Germany

TEAM PROPOSAL

This research seeks to evaluate the importance of reliability and delay times in a communication network. Voice messages and data transmission in the network have to be considered distinctively. One part is the analysis of legacy sensor/effector and information systems currently used in the German Armed Forces. Due to historical developments information is transmitted non-optimally. Based on this research it is planned to envisage, specify and analyze a future sensor, communication and effector system. The system should be able to provide all information needed fast enough to maximize the soldiers' benefit.

We will perform this investigation mainly with ITSim. Here we can continue our investigations from PAIW12 in Boppard. Based on NATO NEC Scenario B we developed a test scenario called CANCOP (Communication Aspects in Network Centric Operations) with the following aspects:

NATO forces mount a peace-enforcement operation in the city of Khapital. The regular government only controls the city center and the airport. Armed gangs perpetrate organized crime. Intelligence has located one of the warlords in his stronghold. Also present are a number of his lieutenants, a substantial number of gang members as well as hostages. NATO commanders decide to move on the stronghold. Infantry forces including SOF are likely to be the main element of the assault force.

We planned to focus research on the following vignettes.

Vignette 1: Convoy through narrow streets under attack by red militia.

- The route finding is led by a helicopter pilot or by a UAV sensor grid.
- Objective: Ensure the arrival of the convoy at the secure area with minimal loss.

Vignette 2: Convoy moving between the suburban parts of the town and a blue camp outside the town near the airport.

- There are different threats to be regarded compared to the previous vignette.
- The convoy leader has to cope with changing situations in urban, sub-urban, and free terrain.

TEAM 3 EXECUTIVE SUMMARY

RESEARCH QUESTION - What effect does the communication aspect have on operations in urban terrain?

MAIN GOALS IDFW13

This research seeks to evaluate the importance of reliability and delay times in a communication network. Furthermore, the effect of jamming the Red communication for force protection is investigated. Voice messages and data transmission in the network have to be considered distinctively. One part is the analysis of legacy sensor/effector and information systems currently used in the German Armed Forces. Due to historical developments information is transmitted non-optimally. Based on this research it is planned to envisage, specify and analyze a future sensor, communication and effector system. The system should be able to provide all information needed fast enough to maximize the soldiers' benefit.

During the IDFW13 a vignette for the below described scenario was prepared to be further investigated during IDFW14.

CHOSEN SCENARIO

Based on NATO NEC Scenario B we developed a test scenario called CANCOP (Communication Aspects in Network Centric Operations) with the following aspects:

---

[1] For more information contact: Cpt Thomas Doll, Mail: ThomasDoll@bundeswehr.org

- NATO forces mount a peace-enforcement operation in the city of Mazar-e Sharif.
- The regular government only controls the city centre and the airport.
- Armed gangs perpetrate organized crime.
- Intelligence has located one of the warlords in his stronghold.
- Also present are a number of his lieutenants, a substantial number of gang members as well as hostages.
- NATO commanders decide to move on the stronghold
- Infantry forces including SOF are likely to be the main element of the assault force.

The following vignette was developed:

- The raid of the stronghold is finished.
- A convoy with the captured warlord and freed hostages has to reach the base-camp at the airport.
- UAVs are patrolling the north-eastern part of the city.
- Blue motorized Infantry patrols the city centre.
- Red forces try to stop the convoy with mobile road-blocks.
- Some civilians sympathize with Red and provide information about the convoy.

CONCLUSION

The vignette was set up and tested within ITSimBw Version 2. Based on the finished vignette data farming experiments will be conducted during IDFW14.

"So many factors, so little time."

"Never stop questioning" ~ A. Einstein

"Imagination encircles the world." ~ *A. Einstein*

"Logistics sets the campaign's operational limits." ~ JOINT PUB 1,, JOINT WARFARE

"I live for surprises!" ~ G. Horne

"Life is like a box of chocolates, you never know what you're gonna get." ~ F. Gump

"Rally behind the Virginians!" ~ BG B. Bee 1861 Battle of Bull Run

"Real success is finding your lifework in the work that you love." ~ David McCullough

"A discovery is said to be an accident meeting a prepared mind."

"It's tough to make predictions, especially about the future." ~ Yogi Bera

"The answer is 42." ~ *Deep Thought*

Team 4: Exploring Information Sharing and Shared Awareness Generation with Agent-Based Simulation and Data Farming

TEAM 4 MEMBERS

D. MARTIN
EBR Inc., USA – Lead, Contact[1]

J. HENK
TNO, Netherlands

ABSTRACT

For the past two years, the Command and Control Research Program (CCRP) has been involved in experimental activities to investigate the C2 impact of cognition and collaboration processes, the distribution of decision rights, patterns of interaction, the structures of information flow, and other net centric related concepts.

As a part of that effort, the U.S. DoD has sponsored the design and development of a software environment for conducting human-in-the-loop experiments focused on information- and social-domain phenomena. This experimental environment, named ELICIT (Experimental Laboratory for Investigating Collaboration, Information-sharing, and Trust), presents a group of 17 players with an information distribution and assembly problem to explore how people share information and generate shared awareness. In the experimental scenario, subjects receive information about a future attack. The information relates to the party carrying out the attack, the form the attack will take, the time of the attack, and the location of the attack – the Who, What, When, and Where of the problem; the information is structured so that various sets of facts combine to allow an information area to be solved. The mission of the participants is to solve each question area by combining and sharing the information to which they have access. Each subject is able to transmit his known facts to other agents or to commonly available websites, as constrained by the network structure.

Introduction

The task completion processes of two alternative social network structures have been investigated to date: an hierarchical organization (tiered/layered network) and an edge organization (completely connected network) – though many additional network structures and other factors could be studied. The existing experimental environment can be varied in a variety of ways by changing the organizational arrangements under study, altering the relationships between the subjects, changing the types of subjects employed, altering the mechanisms by which information is shared, altering the incentives employed, or manipulating a variety of other controllable factors. The objective of this experimentation is to investigate social and cognitive impacts of organizational structure within the context of a simple cognitive task. Analysis will be required to identify relevant variables and determine the enablers and inhibitors of an organization's performance.

As part of an emerging collaboration between Canada and the United States, there is interest in generating an agent-based simulation version of this experiment (agent-based ELICIT, or ABE). Such a simulation would enable expanded exploration of the experimental factors postulated to influence the information sharing and shared awareness-building processes in the experiment, helping to focus the limited number of trials possible in the human-in-the-loop experimental environment. In turn, the live experiments would serve as valuable data points to evolve and validate the simulation. Ultimately, a hybrid experimental environment could be created, in which some players would be humans while others would be software agents.

The objectives of this syndicate are 1) to explore the suitability of various elements of the agent-based modeling environments associated with the IDFW suite of tools for creating an agent-based simulation version of the ELICIT experiment; 2) to interact with the IDFW community to gain insight into how agent-based tools and techniques may have been applied to relevant problems elsewhere; and 3) to demonstrate feasibility and initial progress on creating an agent-based ELICIT simulation by conducting an experiment on a strawman version of the model coded in NetLogo.

The first two objectives will be accomplished through interaction with workshop participants and supporting staff

---

[1] For more information contact: Danielle Martin, martin@ebrinc.com

to share ideas and gather knowledge on agent-based tools and applications of which the current participants are aware. For the third objective, the team must decide how to effectively and accurately represent the human information sharing and problem solving behaviors, network structures, and influencing factors within a simulation design, and then develop and execute an experimental plan relevant to the ELICIT experimental goals. The syndicate will begin with an initial version of a simulation, intended to be a starting point for discussion and development.

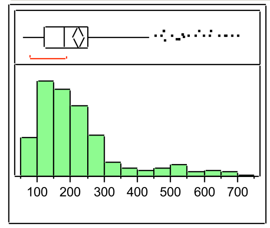
The U.S. DoD (OASD/NII) Command and Control Research Program (CCRP) has sponsored the design and development of a software environment for conducting human-in-the-loop experiments focused on information- and social-domain phenomena. This experiment has come to be know as the ELICIT Experiment. Over the course of several Project Albert International Workshops, EBR has strived to create and improve a simulation version of the experiment. During this week, Ms. Danielle Martin and Mr. Henk Jansen worked together to study how an organization's structure impacts a groups ability to share information in order to solve a simple cognitive task.

In the scenario participants received information about a future attack. The information is parsed into four question categories and the participant's mission is to gain sufficient knowledge related to each topic to solve the four questions. These information facts are periodically distributed and then shared via one on one interactions or website broadcasts. The network's objective is to solve the four task questions by combining and sharing the set of information facts. Participant actions are constrained by the network structure. Any given participant's awareness depends on what combination of facts they have seen.

Analysis Summary

Of all the variables farmed, a participant's tendency to collaborate up the hierarchy chain instead of with peers had the greatest impact on solution time. Second to collaborative probability, a participant's probability to post information to a website was an important factor in decreasing the performance of the Hierarchical Organization. Surprisingly, the Edge Organization was not significantly affected by the network's connectivity and the number of facts that could be gleaned from the website each time step. Ultimately it is important to note that the Edge network solves the task quicker than the Hierarchical network.

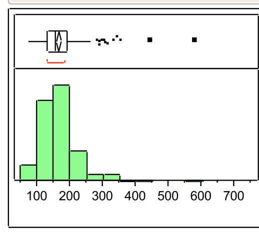
SolutionTime



Quantiles		
100.0%	maximum	704.00
99.5%		700.11
97.5%		620.98
90.0%		438.70
75.0%	quartile	253.00
50.0%	median	182.50
25.0%	quartile	119.50
10.0%		100.00
2.5%		79.03
0.5%		57.08
0.0%	minimum	54.00

Moments	
Mean	221.92083
Std Dev	136.78162
Std Err Mean	8.8292155
upper 95% Mean	239.31385
low er 95% Mean	204.52781
N	240

SolutionTime



Quantiles		
100.0%	maximum	582.00
99.5%		554.32
97.5%		316.83
90.0%		232.00
75.0%	quartile	191.75
50.0%	median	159.00
25.0%	quartile	133.00
10.0%		108.00
2.5%		82.03
0.5%		71.41
0.0%	minimum	71.00

Moments	
Mean	167.37083
Std Dev	60.247051
Std Err Mean	3.8889304
upper 95% Mean	175.03179
low er 95% Mean	159.70988
N	240

Figure 1 – Statistical Summary

It is important to note that each scenario was varied across eight different design points (varying parameters) for thirty replications. A few outliers are present in the data and more design points are required for more complete analysis. Further analysis in expected.

Through data farming, we are able to expand our exploration of the experimental factors postulated to influence the information sharing and shared awareness-building processes.

Interested in evaluating the utility of the replications of processes and information for future work, we intend to utilize these insights and continued data farming results to better focus the evolution and testing of live experiment. For this reason, we plan to continue to refine and analyze the model to more accurately reflect the information facts and the actions performed by the experiment subjects.

The two efforts compliment one another, informing the development and execution of associated human experiments, and leveraging information and data from ongoing experiments. We hope this will assist in achieving our future goal to create a hybrid experimental environment, in which some players would be humans while others would be software participants.

Additional details on the ELICIT Experiment can be found at: http://www.dodccrp.org/html2/parity.html.

Team 5: Investigating Ground Swarm Robotics Using Agent Based Simulation

TEAM 5 MEMBERS

CAPT. T. HO – Lead[1]
Singapore Army, Naval Postgraduate School, USA

CAPT. D. LOVELACE
USMC, Naval Postgraduate School, USA

TEAM PROPOSAL

The concept of employing ground swarm robotics to accomplish tasks in the future is not a new one. Some suggested applications mentioned in the literature include humanitarian de-mining, plume monitoring, search for survivors in a disaster site, etc. More importantly in the military context and with the development of advanced explosive detectors, swarm robotics with autonomous search and detection capability could potentially address the IED problem faced by foot patrols, and aid in the search for hidden ammunition caches and weapons of mass destruction.

The origins of the idea of robot swarms can be traced back to nature, where ant and termite colonies have demonstrated the ability to accomplish complex tasks by following simple sets of rules. Swarm robots are envisaged to be small, autonomous platforms that individually are incapable but collectively and cooperatively are able to produce an emergent behavior that allow them to fulfill a mission. These robots have the characteristic of being simplistic and low cost, so that they could be manufactured and deployed in mass without being overly concerned about their survivability.

The intent of this working group was to leverage on agent-based simulation (MANA) and data farming to model a ground robotic swarm on a search and detection mission and its technical aspects, and attempt to identify factors such as speed of robot, detector capability etc, that will contribute most to its effectiveness. The scenario of interest is to deploy the swarm in a semi-urban environment to search and detect stationary targets (IEDs are modeled as the type of target).

The working group looked into expanding the design of experiment to gain more insights on quadratic effects and interactions as a follow up from previous findings. In addition, the modeling of factors such as attrition of robots and tactics, techniques and procedures (TTPs) was incorporated to investigate the impact of these factors on the effectiveness of the swarm. The validity of the attempt to use agent based simulation, particularly MANA, to model swarm robotics was discussed. The focus of this working group was to explore agent based simulation applied to swarm robotics. The technological and algorithmic aspects was not be delved into.

Background

In the future, it is possible that ground swarm robotics with autonomous search and detection capability could aid in the search for IEDs, hidden ammunition caches and weapons of mass destruction. Swarm robots are envisaged to be small, autonomous platforms that individually are incapable but collectively and cooperatively are able to produce an emergent behavior that allow them to fulfill a mission.

Objectives

The intent of this working group is to leverage agent-based simulation (MANA) and data farming to follow up with some findings obtained prior to the workshop. The working group expanded the design of experiment (DOE) from previous research[2] so as to gain more insights and fidelity on quadratic effects and interactions. In addition, swarm robots getting trapped by terrain hazards while moving within the area of operations are modeled. General insights on the impact of such hazardous terrain on swarm robustness and effectiveness are obtained.

