Proceedings of the 1993 Winter Simulation Conference
G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles (eds.)

MODELING COALITION WARFARE: A MULTI-SIDED SIMULATION DESIGN

Kevin Brandt

Joint Warfare Center
246 Hartson Street
Hurlburt Field, Florida 32544-5254, U.S.A.

Ellen Roland

Roland & Associates
500 Sloat Avenue
Monterey, California 93940, U.S.A.

ABSTRACT

This paper captures high-level design imperatives and solutions used to expand a two-sided model to an n-sided model to premiere interactive, multi-sided, coalition-warfare simulations. We focus on general considerations with impact beyond the bounds of a specific model.

The designs discussed are incorporated in version 1.85 of the Joint Theater Level Simulation (JTLS) and are being tested with a ten-sided Southwest Asia scenario.

This U.S. Department of Defense model now supports a database-defined number of sides ranging from two to ten. Each side maintains an independent intelligence perception of the theater of operations and a designation of its friends, its enemies, and any neutrals. Multiple sides can align in "grand coalitions" or segregate into flexible groups to engage in multi-sided conflict.

All sides are split into a variable number of dynamic factions. Each faction has specific attributes profiled by its Battlefield Operating Systems. Each faction holds a unique set of units and targets. And, they can subdivide, merge, or transfer to another side.

1 INTRODUCTION

Coalition warfare requires extensive functions beyond an enabling multi-sided combat model base. The results achieved in the pioneering effort to develop the capability in a combat model establish a milestone for simulations.

After a brief model overview, we capture high-level design requirements and constraints. Next, we detail the transition of an established two-sided model to a multi-sided, coalition-warfare simulation. Then, we end with applications that have been tested or projected.

The design team started with a broad statement-of-need approved by the Configuration Control Board. Within six months, the team devoted more than fifteen man-months to develop both high level and detailed designs. The scope of change and level of effort to transition the model to a multi-sided coalition warfare simulation was substantial. But, compelling needs justified the effort.

2 MODEL OVERVIEW

JTLS models combat and conflict at the operational level of war with tactical fidelity. Stochastic and deterministic processes drive the interactive simulation. An integrated, mixed-level resolution paradigm leverages critical events and systems to simulate joint-combined coalition force operations (including air, land, sea, and SOF missions) and noncombatant actions across a theater of operations.

Major components of the model are summarized in the appendix and detailed in JTLS manuals. (DISA 1992).

3 MULTI-SIDED IMPERATIVES

The first and most important necessity in a multi-sided model is usability. Tractable data must drive a reliable, responsive, robust simulation.

Coherent data sources, data structures, data definitions, data management, and data security are essential. Links to and use of standard data items and sources should be a major objective. Overall data structures must not grow as an exponent of the number of sides. Data filtering and partitions are essential...especially for distributed models.

If open, coherent, flexible logic is not used or if side-specific doctrine is embedded, reliability will decrease and quality assurance test cases will expand exponentially. Code and data costs can quickly become overwhelming.

Level of detail directly impacts on performance. Details significantly in excess of the needed output resolution absorb resources, like critical CPU and I/O bandwidth, without contributing to the effective endstate. Thus, mixed resolution in large scale models is indicated.

Concurrent conflict management between all sides is essential. Rules of engagement must be both selective and dynamic. When engaged, force-on-force-on-force fights must produce robust results. Credible results require that information provided to each side be linked to available intelligence resources. This information should impact on conflict results via friend-or-foe identification.

4 COALITION WARFARE IMPERATIVES

Simulation of coalition warfare requires more that just multi-sided conflict analysis. Models must retain explicit visibility of political changes and policy shifts and must track the execution and results of key coalition warfare operations. Concurrently, the potential for change in our doctrine in the face of dynamic threats favors models that enable a range of possible operations without imposing a set doctrine across the model. Additionally, tractable data must be available at appropriate security levels.

Other capabilities are highly desirable, but may not be essential. One example: technologies that allow model input and results to be distributed to dispersed sites while retaining discrete data partitions are preferred. Another: ease of use is generally improved whenever a model can be linked to established command and control systems to provide results and reports in "standard" formats.