DESIGN OF EXPERIMENT AND SUMMARY OF RESULTS

Scenario 1

A foot patrol suspects that an area is rigged with IEDs and sends in swarm robots to clear a 50m x 50m area in 30

---

[1] For more information contact: Terence Ho, xterencex9@yahoo.com.sg, +65-91764203, Singapore

[2] NPS Thesis by Terence Ho

minutes or less. An unmapped environment is planted with 10 IEDs out of 30 candidate positions. The robots in this scenario are "virtual pheromones-capable". They navigate autonomously, perform their search-and-detect mission, and report back to their commander when detections are made. Previously, an 8 to 11 factor Near-Orthogonal Latin Hypercube (NOLH) was used for the 11 factor experiment. Team 5 expanded the DOE by using a 12 to 16 factor NOLH and performed 30 replications on 65 design points, resulting in 1950 runs. The MOEs capture both the time taken to accomplish the mission as well as the number of IEDs detected within the 30min mission time, and are as follows. It should be noted that there will be correlation between the MOEs.

- MOE 1: Time to Accomplish Mission
- MOE 2: Mission Accomplishment or Failure (binary response)
- MOE 3: Number of IEDs with at least 3 unique detections

The factors of interest are summarized as follows.

	50m by 50m Terrain		
	Decision Factors	**Low**	**High**
1	No of Robots	20	200
2	Sensor	2 (0.5m)	40 (10m)
3	Speed of Robot	4/100 (0.1m/s)	80/100 (2m/s)
4	Detector Capability, continuous function (95% detection after x seconds)	0.74 (1sec)	0.98 (15sec)
5	Detector Reset Time	1 (0.1sec)	100 (10sec)
	Noise Factors	**Low**	**High**
1	Uninjured Friends	-60	-25
2	Cover	-50	-10
3	Precision Move	100	300

Table 1: Factors to be farmed over for Scenario 1

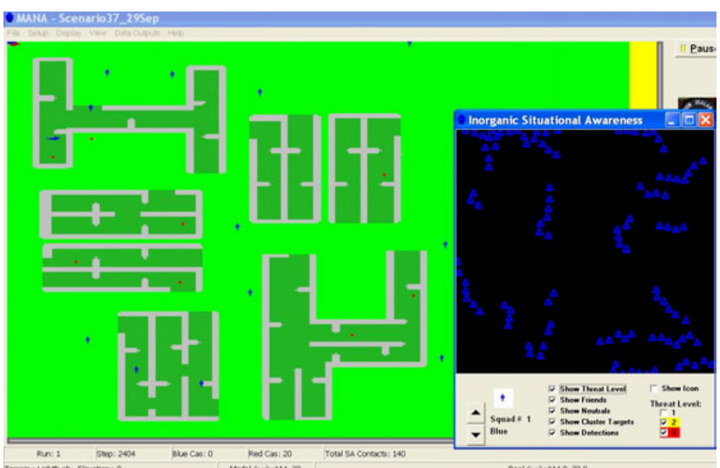
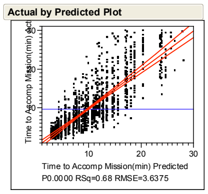


Figure 1: Snapshot of scenario with virtual pheromone trails of swarm robots

Results and Analysis

The analysis is done using regressions, partition trees, contour plots, profilers and distribution plots. The regression of time to accomplish mission (conditioned on mission accomplishment), shown below, is used to gain insights on significant main effects, interactions and quadratic effects.

Actual by Predicted Plot



Summary of Fit

RSquare	0.682586
RSquare Adj	0.679366
Root Mean Square Error	3.637539
Mean of Response	9.781179
Observations (or Sum Wgts)	1594

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	21954.552	1659.24	<.0001
Sensor Range(m)	1	1	6607.854	499.3960	<.0001
Speed(m/s)	1	1	8308.720	627.9408	<.0001
Det Capab	1	1	290.498	21.9547	<.0001
Det Reset Time(sec)	1	1	273.321	20.6565	<.0001
Pheromone Sensor Range (m)	1	1	0.079	0.0060	0.9383
Pheromone Persistence (sec)	1	1	398.525	30.1189	<.0001
No of Robots*Sensor Range(m)	1	1	396.604	29.9738	<.0001
No of Robots*Speed(m/s)	1	1	193.524	14.6258	0.0001
Sensor Range(m)*Speed(m/s)	1	1	434.009	32.8008	<.0001
Sensor Range(m)*Pheromone Sensor Range (m)	1	1	90.164	6.8142	0.0091
Sensor Range(m)*Pheromone Persistence (sec)	1	1	142.666	10.7821	0.0010
No of Robots*No of Robots	1	1	4706.072	355.6667	<.0001
Sensor Range(m)*Sensor Range(m)	1	1	243.478	18.4011	<.0001
Speed(m/s)*Speed(m/s)	1	1	3389.588	256.1719	<.0001
Pheromone Sensor Range (m)*Pheromone Sensor Range (m)	1	1	736.879	55.6905	<.0001

Figure 2: Conditional regression of time to accomplish mission

Blocked in red are factors that explain a large part of the model obtained by stepwise regression. The factors that show up as quadratic, i.e. number of robots, speed and pheromone sensor range are consistent with reasoning and previous findings. Interaction terms tend to be more volatile, depending on the MOE that is regressed upon.

Summary of Findings:

- Re-established main effects as being number of robots, speed and detector range
- Re-established existence of quadratic effect of number of robots, speed of robots and pheromone sensor range
- Interactions exist between Speed*Sensor Range, Sensor Range*Number of Robots, and Speed*Number of Robots, but interactions are "volatile" and are largely dependent on MOE selected.

Scenario 2:

Scenario 2 is similar to Scenario 1 except with the incorporation of hazards that have a size and a probability to trap robots during the simulation. 30 possible hazardous areas were added to the scenario and randomly selected at the start of each simulation. The 14 factor DOE yields 65 design points with 30 replications performed each.

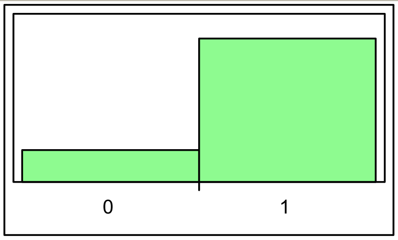
Factors added to Scenario 2

	Factors added to Scenario 2	Low	High
1	Number of hazardous areas	1	30
2	Radius of hazard area	1 (0.25m)	4 (1m)
3	Probability of trapping swarm robot	0	1

Table 2: Additional factors for Scenario 2

Results and Analysis

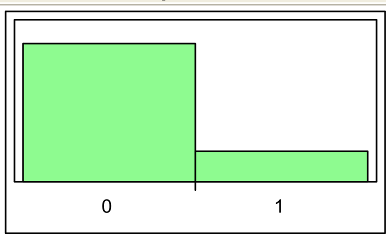
Mission Accomplished



Frequencies

Level	Count	Prob
0	356	0.18256
1	1594	0.81744
Total	1950	1.00000
N Missing	0	
2 Levels		

Mission Accomplished



Frequencies

Level	Count	Prob
0	1597	0.81897
1	353	0.18103
Total	1950	1.00000
N Missing	0	
2 Levels		

Figure 3: Mission Accomplishment deteriorates from 0.82 to 0.18 after hazards are introduced

It is clear that the existence of hazardous areas, which is to be expected in any area of operations, have an alarming impact on the ability of the swarm to accomplish its mission. It is important to identify how we can mitigate this and gain some insights on which factors are having the largest influence on the performance under such conditions. A regression tree is used as shown in the next figure.

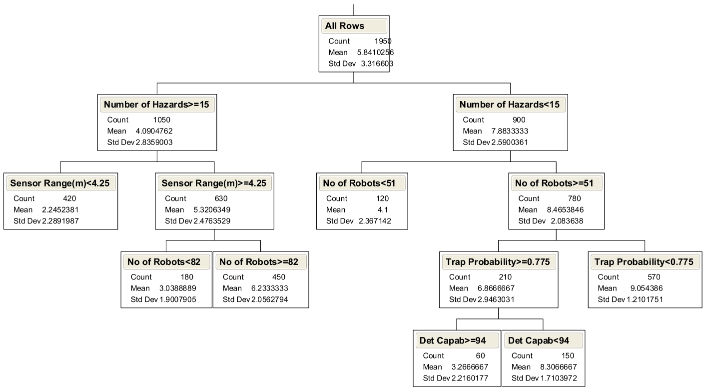


Figure 4: Regression tree of Number of IEDs detected

It can be seen that the factor that is causing the biggest difference to the ability to detect IEDs is the number of hazards. An area of interest that has less than 15 hazards could almost double the number of IED detected when compared to an area with more than 15 hazards (4.09 vs 7.88). Going down the regression tree, it goes on to show that the number of robots as being the next most important factor, followed by how likely the swarm robot gets trapped by the hazard.

Summary of Findings:

- Introduction of hazards deteriorates swarm performance drastically; mission accomplishment probability drops from 0.82 to 0.18
- Number of hazards, number of robots and trap probability are the three most important factors in determining the ability of the swarm to detect IEDs in a hazardous area
- Further analyses (not shown in this report) show that poor performance in Scenario 2 is mainly attributed by insufficient speed, number of robots and number of hazards

LIMITATIONS

It is important to acknowledge that the findings here are only applicable to this particular modeled routine of how the swarm robots perform search and detection. There are certainly many ways in which a robot swarm, e.g. communication ability with its neighbors could be used to enhance multiple unique detections of an IED detected by any robot. It is also worthy to mention that the requirement of three unique detections is to impose a stringent criteria for the robot swarm to achieve. Should the reliability of detectors be so high that only one detection is needed, then the insights gained are certainly the upper bound of what is required of the swarm robots.

"Any intelligent fool can make things bigger and more complex... It takes a touch of genius - and a lot of courage to move in the opposite direction." ~ A. Einstein

Team 6: System of Systems Test Planning in a Joint Mission Environment

TEAM 6 MEMBERS

S. MAHEU, LTC, – Lead, Contact[1]
US Army, JTEM Office, USA

T. BEACH
E. BJORKMAN, COL
JTE, OSD, USA

L. CLITES
G. HORNE
S. UPTON
Referentia Systems Inc., USA

O. SENDSTAD
FFI, Norway

C. WEGNER, MAJ.
Naval Postgraduate School, USA

PROPOSAL ABSTRACT

The concept of system-level testing of defense acquisition programs has changed. While system level testing remains necessary, it is no longer sufficient for making meaningful acquisition decisions. DoD planning guidance directs that test and evaluation activities include tests in joint environments. This will require acquisition program managers to conduct testing in environments where their new system is a participant in an overarching, joint system of systems. Such a test environment may include live assets along with virtual and constructive simulations. One can immediately recognize the difficulties of such testing. Planning and designing system of system tests is complex due to the need to replicate an entire mission environment, and the number of potential factors that could affect results or outcomes. Agent based simulation and Data Farming are methods that can help testers determine which factors significantly affect mission outcomes and which do not. This, in turn, allows testers to focus scarce resources on the most important test conditions.

The scenario we started with during IDFW 13 was one involving close air support (CAS). CAS is air action by fixed- and rotary-wing aircraft against hostile targets that are in close proximity to friendly forces and which require detailed integration of each air mission with the fire and movement of those forces. The particular joint environment of interest in this phase includes adverse weather, Army and Marine ground elements, Air Force and Navy aircraft, adversary air defense systems with modern surface-to-air missiles (SAMs) and artillery, and adversary anti-precision-guided munitions (PGM) systems employing cover, concealment, camouflage, decoys, and deception. Joint forces will use a joint network-enabled command, control, intelligence, surveillance, and reconnaissance (NEC2ISR) structure to process CAS requirements.

In this scenario, the targets will be both mobile and static. Targets can be detected by airborne and ground based sensors. Target information is transmitted via data link to the NEC2ISR structure for processing. The command and control element will use target information in the CAS request, together with information available on the network, to select CAS aircraft. CAS aircraft will receive information about the target and the specific joint terminal attack controller (JTAC) who will provide third-party targeting. Using information from the C2 network, the CAS aircraft releases a NEW within the computed launch acceptable region (LAR). The JTAC continues to track the target and sends updated target locations through the weapon control network. The NEW recognizes updated target locations (or new targets) from its assigned JTAC (while ignoring target locations from other JTACs) and guides to the assigned target.

Our work during IDFW 13, contributed to insight on the following question: What might matter and what doesn't appear to matter in the employment of systems before actual testing takes place? Through modeling and Data Farming of the scenario and variations described above, the goal was to better understand what might be tested. The plan for the workshop was to utilize MANA, Pythagoras, and perhaps other models to model the scenario and excursions. The plan was then to use data farming methodology to

---

[1] For more information contact: LTC Stafford Maheu, Stafford.Maheu@kirtland.af.mil, 505 846-8318, USA

understand the possibility space for important variables in the systems under test. The work at IDFW 13 was intended to be part of a continuing process to assist in the test planning in particular and in the overall development of a Joint Test and Evaluation Methodology.

Scenario

In the joint close air support (JCAS) scenario, mobile and static targets are detected by airborne and ground based sensors. Target information is transmitted via data link to the NEC2ISR* structure for processing. The command and control element uses this target information to direct aircraft to perform JCAS. The JCAS aircraft fly to launch position and release a Network Enabled Weapon (NEW) which receives updated target information (or a different target) from a Joint Terminal Attack Controller while guiding to the assigned target.

Activities

Team 6 consisted of members representing JTEM, NPS, FFI and Referentia. The team posed the question "What might matter and what doesn't appear to matter in the test planning of a system of systems acquisition test event before actual testing takes place?" Through modeling (in MANA, Pythagoras, and NetLogo) and Data Farming the team goal was to understand the possibility space for important variables in the system of systems to be tested.

Initial Results

The work at IDFW 13 provided promising results which will lead to continued work in support of acquisition test planning. The team ran the MANA scenario in which five factors were varied over selected ranges. A regression tree plot indicated that almost 50% of the variation in the response could be attributed to the absence or presence of countermeasures in the test. Three other variables showed that each accounted for some amount of variation; while the fifth did not show any variation within the range of values over which data farming occurred. Pythagoras modeling was stopped after day three of the IDFW as there were difficulties depicting the command and control relationship with that modeling tool. Preliminary results utilizing NetLogo show significant potential. It is possible that the system of system models using both MANA and NetLogo will be viable.