We advocate that implementing multiple sides alone is not sufficient to meet coalition warfare simulation tasks. But, multi-sided simulation is a necessary capability .

5 MULTI-SIDED SIMULATION

5.1 Design Constraints

First, our design team developed the basic concepts of coalition warfare simulation absent comprehensive Joint doctrine. Thus, our concepts had to be flexible and the techniques adopted had to support doctrine potentials.

Second, given established needs to drive staff training exercises and plan analysis, designs stressed reliability, model speed, and robustness. Procedures favor reliability over detail. Database preparation requirements and access to verifiable data also restricted fidelity. And, a runtime performance objective time ratio of 4:1 limited options.

Third, legacy map displays restricted the options for a unique graphics color to represent a side. This constraint imposed the ten-side limit.

Fourth, legacy data structures limited design options. SIMSCRIPT entity structures, used by the model's main events program, favored some designs and made other options too costly.

Finally, established timelines, budgets, and a pending transition to POSIX standards forced other trade-offs.

5.2 Design Structure

An object hierarchy paradigm enables coalition warfare dynamics and multi-sided simulation. At the top, n-sides are defined for each scenario. All sides contain zero or more factions. Each faction holds a unique set of units and targets.

Table 1: Object Hierarchy and Inheritance

SIDES	↳ inherit color	↳ inherit prototypes	↳ inherit prototypes
hold →	FACTIONS		
	hold →	UNITS	
		hold →	TARGETS

Throughout the structure, subclasses and categories of objects inherit characteristics from a base class, the side, and all intermediate classes. Example: factions inherit color from side (of current affiliation) while units inherit common characteristics, by prototype, from faction.

Table 2: Example Hierarchy

SIDES	FACTIONS	UNITS	TARGETS
UN Forces	Coalition	CJTF HQ	C3I ...
(BLUE)	US	1/17 Cav	Cbt Systems
	FR	3d UAV Co	Sensors...
	US	NTPS MEB	Ships
	UK	HMS Bristol	Jammers...
	EG	380 ADA	SAMs ...
Regional Forces	SA	3d Wing	Runway Shelters ...
(GREEN)	KU	44 FS	Aircraft ...
Non-combatants	Civilians	City Center City Airport	C3I : TV Runway ...
(WHITE)	Port Auth	Docks 1-12	MHE ...
	Airlines	Delta	Aircraft
	POWs	Work camp	Supplies ...
	Defectors	OPFOR 23	People ...
Threat Forces	Regular Army	SCUD Btry Infantry Bn	SSMs Cbt Arms...
(RED)	Security Forces	Border Unit Int. Police	Sensors ... Minefield
Neutral	none	none	Tunnel ...
(BLACK)	none	none	Bridge ...

5.3 Design Details

5.3.1 Sides

Each side is a unique collection of objects. These inherit their side name and color. All objects on a side share intelligence and maintain a common perspective of other objects. Each side has a specified leader who establishes a uniform relationship -friend or foe- between objects on her side and objects belonging to each of the other sides.

Sides are base objects in JTLS - permanent entities in SIMSCRIPT (CACI, 1991). Each scenario sets the number of sides (permanent entities are not removed during execution). We enhanced flexibility within these bounds by allowing sides to start as open sides - i.e. void of objects (factions, units, or targets). The number of sides and their other attributes are database parameters. An alternative design to represent sides as dynamic objects (temporary entities) was rejected as too burdensome for database development.

Each side has a unique name, side number, and color. The name is a descriptive text string. Side number provides consistent data representation for internal uses. And, color organizes graphics display. In the absence of established doctrine, we selected color for graphic identification as a logical extension of convention.

All sides have a perception of the world shared by all objects on the side. For example, all BLUE units share common data on GREEN units, RED units, and GRAY units... . This simplification limits computer memory requirements but enables unique perception for each side.