Way Ahead

This effort will continue with the goal of having a model which can be utilized by JTEM during the evaluation of joint test planning activities. Future work will include the application of the model to the test planning of an actual Network Enabled Weapon test event in 2007. Additionally, an upcoming NPS thesis on the Two Phase Adaptive Sequential Factor Method will utilize and enhance this model for the purposes of test planning.

> "When the number of factors coming into play in a phenomenological complex is too large scientific method in most cases fails. One need only think of the weather, in which case the prediction even for a few days ahead is impossible. Nevertheless, no one doubts that we are confronted with a causal connection whose causal components are in the main known to us. Occurrences in this domain are beyond the reach of exact prediction because of the variety of factors in operation, not because of any lack of order in nature."
>
> *~ Albert Einstein*

Team Photos

Team 1

Team 2

Team 3

Team 4

Team 5

Team Photos

Team 6

Team 7

Team 8

Team 9

Team 7: Applying Automated Red Teaming in an Urban Ops Scenario

TEAM 7 MEMBERS

M. LEE, CPT - Lead, Contact[1]
SO Analyst, GS (Dev) G5 Army, Singapore

D. ANG
SMTS, DSO National Laboratories, Singapore

In consultation:
L. FUNG HUEE, LTC
Hd BattleLab, GS (Dev) G5 Army, Singapore

INTRODUCTION

With rapid urbanisation, troops today will have to operate in an increasingly complex and urbanised environment. Together with a more potent enemy capability, the troops will have to be highly armour protected even at the lowest level (company size) in order to minimise the casualty rate. The fighting force will need to be a combined force to achieve a swift and decisive result in an urbanised terrain. This study explored the Coy level urban fighting force packages operating in a built up area.

AIM

To present the results on the relative performance of each proposed Coy level urban fighting force structure.

OBJECTIVES OF THE STUDY

The objectives of this study were to determine the performance of the various proposed Coy level urban fighting force structures to open and clear an axis through a built up area.

The Auto Red Teaming (ART)[2] framework developed by DSO National Laboratories was used to identify the key parameters that would affect the outcome of the urban war fighting scenario.

DESIGN OF EXPERIMENT

Blue Urban Fighting Force Structure. In this study, three Coy force structures were studied. The three structures would be analysed with two different armour platform (medium and heavy) for IFV/NLOS. The structures proposed were namely Tank heavy company, Balanced company and NLOS heavy company. The compositions of each structure were shown in Table 1.

The three force structures represented a wide spectrum of possible combinations of Tank platoons and NLOS sections within a company size force. Engineer elements were left out in this study as no obstacles were modelled in the scenario. The study assumed that an NLOS section was a reasonable trade-off with a Tank platoon.

Structure	Tank Heavy	Balance	NLOS Heavy
HQ	2 tanks	2 tanks	1 tank
Tank platoon (4 tanks each)	3	2	1
AI platoon (3x IFVs each)	1	1	1
NLOS section (2x NLOS + 1x UAV each)	1	2	3
Total	14 x tanks 3 x IFVs 2 x NLOS 1 x UAV	10 x tanks 3 x IFVs 4 x NLOS 2 x UAV	5 x tanks 3 x IFVs 6 x NLOS 3 x UAV

Table 1: Composition of Coy Level Urban Fighting Force

DESCRIPTION OF SCENARIO

In this scenario, the terrain profile in the area of operations comprised of mainly High Density built up area. No neutral or civilian exist in the AO as they were not modeled in this study. The primary task for the Blue forces was to clear an axis to open a path for follow up forces. The secondary task for the Blue forces was to attract enemy fires and inflict as much damages as possible to the enemy forces.

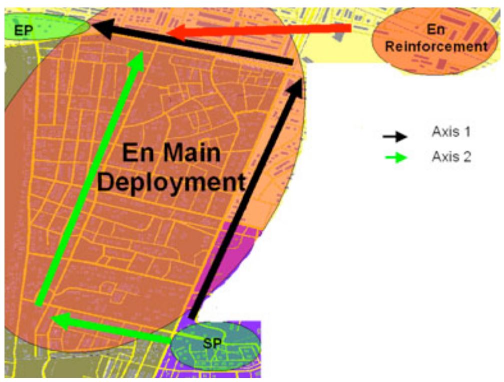
The Blue force behaviour modeled in this scenario was defined as follows:

---

1 For more information contact: LTC Marcus Lee, marcuspiggy@yahoo.com

2 ART is a technique to uncover system vulnerabilities or to find exploitable gaps in operational concepts, with the overall goal of reducing surprises, improving and ensuring the robustness of the Blue ops concepts.

a. The Blue tanks will manoeuvre along the pre-defined axes and will engage the Red forces when detected or being fired upon, the tanks will resume their movement along the intended axes after the engagement.
b. The Blue IFVs and NLOS will slow down when the Blue UAV or tanks detected the Red forces. The NLOS will engage according to their pre-defined target engagement priorities. Upon no further detections of enemy, they will resume their movement along the intended axes.



The AO would be defended by an enemy Armoured Cbt Tm (Company size) with the support of a RPG platoon and reinforced by another Amoured Cbt Tm. The Red force profile modeled in this scenario is illustrated in detail as follows:

a. Composition:

Types	Ambushed	Reinforcement
Red Tank	4	4
Red IFV	8	8
RPG gunners	18	0
Mobile ATGM vehicle	2	2

Table 2: Red Force composition.

b. Red Course of Action:
Tanks, IFVs, ATGM vehicles and RPG gunners pre-deployed in the area of operation will launch surprise attacks on the approaching Blue force, with the order of engagement defined by their priority targets of engagement. The RPGs and ATGMs will embark on "Hit and Run" tactics, springing surprise attacks from their ambush positions and move to new positions to spring the next phase of surprise attacks. The tactic was designed to trap, delay and kill the Blue vehicles while the mobile Armour Cbt Tm rushed in to interdict from the sides.

MEASURES OF EFFECTIVENESS (MOE)

The MOEs were:

a. Blue Attrition – attrition of each component of Blue Force (At least 80% survivability)
b. Red Attrition – attrition of each component of Red Force (At least 50% attrition)

KEY PARAMETERS AND ASSUMPTIONS

The following were important assumptions made in the scenario:

a. Perfect communication networks existed for both the Blue and the Red forces. Hence the effects of imperfect comms were not represented or examined in this study.
b. UAV did not have the capability to detect RPG in ambush position (inside buildings).
c. Red forces in ambush positions had the benefit of firing the first shot before they can be detected.
d. No dismounting of AI Platoon from IFV vehicles was modeled.

The following platform classes were modeled:

Platform Type	Class of Protection
Blue Tank	Heavy
Blue NLOS	Medium or Heavy
Blue IFV	Medium or Heavy
Red Tank	Heavy
Red IFV	Medium

Table 3: Platforms and their Protection Levels.

It was assumed that the platforms have a priority of engagement as follows:

Shooter	Targeting Priority				
	1	2	3	4	5
Blue Tank	Red Tank	Red ATGM	Red RPG	Red IFV	-
Blue NLOS (Medium Class)	Red Tank	Red IFV	Red RPG	Red ATGM	-
Blue NLOS (Heavy Class)	Red Tank	Red ATGM	Red RPG	Red IFV	-
Blue IFV (Medium Class)	Red IFV	Red RPG	Red ATGM	-	-
Blue IFV (Heavy Class)	Red ATGM	Red RPG	Red IFV	-	-
Red Tank	Blue Tank	Blue NLOS	Blue IFV	-	-
Red IFV	Blue NLOS (Medium Class)	Blue IFV (Medium Class)	-	-	-

Red RPG	Blue NLOS (Medium Class)	Blue IFV (Medium Class)	Blue Tank	Blue NLOS (Heavy Class)	Blue IFV (Heavy Class)

Table 4: Platforms and their Targeting Priorities.

PRELIMINARY STUDY ON FACTORS OF INTEREST

Sensitivity analysis was carried out for the list of factors over the following set of values:

a. Level of Protection for Blue IFV/NLOS:
   - Medium Class, Heavy Class
b. No. of Red Reinforcement.
   - Tank and IFV

The data for the sensitivity analysis was generated using the Data Farming technique.

RESULTS AND ANALYSIS

As explained under the section of "Design of Experiment", this part of the study examined the performance and survivability of each force structure when tasked to conduct the battles depicted in the earlier paragraph. For each force structure, two variants of the IFV/NLOS platforms were examined, namely the Medium Class and Heavy Class types.

The results for the force structures equipped with the medium class IFV/NLOS platforms were presented in Table 5.

Table 5 shows the results of the 03 x proposed Urban Fighting structures equipped with medium class IFV/NLOS platforms. The results showed that the three structures all achieved comparable Red reinforcement attrition levels. Most of the Red reinforcement were attrited as the reinforcement were the first to be spotted and engaged by the Blue forces. For the Red ambush force, the Red IFV faced a relatively high attrition rate (80 %) by the Tank company structure while the other two structures only manage to achieve around a 50 % attrition rate. On the other hand, the Tank heavy company structure achieved the lowest attrition rate for the Red RPG compared to the other two structures. These results shown that the Tank heavy company structure is more capable in fighting a mobile force and less efficient against a static force.

On analysis of the Blue attrition figures, the number of platforms killed across the three proposed structures was approximately similar (between 5 to 9 vehicles). None of the three structures meet the 80% survivability benchmark. The attrition of NLOS and IFV platforms were noticeably low under the Tank heavy company structure, with the Blue tanks taking the highest attrition at 4 Tanks. The Balanced and NLOS heavy company structures show an inverse result whereby the attrition of the Blue tank is low but high for the IFV and NLOS. For these two structures the Blue tank attrition rate is between 2 to 3 while the IFV and NLOS faced high attrition rate between 2 to 3 (IFV) and 3 to 4 (NLOS).

This implied that the decrease of 1x Tank Platoon from the Tank company structure to the Balanced company structure and subsequently to the NLOS company structure had caused a transfer of enemy's concentration of firepower from Blue tanks to Blue IFVs and Blue NLOS. This transfer of enemy's firepower had resulted in a proportionate decrease in the number of Blue tanks killed, but had caused a greater than proportionate increase in the number of IFVs and NLOS killed.

Platforms	**Tank Heavy**			**Balanced**			**NLOS Heavy**		
	Qty	**No. Killed**	**% Killed**	**Qty**	**No. Killed**	**% Killed**	**Qty**	**No. Killed**	**% Killed**
Blue Tank	14	4.05	28.9	10	2.86	28.6	5	2.21	44.2
Blue NLOS	2	0.96	48.0	4	3.36	84.0	6	3.91	65.2
Blue IFV	3	0.43	14.3	3	2.62	87.3	3	2.69	89.7
Total Blue	19	5.44	28.6	17	8.84	52.0	14	8.81	62.9
Red Ambush									
Red Tank	4	3.55	88.8	4	3.14	78.5	4	2.81	70.3
Red APC	8	6.41	80.1	8	3.93	49.1	8	4.4	55.0
Red ATGM	2	1.92	96.0	2	1.82	91.0	2	1.62	81.0
Red RPG	18	5.55	30.8	18	10.38	57.7	18	7.9	43.9
Red Reinforcement									
Red Tank	4	4	100.0	4	3.99	99.8	4	3.99	99.8
Red APC	8	8	100.0	8	7.82	97.8	8	7.89	98.6
Total Red	44	29.43	66.9	44	31.08	70.6	44	28.61	65.0

Table 5 - MOEs for Blue structures with medium class IFV/NLOS platforms.

Platforms	Tank Heavy			Balanced			NLOS Heavy		
	Qty	No. Killed	% Killed	Qty	No. Killed	% Killed	Qty	No. Killed	% Killed
Blue Tank	14	3.63	25.9	10	1.65	16.5	5	1.61	32.2
Blue NLOS	2	0.04	2.0	4	0.54	13.5	6	1.06	17.7
Blue IFV	3	0.1	3.3	3	0.91	30.3	3	0.93	31.0
Total Blue	19	3.77	19.8	17	3.1	18.2	14	3.6	25.7
Red Ambush									
Red Tank	4	3.58	89.5	4	3.38	84.5	4	2.98	74.5
Red APC	8	6.08	76.0	8	4.44	55.5	8	5.36	67.0
Red ATGM	2	1.9	95.0	2	1.88	94.0	2	1.72	86.0
Red RPG	18	12.51	69.5	18	9.24	51.3	18	9.57	53.2
Red Reinforcement									
Red Tank	4	4	100.0	4	3.99	99.8	4	2.97	74.3
Red APC	8	8	100.0	8	7.79	97.4	8	8	100
Total Red	44	36.07	82.0	44	30.72	69.8	44	30.6	69.5

Table 6 - MOEs for Blue structures with heavy class IFV/NLOS platforms.

The results for the force structures equipped with the heavy class IFV/NLOS platforms are presented in Tables 6.

Table 6 shows the results of the proposed Urban Fighting structures equipped with heavy class IFV/NLOS platforms. The results showed that the Balanced and NLOS heavy company structures achieved comparable Red reinforcement attrition levels while the Tank company structure had a significantly higher Red attrition rate. The Tank heavy company structure achieved the highest attrition rate for the Red RPG compared to the other 2 structures. These could be due to the increase defense capability of the IFV/NLOS which enable them to survive the Red RPG attacks and thus create the opportunity for the Blue tanks to engage the Red RPG while they are exposed. All three structures meet the 50% attrition rate inflicted on the Red forces.

The overall Blue attrition rates were similar across the 3 structures but only Tank heavy and Balance structure meet the 80% survivability condition. This means that these two structures are the only structures that meet the MOEs requirement with the Tank heavy structure fairing better on the attrition on Red forces.

Sensitivity Analysis of Red Reinforcement Forces

Sensitivity analysis was performed on the number of tanks and IFV in the Red reinforcement forces to determine their impact on the Blue's survivability. The results are presented in Figure 1 to 6.

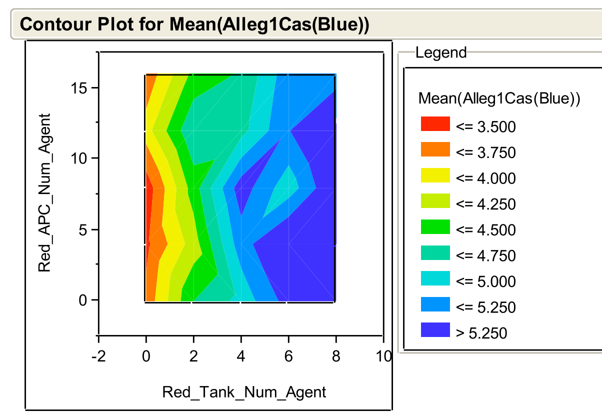


Figure 1: Tank structure with medium class IFV/NLOS platforms.

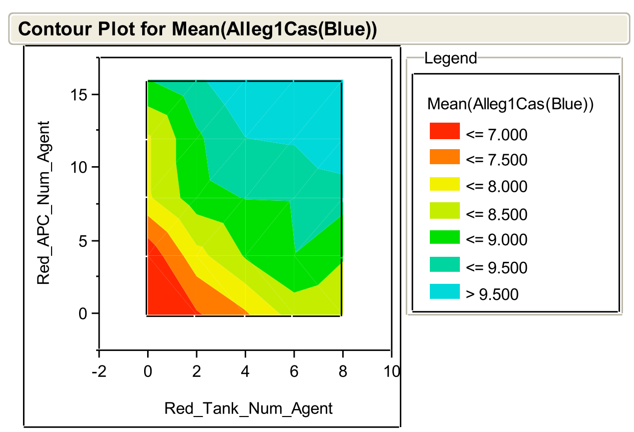


Figure 2: Balanced structure with medium class IFV/NLOS platforms.

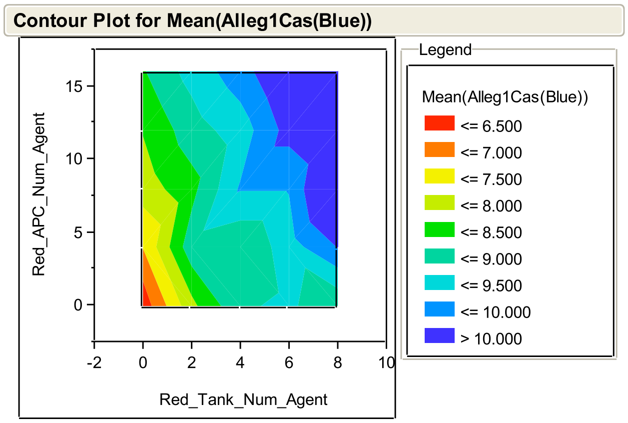


Figure 3: NLOS structure with medium class IFV/NLOS platforms.

Based on the Figures 1, 2 and 3, for the medium class IFV/NLOS studies, the Blue attrition increased when the Red reinforcement increased. However it is also noted that for the Tank heavy company structure, an increased in Red IFV does not contribute to the attrition rate of the Blue forces. This could be attributed to the inability of the Red IFV to inflict any damage to the forward forces of Blue tanks.

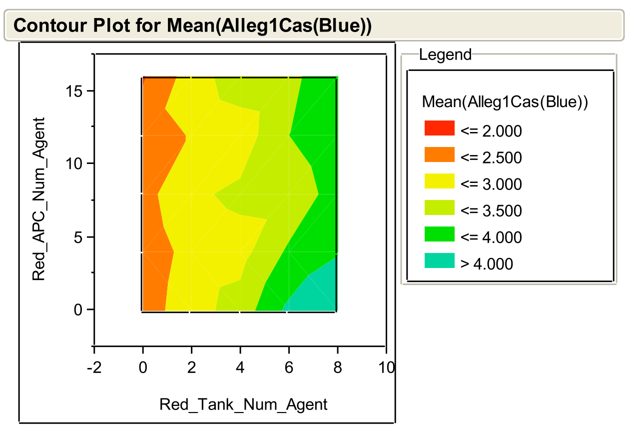


Figure 4: Tank structure with heavy class IFV/NLOS platforms.

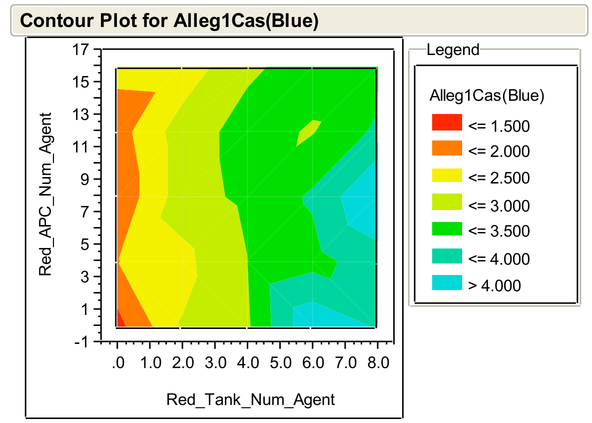


Figure 5: Balanced structure with heavy class IFV/NLOS platforms.

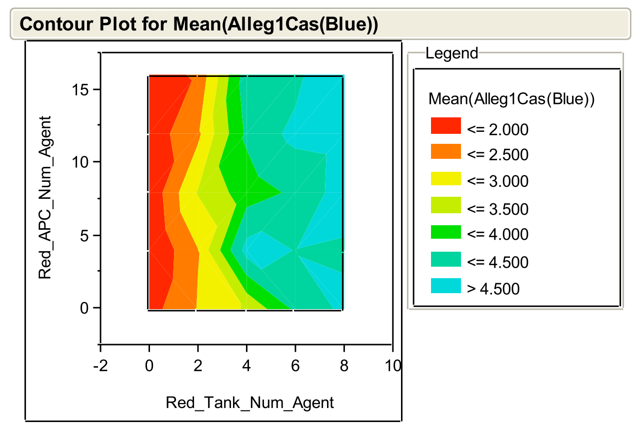


Figure 6: NLOS structure with heavy class IFV/NLOS platforms.

Figures 4, 5 and 6 depicts the effect of Red reinforcement on the attrition rate of Blue forces with heavy armoured IFV/NLOS. It clearly showed that an increase in Red IFV quantity has no significant effect on the Blue forces and this is due to the inability of the Red IFV to inflict any damage to the heavy armoured Blue IFV/NLOS.

Automated Red Teaming (ART) Framework

The intent of this study was to explore how intangibles could lead Red to break Blue. The scenarios used in this study were the 3 proposed structures with medium class IFV/NLOS platforms. We short listed the parameters in Table 7:

Red Farming Parameters	Min	Max
Red Reinforcement Tank Individual Aggression	-100	100
Red Reinforcement Tank Squad Aggressiveness	-100	100
Red Reinforcement Tank Response To Injured Red	-100	100
Red Reinforcement Tank Clustering	-100	100
Red Reinforcement Tank Squad Cohesion	-100	100
Red Reinforcement IFV Individual Aggression	-100	100
Red Reinforcement IFV Squad Aggressiveness	-100	100
Red Reinforcement IFV Response To Injured Red	-100	100
Red Reinforcement IFV Clustering	-100	100
Red Reinforcement IFV Squad Cohesion	-100	100
Red Ambush Tank Individual Aggression	-100	100
Red Ambush Tank Squad Aggressiveness	-100	100
Red Ambush Tank Response To Injured Red	-100	100
Red Ambush Tank Clustering	-100	100
Red Ambush Tank Squad Cohesion	-100	100
Red Ambush IFV Individual Aggression	-100	100
Red Ambush IFV Squad Aggressiveness	-100	100
Red Ambush IFV Response To Injured Red	-100	100
Red Ambush IFV Clustering	-100	100
Red Ambush IFV Squad Cohesion	-100	100
Red Ambush IFV Stealthiness	0	99

Table 7: Red Parameters for ART.

A negative value for the parameter denotes an aversion to the particular attribute. For instance, -100 for clustering means the agents prefer to spread out rather than sticking as a group. A neutral value, 0, would mean that the agent is indifferent. For stealth, the value ranges between 0 and 100, however, 100 was not taken as it would mean the unit is completely invisible.

The Measures of Effectiveness (MOEs) to be collected for analysis were:

a. Maximize Blue Attrition.
b. Minimize Red Attrition.

The data were then analyzed using the Clustering and Outlier Analysis for Data Mining (COADM)1 tool developed by DSO National Laboratories to identify the parameters associated with the best Red cluster, i.e. the cluster with the lowest Red attrition and highest Blue attrition. Below is a summary of the results in Table 8:

Red Farming Parameters	Tank		Balanced		NLOS	
	Mean	Var (+/-)	Mean	Var (+/-)	Mean	Var (+/-)
Red Reinforcement Tank Individual Aggression	-80.18	55.209	-76.97	0.504	29.51	0.189
Red Reinforcement Tank Squad Aggressiveness	-92.02	43.763	63.52	25.021	-13.74	0.33
Red Reinforcement Tank Response To Injured Red	-32.36	26.372	-59.71	8.132	-26.48	0.002
Red Reinforcement Tank Clustering	-77.11	1.285	-91.5	7.565	-86.2	14.901
Red Reinforcement Tank Squad Cohesion	-24.02	15.976	-28.57	8.198	-94.12	0.208
Red Reinforcement IFV Individual Aggression	2.64	29.89	62.8	17.784	45.92	0.901
Red Reinforcement IFV Squad Aggressiveness	3.4	24.593	65.38	36.251	92.1	0.332
Red Reinforcement IFV Response To Injured Red	-5.3	29.807	23.08	22.677	43.02	0.315
Red Reinforcement IFV Clustering	17.98	35.456	-75.54	3.324	-41.76	0.582
Red Reinforcement IFV Squad Cohesion	-87.7	0	-2.04	14.523	-3.07	11.006
Red Ambush Tank Individual Aggression	10.42	11.581	-85.02	1.418	-17.99	0.21
Red Ambush Tank Squad Aggressiveness	-82.12	35.319	10.21	0.925	87.83	0.375
Red Ambush Tank Response To Injured Red	75.22	38.811	26.41	26.874	56.93	0.155
Red Ambush Tank Clustering	37.99	41.54	-35.69	33.238	-50.11	1.412
Red Ambush Tank Squad Cohesion	-15.64	9.038	-48.91	5.362	-0.99	0.468
Red Ambush IFV Individual Aggression	28.64	86.719	17.55	11.594	35.5	0.332
Red Ambush IFV Squad Aggressiveness	47.52	27.774	98.75	28.474	-55.17	0.302
Red Ambush IFV Response To Injured Red	-24.1	39.793	11.21	0.525	88.16	0.465
Red Ambush IFV Clustering	43.64	42.42	-35.41	64.993	-59.38	0.233
Red Ambush IFV Squad Cohesion	94.39	0	-49.43	2.009	-21	0.692
Red Ambush IFV Stealthiness	97.02	0	92.29	1.326	94.8	0.51

Table 8: Results of Red Teaming Runs.

The above results indicated that an effective Red force against the Blue Tank structure would be for the Red reinforcement tanks not to cluster during movement and the Red reinforcement IFVs not to move cohesively. The Red ambush IFVs need to move in a cohesive and stealthy manner to avoid the forward deployed Blue tanks.

The above results indicated that an effective Red force against the Blue Balanced structure would be for the Red reinforcement force not to cluster during movement to avoid Blue fire support. The Red reinforcement and ambush tanks also need to be less aggressive individually. The Red ambush IFVs need to be highly stealthy and less cohesive to avoid the Blue tanks.

The above results indicated that an effective Red force against the Blue NLOS structure would be for the Red

1 CODAM is a data mining software package that is capable of visualizing complex data set and it keeps track of information which greatly facilitates the data mining process.

reinforcement force to be more aggressive individually and not to cluster during movement to avoid Blue fire support. The Red ambush tanks need to be more aggressive as a squad and with a propensity to move towards fellow injured Red. The Red ambush IFVs need to be highly stealthy, less cohesive to avoid the Blue tanks and with a propensity to move towards fellow injured Red.

	Tank		Balanced		NLOS	
	Base Case	ART	Base Case	ART	Base Case	ART
Blue Force						
Mean Attrition & Percentage	5.44 (28.6)	6.74 (35.5)	8.84 (52.0)	11.08 (65.2)	8.81 (62.9)	12.06 (86.1)
Red Force						
Mean Attrition & Percentage	29.43 (66.9)	23.4 (53.2)	31.08 (70.6)	25.92 (58.9)	28.61 (65.0)	21.98 (50.0)

Table 9: Comparison between Base Case Run and Red Teaming Results.

The Red Force recommended by ART has shown to achieve higher Blue attrition and lower their own attrition. By applying ART, we have effectively found gaps in performance of Blue's plan which would otherwise not be so easily identified.

Based on the indications of the red teaming results, the Blue should be prepared to face a possibly challenging Red Force and hence improve their capability and plans to counter the following red characteristics:

a. Stealth. Using better or more sophisticated sensors to identify stealthy Red agents hiding within buildings, can greatly aid in survivability of Blue. This is to ensure that the Red Force would not be elusive.
b. Cohesion. In order to counter the dispersion of the Red defending forces, it is important to derive plans to force the defence to cluster or co-locate at known positions to Blue. Carefully planted support fire and deceptive tactics can help Blue achieve this effect.
c. Aggression. Behavioural techniques to reduce aggression can also reduce Red's effectiveness. For instant, using a show of force (shock and awe) to intimidate the enemy.

With the results obtained, we have demonstrated the ability of using ART to search for associated parameter values that improved red force performance. In understanding what constitutes a potent Red Force, the Blue then has the ability to refine their plans and capability to ensure a more favourable and robust outcome when engaging an unpredictable Red Force.

SUMMARY OF FINDINGS

Blue structures with medium class IFV/NLOS platforms

Platforms	Tank Heavy			Balanced			NLOS Heavy		
	#	No. Killed	% Killed	#	No. Killed	% Killed	#	No. Killed	% Killed
Total Blue	19	5.44	28.6	17	8.84	52.0	14	8.81	62.9
Total Red	44	29.43	66.9	44	31.08	70.6	44	28.61	65.0

Blue structures with heavy class IFV/NLOS platforms

Platforms	Tank Heavy			Balanced			NLOS Heavy		
	#	No. Killed	% Killed	#	No. Killed	% Killed	#	No. Killed	% Killed
Total Blue	19	3.77	19.8	17	3.1	18.2	14	3.6	25.7
Total Red	44	36.07	82.0	44	30.72	69.8	44	30.6	69.5

Table 10: Summary For The Force Options

The study indicated that up-armouring of IFV/NLOS from the medium to the heavy class is probably required given the threats they would face in the urban environment. Both the Tank Heavy and Balanced company with the heavy armoured IFV/NLOS met the criteria of at least 80% survivability rate (Redcon 1) and are plausible force structures for urban fight. However, the Tank Heavy company is recommended as it inflicted more damage to the Red forces as compare to the Balanced structure.

Based on the ART findings, the Blue force should employ effective sensors to seek and destroy the Red ambushed forces. Carefully planned fire would be required to prevent the Red from scattering and behavioural techniques such as a show of force (shock and awe) could be used to curb the Red forces aggressiveness.

CONCLUSIONS

The findings presented in this paper highlighted some of the key issues for a force to fight in built up areas and can perhaps provide a useful basis for the future studies to be conducted.

It is important to keep in mind that the results were preliminary as many unique features of urban operations were, unfortunately, not possible to model here. To draw more conclusive answers and refine the options, it is recommended that further experimentation be conducted using other modeling and simulation tools, as well as to focus on the other potential operational tasks (such as obstacle clearance under hostile fire) of the Combat Team as part of the urban fighting mission.

Team 8: The Perfect Match for Virtual Combat ID Experiments

TEAM 8 MEMBERS

M. SPAANS – Lead, Contact[1]
M. SPOELSTRA
J. VOOGD
TNO Defence, Security and Safety, Netherlands

S. SANCHEZ,
Naval Postgraduate School, USA

ABSTRACT

The ultimate aim of the Combat Identification (Combat ID) team is to enable the representation of Combat ID characteristics and behavior within a constructive simulation in order to enable the exploration of the benefit of system interventions based on Situational Awareness, Target Identification, Human Factors and TTPs in terms of increasing combat effectiveness and reducing fratricide levels.

BACKGROUND

Fratricide incidents during recent conflicts have placed combat identification high on the agenda of coalition members. Several measures can be taken to support combat identification which decrease fratricide rates and increase combat effectiveness. Improving combat identification can only be accomplished by a combination of measures in several areas:

- training,
- tactics, techniques and procedures,
- technical systems to enhance situation awareness and
- target identification systems.

Apart from these measures, the quality of combat identification is influenced by operational circumstances and human factors. It is clear that no single system or measure can solve the problems faced during the combat identification process. It is important to have the right balance of investments to optimize results within operational, budget and other restrictions.

Large scale field experiments as part of the Coalition Combat ID Advanced Technology Demonstrator (CCID ACTD) have been and will be conducted to determine the influence of different measures. During these experiments only a limited number of technical and procedural measures are fielded and tested under a limited number of scenarios. It is useful to have a series of complementary virtual experiments to test different sets of measures under a broader spectrum of circumstances and scenarios.

During The PAIW 12 in Germany, a global design was developed that frame our first thoughts about a combat ID simulation model. This workshop was used to continue these efforts.

RESEARCH QUESTION

What is the effect of (a large number of) different variations in Situational Awareness, Target Identification, Human Factors and Tactics, Techniques and Procedures under different circumstances (scenarios) on mission level combat effectiveness and a mission level degree of fratricide?

MAIN GOALS

The ultimate aim of the Combat Identification (Combat ID) team is to enable the representation of Combat ID characteristics and behavior within a constructive simulation in order to enable the exploration of the benefit of system interventions based on Situational Awareness, Target Identification, Human Factors and TTPs in terms of increasing combat effectiveness and reducing fratricide levels.

The 13th International Data Farming Workshop following from Project Albert increased the potential to facilitate the instantiation of Combat ID processes within intelligent agents in order to enable the investigation of a large number of variations within different operational contexts (scenarios).

The two main goals of this team during the IDFW 13 workshop were:

1. The refinement of the model that was developed during PAIW 12. This model consists of Measures of

[1] For more information contact: Mink Spaans, mink.spaans@tno.nl, +31 703 740211, The Netherlands

Merits, Independent and dependant influencing factors and their relationships. This goal involves both detailed considerations about CombatID related characteristics and characteristics related to combat behaviour.

2. The development of an action plan to embed this model into PA-distillations or other agent based frameworks like the TNO-developed environment SPYSE.

Results and Further Plans

Team 8 was able to refine the process model from PAIW12, focusing on both the specification of the information models in the system and the agents, and on the choice of a general implementation method for the simulation. The discussions about the details, like situation awareness representation and information system representation proved to be very fruitful, but further discussion and refinement is needed in the following weeks.

The two implementation methods considered were the time-step driven model and the event-driven model, both of which have advantages for a Combat ID simulation. A hybrid approach, a combination of the two, was considered as a possible solution.

Most known simulation platforms are time-step driven. According to the discussions, a decision was made to gain experience with event-driven modeling techniques by constructing a simulation with very few elementary agents. In this way, more insight will be created in the possibilities of those techniques for the Combat ID domain.

Apart from the agent-based platform SPYSE, other modeling languages are also considered for the implementation of the simulation. One plan for the coming weeks is to create an overview of the possible simulation platforms that were previously not known to the team, like Simkit and other already existing frameworks.

Because of the scope of the project, possibilities will be looked into of having a student (from a Dutch university or abroad, perhaps NPS.) working on the simulation in the context of an internship or graduate project.

Conclusions and Future Goals

This week was a week full of fun and interesting discussions in which we made a lot of progress for future work. We have decided on a direction in which to go with the start of the implementation, and will try out the event-driven approach to gain insight in the value of that method for our simulation.

For the next Workshop, one goal is to have made all the choices necessary to start implementation of the simulation, and quickly start implementing, possibly with the help of a student.

"If you can't solve it on paper, you can't solve it on the computer."

"Everything should be made as simple as possible, but not simpler." ~ *A. Einstein*

"Research is what I'm doing when I don't what I'm doing" ~ *Wernher von Braun*

"And the beat goes on…"

"The whole is more than the sum of its parts."

"It's not that I'm so smart, it's just that I stay with problems longer." ~ A. Einstein

"I don't know of a solution, but I certainly admire the problem." ~ Ashleigh Brilliant

"What is in the Black Bag?"

Team 9: Using Agent Based Simulation to Mimic a Specific Human's Decisions

TEAM 9 MEMBERS

W. STILWELL, LTC, USA – Lead, Contact[1]
U.S. Army, The National Simulation Center

W. KENT, CPT, USA
Naval Postgraduate School

TEAM PROPOSAL

Question: Can agent-based simulation provide calibrated command agents for co-adaptive synthetic environment research? The Calibration and Validation Process for complex adaptive system simulation provides an iterative methodology to calibrate agent behavior within a simulation. This methodology can provide a platform for the development of co-adaptive systems in agent-based simulation by calibrating a command agent that senses the battlefield and makes decisions much like a targeted human commander. This group will develop a simple scenario in netlogo that uses a command agent to replicate a human commander's simple battlefield decisions such as retreat, attack, and commit the reserve. The expected result will be a simple co-adaptive combat scenario that can be calibrated to a particular real commander's decision style or profile.

TEAM ACTIVITIES

The ability to represent the decision style of a particular human may now be possible due to advances in NOLH Design of Experiments and VVA of complex adaptive system simulation. Team 9 is developing two simulation experiments as an exploration of this problem space. First, a simulation of a large medieval battle using primitive weapons seeks to capture a human commander's decision to commit reserves and to, if necessary, retreat. Second, a modern combat scenario models the risk attitude of the blue commander given the likelihood of chemical weapons use by the enemy.

A four phase development concept is employed. Phase I entails building the simulation. The large battle is modeled in Netlogo, while the modern battle is modeled in Pythagoras. Phase II is an exploration of the problem space, using data farming to develop the known behaviors of the model over a reasonable range of agent parameters. Phase III is a human-in-the-loop experiment that will capture a range of human decision styles. Phase IV is a calibration of agents that will mimic the decision style of those particular humans.

The International Data Farming Workshop #13 was used for primarily Phase I and Phase II. Development of Netlogo and Pythagoras to conduct new behaviors will be necessary to support human decision making experiments. In particular, the ability to model and capture decisions for reserve force commitment and retreat must be developed, and the ability to model chemical weapons and unit reactions to chemicals must also be developed.

The mimicry of a specific human's decision making style has great application throughout human endeavor. For the military, this work can have significant impacts on the modern battlefield in the development of better semi-automated forces, the command and control of robots and machines, and a host of communications and command and control applications.

> *"As far as the laws of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality"*
> *~ A. Einstein*
>
> ***"I have no particular talent. I am merely inquisitive."***
> ***~ A. Einstein***

[1] For more information contact: Wayne Stilwell, wayne.stilwell@us.army.mil, 804-765-1768, USA

Plenary Sessions

Presentations from Around the World

The plenary sessions at a Data Farming Workshop provide an opportunity for attendees to take a break from their team's high density cogitation and interaction and to go into "input" mode. IDFW13 provided plenty of options for attendees to expand their knowledge of other community activities and expertise. The presentation material from the plenary sessions, as well as team in- and out-briefs, team reports, models and data can be found on the IDFW13 CD or website (see inside back cover for details).

Keynote: Complex Adaptive Systems Applications to Test and Evaluation

Colonel Eileen Bjorkman

Joint Test Director

Joint Test and Evaluation Methodology (JTEM)

Plans, processes and methods for testing, evaluation and capabilities assessment of complex adaptive systems of systems in a joint mission environment using Data Farming.

Data Farming for New Members

Gary Horne

Referentia Systems

An expansive introduction to the concepts and methods of Data Farming.

PAX Model

Gunther Schwarz

EADS, Deutschland

PAX is a flexible tool for the modeling and simulation of Peace Support Operations. This presentation summarizes the PAX development motivation, PAX's functions and capabilities, and plans for the future of PAX.

MANA Post Processing Tools

Ted Meyer

An overview of post-processing tools developed to aid in the analysis of Mana output. The tools provide time-series of casualties, and a killer-victim analysis matrix.

Pythagoras Applications

Lawton Clites

Referentia Systems

Overview of the Pythagoras modeling environment and its applications. See "Pythagoras: An Agent-based Simulation Environment" in this issue for more information.

ABM Report

Ole-Jakob Sendstad

FFI Forsvarets Forskningsinstitutt, Norway

A summary of FFI's study to date of the utility of agent-based modeling for operational analysis.

SPYSE Agent Based Framework

Andre Meyer

TNO, Netherlands

SPYSE is an open-source advanced multi-agent simulation programming framework. This presentation overviews agent programming techniques and concepts in the context of SPYSE.

Design of Experiments

Susan Sanchez

Naval Postgraduate School

An overview of the NPS's SEED Center's research into effective experimental design and the tools and resources needed to support it.

Automated Red Teaming

Dave Ang

SMTS, DSO National Laboratories, Singapore

A description of the DSO developed Automated Red Teaming frameworks and its capabilities and some applications.

Data Farming Competition I

Gary Horne and Ted Meyer lead two teams in head-to-head data farming of symmetric Mana scenarios. Both teams attempt to select the parameters that lead to victory!

Using NetLogo in the the Data Farming Environment

Matthew Koehler
The MITRE Corporation[1]

NetLogo is a freely available agent-based modeling environment being developed by Northwestern University's Center for Connected Learning (ccl.northwestern.edu/netlogø). NetLogo is an excellent environment for creating simpler or smaller-scale agent-based models or prototyping more complex models. NetLogo's strengths include using a very easy to learn and flexible scripting environment, a GUI interface that handles all the necessary code for you, and a section dedicated to documenting your model, and a very large sample model library with very good documentation. The down side of NetLogo is that you must create all functionality you desire to have in the model, which can be time consuming if you have a great deal of complicated behaviors. Furthermore, NetLogo is written in Java and its scripting language is only semi-compiled (some primitives are compiled into Java byte-code, other primitives are interpreted), which can lead to some performance issues if your models is very large or involves a great deal of computation. Finally, NetLogo is compatible with the Data Farming and cluster computing methods and tools created by Project Albert and its collaborators.

Structure & Features of NetLogo

NetLogo contains three basic types of entities within it: turtles (agents), patches (the landscape), and the observer. Turtles can be subdivided into different classes (called breeds). All of these entities can run code and interact with each other and with other types of entities. Variables can be assigned globally, all entities having access to them, or specifically (in which case only the specified group has access to that variable).

This structure gives modelers a great deal of flexibility when creating a model. Different types of agents can have a common set of variables as well as unique sets used to create specific behaviors. For example: all agents that move around on the ground could have a common set of variables used for movement, however, they could all have unique variables associated with other capabilities. Discretizing the landscape into autonomous regions (patches) all of which can execute code and maintain a unique internal state presents many opportunities to the modeler. For instance, one can import a .bmp image file (perhaps created with GIS data) which the patches can use to set internal parameters. Then the agents moving around on the patches can query the patches for the values of the parameter and then behave appropriately.

The NetLogo user's interface is also fairly flexible and very easy to use. You can create sliders, buttons, switches, and monitors with drag-and-drop convenience. Plots can also be created very easily; however, they will require a few lines of code in the procedures also. NetLogo can also print output to a window on the user interface or print to a file. Finally the entire NetLogo state (the value of all parameters and the states of all agents, patches, etc.) can be saved and reloaded.

Although the NetLogo scripting language is not extensible per se, NetLogo can be setup to access external Java programs so one could create external programs to handle particularly computationally intensive procedures or create NetLogo models that update themselves based upon a web service or database call. NetLogo also has the ability to act as a server and receive input from Texas Instrument calculators or other computers. This gives modelers the ability to create human-in-the-loop models. The next section contains details on how to create a NetLogo model that is compatible with the Data Farming Environment (DFE). Full detail can be found in Koehler (2005).

Flow of the DFE with NetLogo

The general flow of the system is as follows: 1. create a NetLogo model following a set of conventions; 2. parse the NetLogo file into an input XML file; 3. using the XStudy tool, pick the sliders, choosers, or switches that will be varied during the runs; 4. use Old McData (OMD) and Condor to kick-off the runs and collect the data (this can de done on a single machine or multiple machines). All of the software is written in Java and should work on any machine

[1] The author's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the author.

with a Java Virtual Machine. This system, though not perfected, is robust enough to handle the pressure of workshop demands—including thousands of runs done remotely on clusters in different countries. We have successfully run Netlogo in two different cluster computing environments: the Maui High Performance Computing Center and on a cluster maintained by the Singapore Defense Science Organization. The system is capable of handling any sort of experimental design from full factorial to Nearly Orthogonal Latin Hypercube. Furthermore, OMD has post-processing capabilities that can be used with evolutionary programming algorithms and other types of user defined algorithms to create a more dynamic study.

In the following discussion we will examine the conventions necessary when putting together a Netlogo program, as well as general instructions for the use of the other software used for the multiple runs; however, it is assumed the reader is familiar with Condor. The software discussed in this paper is, or soon will be, available on SourceForge. Alternatively, the software is available from the authors. Condor is available from its developers at: http://www.cs.wisc.edu/condor/. NetLogo is available from its developers at: http://ccl.northwestern.edu/netlogø.

Setting up the NetLogo Model

The current system requires certain features within the NetLogo model.[1] These requirements will be discussed below. These requirements have minimal impact on the structure of the program and on the speed of execution and are designed to allow an external Java program to start the model, set parameter values (sliders, choosers, and switches), start and end a run, and collect output data (both end of run and time series). In general, the wrapper starts Netlogo and loads the model, and then it tells Netlogo to iterate a certain number of times. At the end of the requisite number of iterations, output data is collected and the Netlogo run is terminated.

Global Variables

The model needs three global variables: **`stopped`**, **`filename`**, and **`clock`**. These are used by the external program to run NetLogo, keep track of output data, and allow the modeler to control the behavior of their model separately from the Java wrapper.

The `Setup` Procedure

First, the NetLogo model must have a procedure called **`setup`** to instantiate the model and to prepare the output files. At a minimum it will need the following lines of code:

```
to setup
set clock 0
set stopped false
setup-file
end
```

Every time the model is run it will be in a newly started instantiation of NetLogo; therefore, one is not required to set variables (unless they need to be something other than zero). However, you may want to clear values and set others so that you will know exactly how the model is starting up. If you do clear values DO NOT use the command **`clear-all`** or **`ca`**. If you want to clear values use commands such as **`clear-turtles`**, **`clear-patches`**, **`clear-all-plots`**, **`clear-output`** and then manually set the variables. If you use **`clear-all`** you will set the variable **`filename`** to 0. This will cause problems later on, when the output from all the runs is collected because all the files will have the same name. The batch version of NetLogo is run by a Java program that will set certain parameters, among those is **`filename`**. Once NetLogo is started, the Java program will call the **`setup`** procedure. If **`setup`** then resets the value of **`filename`** Condor and OMD will have trouble keeping track of the output files because they will all have the same name. A more comprehensive version of the setup procedure that includes resetting of values is below:

```
to setup
ct
cp
clear-output
clear-all-plots
;; manually set all variables
set clock 0
set stopped false
setup-file
end
```

The setup-file procedure is very short and could be called from within the setup procedure. It is recommended to keep them separate for clarity. A sample of this procedure is below:

```
to setup-file
 ifelse filename = 0
 [file-open "Your_BackUp_Name_Here.csv"]
 [file-open filename]
end
```

[1] It should be noted that this system was created in 2005; the current version of NetLogo may require slight changes to the code described herein. Please contact the author with questions.

This procedure allows you to run the NetLogo program inside the cluster computing environment or in the standard NetLogo program for testing purposes. This works because it checks to see if the variable **filename** has been set by the Java wrapper program. If it has not been set by the Java wrapper, it will open a default file of your choosing.

The Go Procedure

All models must also have a **go** procedure. The **go** procedure is a little different than the usual NetLogo program. First of all, the procedure must be called "go." Second, the wrapper runs the NetLogo program by asking it to step a certain number of times. Due to this structure, it is important to "protect" your runtime code by nesting it inside an **if** statement that returns true if stopped is false. Sample code for the **go** procedure can be found below:

```
to go
set clock clock + 1
if not stopped
  [
  ;;runtime code goes in here

  if 'stop condition is true'
       [do-file-print close-files set stopped true]
  ]
end
```

By nesting the runtime code inside the **if** statement, the wrapper can run the model any number of times without any potential damage to the output after the stop condition is met. For example, if you have set up the wrapper to run your model 6000 times but you have a stop condition that is triggered at time step 3500, the wrapper will continue to tell your model to step another 2500 times. If you generate output every time step and do not protect it, then you will end up with another 2500 lines of output. As your stop condition could be triggered at different times it could be very difficult to fix your data post run. It is also important to segregate any end-of-run printing procedures from the file close procedure. Once the wrapper is done stepping the NetLogo program, it will tell the program to **close-files**. Therefore, you must have a procedure in your program that is called **close-files**. If this procedure includes anything other than file closing code, it may cause a problem as it will be run anytime files are closed. If you close files anytime the stop condition for your model is true, then any other code will be run every time the wrapper steps your program once the stop condition is met (this is not an issue if you protect your runtime procedures in the aforementioned **if** statement and make the **close-files** procedure exclusively devoted to closing files). However, this does require that your model have a stop condition that will be triggered at least one time step before the wrapper ends the run because the wrapper will simply stop telling the program to step and then call the **close-files** procedure. Sample code for the **do-file-print** and **close-files** procedures can be found below:

```
to do-file-print
file-print "output goes here"
end

to close-files
file-close-all
end
```

Also, there is no post-processing currently associated with NetLogo runs, so if you want something in the output file, such as input parameters, you must write it there in the program (in something like the **do-file-print** procedure). This file will be a single line if you are only collecting end of run data. If, however, you are collecting time series data, this file may be very large.

The above represents all the requisite code for a NetLogo program to set it up for cluster computing. Now, part of the utility of cluster computing is being able to run a model many times with different parameter values. The system we have developed can run NetLogo programs many times and change parameter values. However, the parameters that will change need to comport with a set of standards. First, they must be sliders, choosers, switches, etc. and, therefore, must appear in the "Interface" tab of the NetLogo environment. Second, these parameters may not contain any special characters like ?, %, $, *, and so on. Third, they may be set to numeric values only--no strings. For example, a chooser with the values: High, Medium, and Low would not be acceptable. The chooser should have values such as: 1, 2, and 3 which could then be mapped to High, Medium, and Low in the procedural part of the NetLogo model. This does not preclude other parameters from taking on any values you wish and having special characters in their name…these standards only apply to parameters values you wish to change in an automated fashion.

References

Koehler, M. and S. Upton, B. Tivnan, Clustered Computing with NetLogo and RepastJ: Beyond Chewing Gum and Duct Tape. Proceedings of Agent 2005. Argonne National Lab, Chicago, IL. (2005).

Wilensky, U. NetLogo. http://ccl.northwestern.edu/netlogø. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL. (1999).

Recent Developments in the MANA Agent-based Model

Gregory C. McIntosh
David P. Galligan
Mark A. Anderson
Michael K. Lauren - Contact[1]
Defence Technology Agency, Naval Base, Private Bag 32901, Devonport, Auckland, New Zealand.

Agent-based models have recently gained in popularity for modelling military operations. They purposefully leave out detailed physical attributes of the military entities concerned if this is not expected to have any bearing on the study at hand. This allows scenarios to be run relatively fast, over many excursions in order to discover unique situations or tactics where friendly forces can achieve dominance over an enemy. Another key feature of agent-based models is that, although the one-to-one interaction between various agents and their environment may be quite simple, the combined effect of many agents interacting can lead to complicated group dynamics and emergent behaviour. In this regard, agent-based models have the potential to represent the more chaotic and intangible aspects of military conflicts [1].

MANA (Map Aware Non-uniform Automata) is an agent-based model developed by the Operations Analysis group at Defence Technology Agency in New Zealand [2]. MANA has been used in a number of studies: modelling civil violence management, the modelling of maritime surveillance and coastal patrols, investigating modern warfare as a complex adaptive system and a range of studies carried out at the bi-annual Project Albert meetings. MANA is being used by a number of military colleges and defence science establishments amongst the TTCP nations and has also been used for various Master's theses at the Navy Postgraduate School in Monterey.

Amongst the available agent-based models, MANA has a number of strengths. It is user friendly with an easily navigable user interface. Scenarios can quickly be edited 'on the fly' during their development. Being a pre-compiled executable, MANA runs relatively quickly so that many scenario excursions can be run through within a reasonable space of time. Furthermore, MANA has a built in data farming capability, allowing a scenario's parameter space to be rapidly explored. In line with 21st century warfare concepts, MANA can simulate communications links for information sharing between groups of agents so that aspects of network centric warfare (NCW) may be studied. To model terrain features, MANA makes use of colour-coded bitmaps. This has the advantage that terrain features can quickly be edited 'on the fly' while a scenario is being developed.

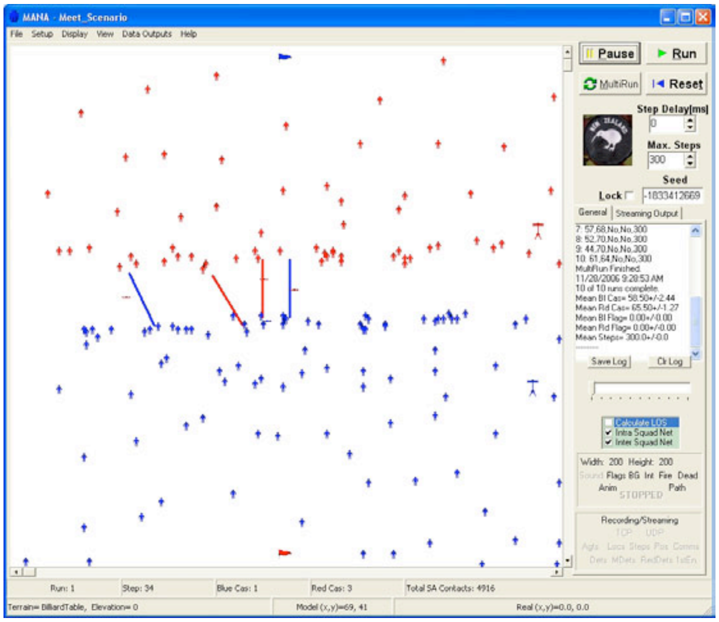


Figure 1: MANA 4 screenshot showing a conventional two-sided battle.

Development of MANA commenced in approximately 2000, with initial inspiration coming from Ilachinski's agent-based model, ISAAC. It was decided that MANA should be engineered to suit the NZDF needs. For example, with the NZDF being a small-scale expeditionary force, a flexible low maintenance model is more appropriate than a hugely detailed military simulation such as JANUS or JSAF. MANA has been developed with Delphi as the programming language. This language has a user friendly quality, with good readability in terms of exchanging code

[1] Naval Base, Private Bag 32901, Devonport, Auckland, New Zealand

between software developers. By having good control of the source code we can quickly add new features to MANA as various studies require. MANA development has proceeded in several stages as follows.

MANA 2 (2002/2003): This represents the first fully usable version of MANA. It comprises the core of the present-day model with all essential movement weightings and terrain editing features in place. Sensor and weapons characteristics can be represented using a simple cookie cutter scheme or with tables of range-dependent probabilities. In MANA, agent properties are specified for groups of agents defined to be squads. The concept of a shared situational awareness (SA) map for squads is included in MANA 2 which allows rudimentary aspects of NCW to be modelled. For example, sharing information amongst squad members or the delay in information getting from sensors and onto a squad's SA map can be simulated.

MANA 3 (2003/2004): This is a fully developed version of MANA which is now in wide-scale use. It has been built from MANA 2 with several refinements. As well as the squad SA map, communications links and information sharing between squads has been explicitly modelled so that aspects of NCW may now be fully explored [3]. Accordingly, additional movement weightings have been added so that agents can respond to information received from other squads. A data farming capability has been included to fully automate the mapping of a scenario's parameter space. Special aircraft movement and search algorithms have been added for search and patrol scenarios. (This feature was subsequently reverse engineered back into MANA 2.)

MANA 4 (2005/2006) MANA 4 has recently been developed and is soon to be released. At first glance MANA 4 appears similar to MANA3 but includes several new features. A data streaming capability has been added so that MANA can now be used for human-in-the-loop experiments. Indeed, preliminary studies at RAND using this feature have proved promising. The main battlefield display can be zoomed to allow better control over agent placement and terrain features for larger, more intricate scenarios. A genetic algorithm has been included to automate scenario development and as a research tool [4]. A data analysis tool has also been added so that results from multiple scenario runs may be post-processed into a number of time-dependent averages and graphed. Finite sensor and weapons apertures are now included. Correspondingly, angular dependencies have been added to the movement algorithms such that agents now include a direction of facing and can look around to spot enemies. Furthermore, there are movement weightings depending on the enemies' direction of facing so that agents can choose a direction of approach towards enemies. Squad formation shapes have also been added.

MANA 5 (2006/2007): This is currently in development. Here, we are attempting somewhat of a paradigm shift while still maintaining a distilled agent-based aspect to the model. There will be the ability to define larger battlefields which extend beyond the viewing region in order to cater for larger scenarios. Hence, full zooming and panning ability will be available to observe how a scenario is playing out in various areas of the battlefield. Instead of the cell-based movement algorithms of previous MANA versions, a vector-based movement algorithm is being experimented with. Here, a weighted vector is calculated towards targets of interest and a movement 'force' is applied to the agents. It is anticipated that a vector-based movement system will open up the flexibility to develop more intelligent agent behaviour. For example, there is an interest in intelligent path finding for agents to navigate urban terrains without becoming lost or stuck in corners.

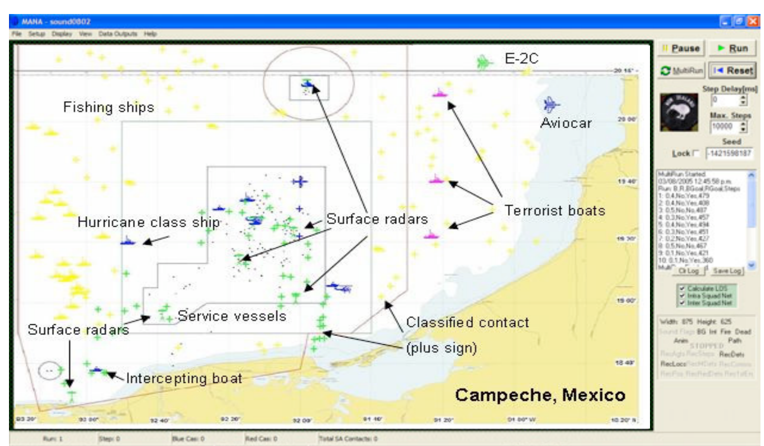
REFERENCES

[1] A. Ilachinski, *Artificial War: Multiagent-Based Simulation of Combat*, World Scientific Publishing, 2004.

[2] M. K. Lauren and R. T. Stephen, 'Map-aware Non-uniform Automata (MANA) a New Zealand Approach to Scenario Modelling', *Journal of Battlefield Technology*, Vol. 5, No. 1, 2002.

[3] D. P. Galligan, 'Modelling Shared Situational Awareness Using the MANA Model', *Journal of Battlefield Technology*, Vol. 7, No. 3, 2004.

[4] G. C. McIntosh and M. K. Lauren, 'Genetic Algorithms Applied to Course-of-Action Development Using the MANA Agent-Based Model', *Journal of Battlefield Technology*, Vol. 9, No. 3, 2006



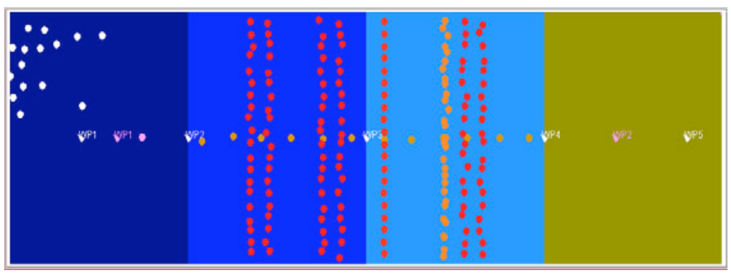
MANA screen shot from a recent NPS Thesis

Pythagoras: An Agent-Based Simulation Environment

Zoe Henscheid - Contact[1]
Donna Middleton
Edmund Bitinas
Northrop Grumman Corporation, USA

Pythagoras

Pythagoras is an agent-based simulation environment originally developed to support Project Albert, a U.S. Marine Corps–sponsored international initiative that focused on human factors in military combat and non-combat situations. Pythagoras enables a user to create intelligent agents and assign them behaviors based on motivators and detractors. The agents can either act as individuals or be loosely or tightly controlled by one or more leader agents. Pythagoras is written in Java, making it platform-independent. It can be run in a supercomputer environment as a batch job, enabling tens of thousands of repetitions to be run in a short time; can be used and run interactively on a PC through a graphical user interface (GUI), or can be run in batch mode from a PC command prompt.



Pythagoras offers the following unique set of capabilities in the area of agent-based simulations:

- Incorporates *soft rules* to distinguish unique agents
- Uses *desires* to motivate agents into moving and shooting
- Includes the concept of *affiliation* (established by *sidedness*, or color value) to differentiate agents into members of a unit, friendly agents, neutrals, or enemies
- Possesses the concept of influences on behavior through color, generic attributes or generic resources
- Allows for behavior-changing events and actions (called *triggers*) that may be invoked in response to simulation activities
- Retains traditional weapons, sensors, communication devices and terrain

A summary of each capability is discussed below.

Soft Rules to Create Individuality

Pythagoras has a feature called *soft decision rules*, which not only assigns each agent its own threshold within the decision variable trade space but also ensures traceability. Pythagoras allows all decision variables to be softened by the user. The approach is to reflect variation between individual agents by establishing a midpoint for the variable in question and then allowing the user to provide a uniformly distributed range around that value. When an agent is instantiated at the beginning of a scenario run, it selects its decision-variable values from the distribution at random. By controlling the spread, agents can be instantiated as very homogeneous (e.g., well-trained, disciplined military troops) or quite heterogeneous (e.g., a crowd of villagers), or some value in between. Each decision variable has its own control, so some aspects of behavior can be tight, others loose. Figure 1 illustrates three alternative degrees of individuality.

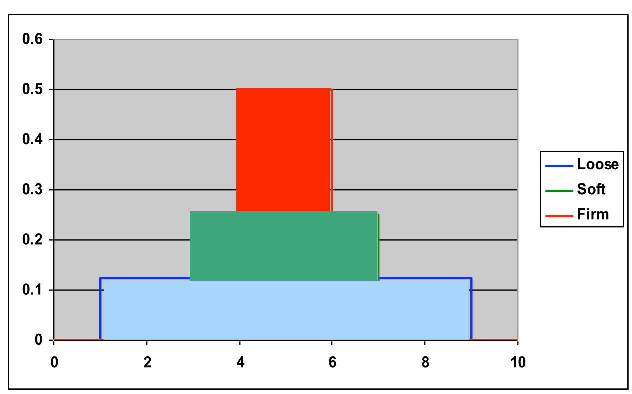
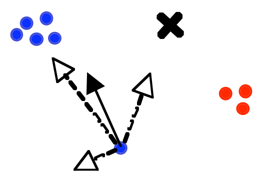


Figure 1: Three alternative degrees of individuality

[1] For more information contact: Zoe Henscheid, zoe.henscheid@ngc.com

In each of the three cases illustrated in Figure 1, the midpoint for the behavior variable is 5. That value might, for example, represent the minimum number of enemies from which an agent would retreat. The red uniform distribution represents a firm, homogeneous behavior variable, for which the range is ±1 around the midpoint of 5. The green uniform distribution depicts a softer behavior variable, for which the range is ±2 around the midpoint. The loosest and most heterogeneous of the three definitions for the behavior variable is described by the blue uniform distribution, with a range of ±4 around the midpoint.

Agent Desires to Move



In Pythagoras, agents can be assigned *movement desires* (listed in the top section of Table 1) to determine their movement paths as a scenario unfolds. During each decision cycle, an agent establishes which desires are active (e.g., too far from a leader), and, if the sum of the *desires to move* exceeds a user-determined threshold, the agent then uses the strengths of the desires to determine a *direction of movement*. If multiple desires are active, they can be adjudicated at the user's discretion by one of four alternative methods:

- Through vector algebra, using direction weighted by desire
- By priority (i.e., the strongest desire)
- At random, weighted by the strength of each desire
- By applying vector algebra for only the two strongest desires

Although the list of existing desires is small, it can be used to represent a variety of behaviors.

Once the agent chooses a direction of movement based on the active desires, the agent reviews the terrain suitability of the selected path. If the terrain is unsuitable because of movement, concealment, and/or protection considerations, the agent first looks to the right and left of the desired path until suitable terrain is identified.

Dynamic Sidedness for Three-Dimensional Affiliation

In Pythagoras, each agent may be assigned a value for each of the three color properties—red, green, and blue—that can be used to establish affiliation among the agents. One, two, or all three of the color properties can be used

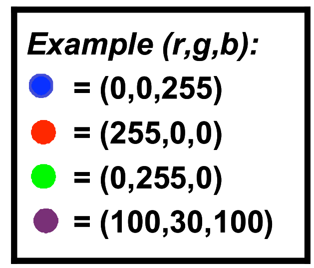


Table 1: Decision variables (desires) used to establish agent movement direction and weapon selection

MOVEMENT	***Toward (if > d) or Away From (if < d) Closest Leader***
	Toward (if > d) or Away From (if < d) Closest Unit Member/Friendly
	Toward Furthest Unit Member/Friendly (if > d)
	Toward Next Way Point (if > d)
	Toward (if > d) or Away From Nearest Enemy (if < d)
	Toward Small Number of Enemy (if # Enemy or Ratio Within d < e)
	Away From Large Number of Enemy (if # Enemy or Ratio Within d > e)
	Toward Objective
	Toward Injured Friend
	Toward Friend (if > d) Needing Fuel, Resource X, Resource Y, Resource Z
	Toward Friend (if > d) Supplying Fuel, Resource X, Resource Y, Resource Z
	At Random
	Continue Same Direction
	Stay in Place If Desire to Move Is Weak
TERRAIN	***Avoid Bad Terrain if (Mobility< or Detect> or Defense<)***
	Prefer Good Terrain if (Azimuth, Mobility> and Detection< and Defense
WEAPON	***Hold Fire/Free Fire***
	Highest, Medium or Lowest Lethality Weapon

d – distance e – enemy

to establish an affiliation. Different agents can use different sets of properties for their affiliation. Pythagoras uses the terms *greenness*, *blueness*, and *redness* to make the properties generic (and to allow for visual display of the property in the scenario playback tool). Each of the three properties can take a value from 0 to 255 (corresponding to standard color-monitor settings used by Java to control image colors). Figure 2 shows an example of both blue and green being used to establish affiliation.

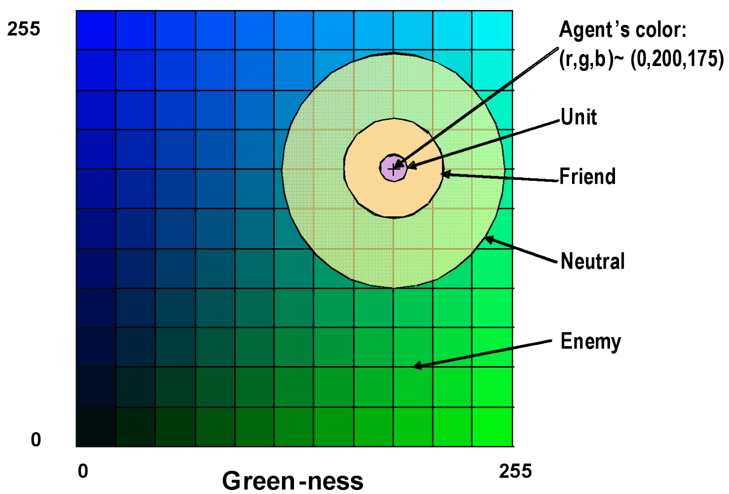


Figure 2: Two-color sidedness used to establish two-dimensional affiliation

Agents with similar color (as measured by either the difference in absolute value or the root sum square of the differences of the active colors) are considered to be members of the same unit. Those whose color is close are considered to be friends. Those whose color is far away are considered to be enemies. Colors between enemy and friendly agents are neutrals. This approach allows for multiple affiliations within a single scenario, as might be found in a crowd; or it can be used to establish command hierarchies, as would be found in a military organization (e.g., slightly different blue uniforms could represent different companies).

Sidedness, or color value, is governed by soft rules at the start of the simulation (agents can be initiated with more or less redness, for example) and can be changed over the course of the simulation by various events and actions. Because not all colors are required to establish affiliation, colors not used for affiliation could be used to represent a different property—for example, fear, hunger, morale, or intelligence. Increases and decreases in that property can alter agents' perception of one another. Alternatively, such a change can cause a behavior-change event, as described in the Behavior-Change Triggers section.

Generic Attributes

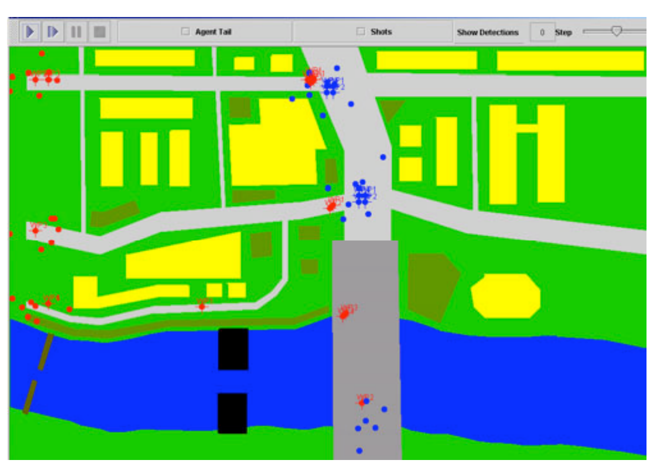
Similar to Sidedness, Pythagoras has three generic agent attributes called *alpha*, *beta*, and *gamma*. These generic attributes act as a supplement to *Sidedness* and as such they do not affect an agent's affiliation/sidedness. The meaning of alpha, beta, and gamma is up to the user to define, based upon the user's scenario. They could be used to represent intangible items such as fear, hunger, and morale or something more concrete such as health or wealth. The generic attributes are also governed by soft rules at the start of the simulation (agents can be initiated with more or less alpha, for example) and can be changed over the course of the simulation by various events and actions. A change in the value of generic attribute can also cause a behavior-change event.

Resources

Additionally, Pythagoras models logistics capabilities, in the form of fuel and three generic resources (X, Y, and Z). The generic resources allow a user to model any type of resource that would be required for their scenario – for example, food, batteries, and medicine. These resources are also governed by soft rules and can be changed over the course of the simulation by various events. Additionally, these resources can cause behavior-change events.

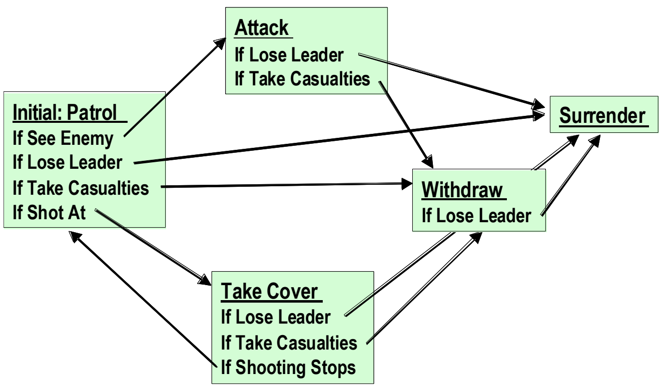
Behavior-Change Triggers

Another feature of Pythagoras is the behavior-change event/action, or *trigger*. When an agent experiences one of the trigger events/actions, the current behavior template is replaced by a new behavior template. This new template is defined by one or more of the following: new movement and shooting desires, new color-change and attribute values, new resources, weapons, sensors, and/or communication devices, and a new set of behavior-change triggers. For example, an agent can be set to walk up and down a street, as if on patrol, but when the agent is shot at, his behavior changes to look for protective terrain, such as a doorway.



The trigger events are (where 'v' is determined by the user):

- Being shot at
- Detecting an enemy (or friend)
- Arriving at a way point or objective
- Friendly casualties (fewer than v% known remaining)
- Loss of leader (no leader within v distance)
- Time step (absolute and relative)
- Red, green, or blue become greater or less than v.
- Alpha, beta, or gamma becomes greater or less than v.
- Fuel, resource X, resource Y, or resource Z becomes greater or less than v%.
- Total ammunition falling below v



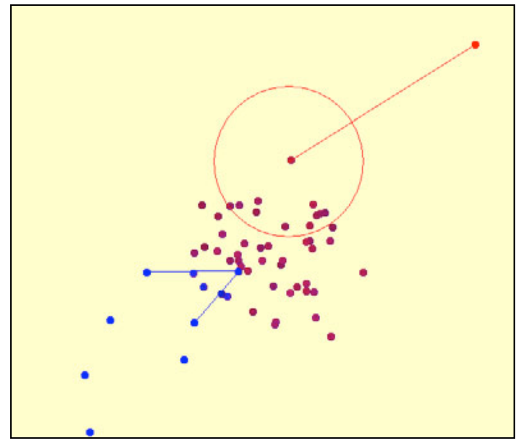
Since the behavior template for an agent includes new triggers, and each template is uniquely named, a series of templates can be constructed to represent a complex behavior tree or network, with agents moving from one behavior to another as a scenario unfolds. Also, since agents are separate objects, different agent types can share the behavior templates. Additionally, there is no practical limit to the number of templates that can be created.

Weapon Options

Pythagoras allows agents to carry as many as three different weapons. Weapons can be either *direct-fire* (which requires a line of sight) or *indirect- fire* (which does not):

- *Direct-fire* weapons have range-dependent probabilities of hit, and of kill given a hit (these factors may be affected by the target agent's vulnerability).
- *Indirect-fire* weapons have circular-error-probable accuracies and lethal radii, both of which may be range-dependant.

An indirect-fire weapon's lethal radii can be modeled using either the traditional Carlton damage function or a "cookie-cutter" blast. The Carlton damage function describes the probability of damage incurred radiating out from a hit point. The cookie-cutter blast causes an equal amount of damage to all objects within the blast radius, regardless of their distance from the impact point. Both weapon types have input rates of fire and limited ammunition.



In Pythagoras, the target may be suppressed for multiple time steps by a hit by a direct-fire weapon or a near miss by an indirect-fire weapon that does not kill the target. Suppressed agents do not move, sense, communicate or shoot, but they recover those capabilities after a user-input amount of time. At the user's discretion, kills can be random (stochastic), deterministic (fractional damage), or a combination of both.

Agents will automatically shoot at enemies within range, unless they are given a "hold fire" command. The user can assign an agent multiple weapons and set the weapon selection criteria to be based on the highest probability of hit (best chance that the target will be suppressed), the lowest probability of kill (for a non-lethal situation), or a medium probability of hit.

Agents can also change each other's colors or generic attributes, through the use of military countermeasures or weapons that can act as paintball weapon. An example of a paintball weapon would be propaganda that alters an agent's affiliation in some way. Propaganda in the form of leaflets can be modeled as a direct-fire weapon aimed at a specific target. On the other hand, a form of public speech or exhortation can be modeled as an indirect-fire weapon, aimed in a general direction and affecting everyone within range.

Multi-Band Sensors & Communication Devices

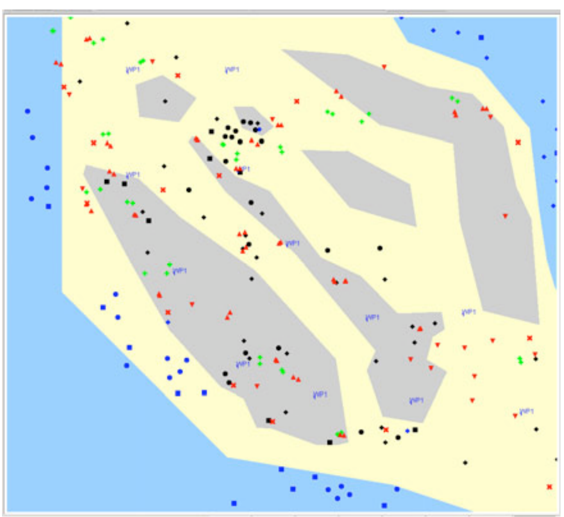
Pythagoras also models sensors and communication devices. Each agent may have up to three sensors, each of which operates in a specific signature band (labeled A, B, or C). The sensors have a range-dependent probability of detection, which is modified by intervening terrain and the target agent's detectability factor. For example, signature band A could represent the bandwidth of the naked eye and signature band B might represent the bandwidth of an infrared device. A terrain feature, such as foliage, would thwart the naked eye's view, but an infrared device could "see" the thermal image of a camouflaged person standing in the foliage. Such foliage could be modeled in Pythagoras as having a concealment value of 100% in band A and 0% in band B. Sensors also have a field of view and, for modeling humans, a probability that the agent is looking forward (i.e., in the direction of travel), to the side, or to the rear.

Similarly, an agent can possess up to three communication devices, each of which operates either via line-of-sight or in a broadcast mode, and allows the agent to use the devices to talk, listen or both. Each communication device also operates via a specific channel or channels (up to three allowed).

Representation of Terrain

All agents exist in a user-defined playbox of up to 1000 × 1000 pixels. In the playbox, a user can create terrain features —polygons that have a floor, a ceiling, and factors for mobility; concealment in each of the three signature bands; and protection that reduces the weapon's effectiveness. Currently, a pixel can be associated with at most one terrain feature. Agents can either move on the terrain or floor or operate at an altitude above the terrain.

Sample Scenarios



Pythagoras has two major advantages:

- It can be used quickly to assess various simple scenarios.
- It can be combined with more complex and detailed models to provide insights into situations that may not be possible with other simulations.

Rapid prototyping of scenarios, weapons, sensors, behavior patterns, etc., allows users to undertake complex problems in new ways without the burden of months of software (and scenario) development. Table 2 provides a representative list of scenarios that have been developed within Pythagoras.

Table 2: Representative Pythagoras Scenarios

Historical Analysis (USNA):	• Battle of Midway
	• Operation Market-Garden
	• Battle of Ia Drang, Viet Nam
Thesis Topics (NPS):	• Sea-Based Logistics
	• Convoy Protection
	• Unmanned Surface Vehicles
	• Future Force Warrior Small Combat Unit With Non-Lethal Weapons
	• Emergency First Response To A Crisis Event
	• Effectiveness of Non-Lethal Capabilities in a Maritime Environment
	• Exploration of Force Transitions in Stability Operations Using Multi-Agent Simulation
Cadet Capstone Project (USMA, West Point)	• Sensor Placement on the Battlefield
Study on Actions Off of Bay of Biscay (AFIT)	• Anti-Submarine Warfare
Thermobaric Weapons Assessment (MCCDC)	• Urban Environment
MAGTF Optical Requirements (Night Vision Lab/Ft. Belvoir)	• Peacekeeping at Night in Urban Environment
Shallow Water Obstacle Clearing (MCWL)	• Use of JDAM and Robotics
Environmental Concerns (MITRE)	• Spread of Hemlock Wooly Adelghid
Homeland Defense (Northrop Grumman)	• Pre-proposal Analysis
Terrorist Development (Sandia Labs with Marc Sageman of University of Pennsylvania)	• Conceptual Model of Human Factors
Soldier Technology Development (Northrop Grumman)	• Less Than Lethal Technologies for Urban Combat
Securing Targets of Interest (JCS/J-8, DMSO)	• Competition Between Multiple Factions
Marine Expeditionary Rifle Squad (MERS) (USMC/OAD)	• Trade-offs Between Different Equipment

Source for Acquiring Pythagoras

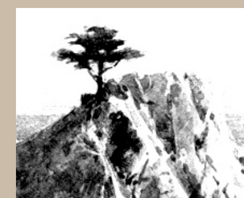
Pythagoras can be acquired from the Simulation Experiments & Efficient Designs (SEED) Center for Data Farming located at the Naval Post Graduate School (NPS). The Northrop Grumman developers listed below can be contacted for further requests for information or questions about Pythagoras:

International Data Farming Workshop 14

When: 25-30 March 2007

Where: Portola Plaza (hotel website is http://www.portolaplazahotel.com).
Monterey, California, U.S.A. ***Please register by 2 March 2007.***

The Portola Plaza Hotel is Monterey's premier waterfront hotel, set within walking distance of all the attractions - from Cannery Row to the Aquarium, historic Fisherman's Wharf, and downtown Monterey. With amenities that match the spectacular bayside location, the hotel offers luxurious accommodations, an award-winning spa, heated pool, intriguing shops and galleries, and many choices for dining by the sea. Guest rooms include residential style furniture, oversized work desks, high-speed Internet access, digital cable and video games; many rooms feature panoramic views of Monterey Bay. Portola Plaza is your headquarters for the International Data Farming Workshop 14 AND for world-class golf, world-class shopping in Carmel-by-the-Sea, Big Sur and Seventeen-Mile Drive, Monterey County wineries and much more.



Tentative Agenda

Sunday 25 March: Opening dinner, Portola Plaza, gather at 6 o'clock
Monday 26 March: Opening briefings, 0800 Portola Plaza
Tuesday 27 March - Thursday 29 March: Work in teams at NPS, Concurrent morning plenary sessions
Friday 30 March: Team Briefouts and Closing Ceremony at Portola Plaza Start at 0800, finish by noon

Call for Team Leaders / Plenary Speakers:

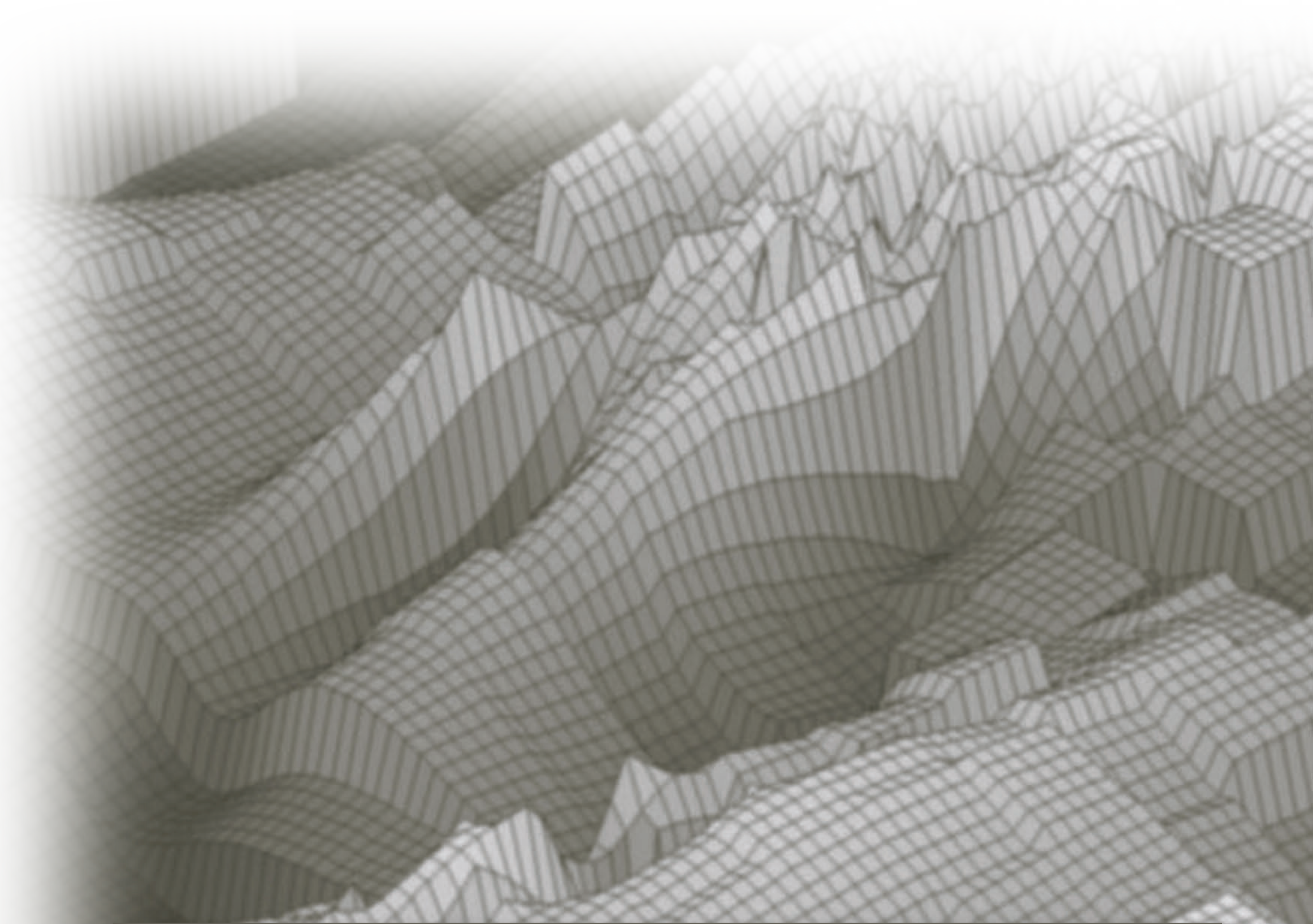
Please email datafarming@verizon.net with your choice of teams and if you want to lead a team or present a plenary briefing.

Conference Fee:

The early bird registration fee is $425. The early bird cutoff date is 15 February, after which the regular cost of $495 will be in effect. You can register online (using a credit card) using our secure server which you can access through our http://harvest.nps.edu site under international workshops. *Included:*

- Conference rooms
- Breakfasts, lunches, coffee/tea/soft drinks
- New one-year membership card with quote
- Opening dinner
- CD with conference materials
- Fun

The Data Farming CD/DVD, if provided, will be attached here. For addtional copies of the CD or of the Scythe please contact Ted Meyer (tedmeyer@mac.com)



Scythe - Proceedings and Bulletin of the International Data Farming Community

Seed Center for Data Farming
273 Glasgow Hall ~ Naval Postgraduate School ~ Monterey, CA 93950

Issue 1 - Workshop 13