Only select object attribute data are maintained for each perception. Currently, we limit perceived (versus truth) data for each object in the game to the owning side (color of an object), owning faction, location, strength, posture (for units), force distribution (for units), supporting units (for units), owning unit (for targets), and time and source of last data update. Planned improvements will expand the number of perceived attributes for units and targets.

Given a side's common perception in the model, all objects on a given side now share intelligence. While the intelligence process may differ between factions within a side, all units automatically share the resultant products.

Finally, each non-empty side has a leader. This leader serves as the highest command and control echelon for the side and is the default recipient of some reports. This leader establishes a relationship matrix for the side.

Side relationships may be dynamic and asymmetric. Relationships impact on event results. (Kelleher, 1993)

Table 3: Side Relationships

	BLUE	GREEN	RED	WHITE
BLUE	friend	friend	enemy	neutral
GREEN	suspect	friend	enemy	suspect
RED	enemy	enemy	friend	suspect
WHITE	suspect	friend	suspect	friend

The current limit on the number of sides is ten. This hardware and human factors restriction is enforced by software restraints. Hardware requirements are driven by the number of sides and number of objects in the game. Each side requires a memory-resident perceived database external to the central "game truth" database. The size of each of these is dependent on the number of objects and attributes that need to be maintained. In addition, each side requires one usable, unique, graphics-display color. Legacy display systems dictate an absolute constraint of ten colors. Moreover, human discrimination of color and shape portends to be the long-term constraint on the effective number that can be handled in an interactive simulation. Given hardware improvements and human interface modifications, ten side restraints can be lifted.

5.3.2 Factions

The design leverages special objects to represent countries, political or ethnic factions, and economic enterprises. Each of these objects, identified as factions, represents a unique collection of units having a common language, integrated logistical infrastructure, compliant political association, and uniform class characteristics. A non-exclusive mix of prototypes define each faction's capabilities within an extended set of battlefield operating systems (BOS). Each faction belongs to only one side, but these dynamic factions can change sides.

Faction prototypes are based on the Army's Battlefield Operating Systems. Minor changes cover extensions for Joint-Combined operations and non-combatant forces. An additional prototype defines combat systems used by each faction. This scheme is based on one proven by elements of the 5th Special Forces Group (Airborne) and SOCCENT during Desert Shield and Desert Storm.

Air Control Prototype: extends a standard Air Defense BOS. This prototype establishes air control and airspace management capability of the faction. Data also defines the faction's ADA and IFF capability.

Command and Control Prototype: defines densities of the combat systems under different unit postures over all terrain types. Potential extensions include IFF of ground and naval forces and allocation of fires. Expansion could capture leadership cohesion, policy implementation, and decision processes.

Communications Prototype: sets organic capability for commander to subordinate communication links. In addition to tactical links, this prototype includes theater assets, satellite access, public communication networks, public media, and informal communications networks.

Fire and Lethality Prototype: establishes basic ability for faction combat systems to acquire, engage, and kill other combat systems. It also assigns scenario dependent data for close-range Lanchester attrition allocations.

Intelligence and Information Prototype: identifies the ability to collect, fuse, and deliver intelligence and/or information. It frames process delays and target detection modifications.

Maneuver Prototype: captures the baseline maneuver capability. Time delays for units attacking, delaying, or withdrawing are established.

Mobility and Countermobility Prototype: benchmarks the ability to project forces via theater and strategic lift (air and sea). It also establishes delays for units moving through obstacles. Expansion will allow adjustments for times to build and destroy bridges, repair road and rail networks, and construct port facilities.

Survivability Prototype: sets the ability of objects in the faction to survive combat with hostile forces and to endure and recover from area or precision fire weapons.

Sustainment and Logistics Prototype: identifies the ability to sustain forces and provide logistical support. It ties directly into the logistical infrastructure available and the ability to project logistical support to the field.

Combat System Prototype: specifies characteristics, capabilities, and names for the combat systems used by each faction. While not an established BOS, this is an essential faction prototype. It allows units belonging to a faction to have specific names for their organic systems and for them to retain those systems if and when their faction moves to another side. Faction-specific systems allow a few generic combat systems to be leveraged to represent the essential characteristics of many systems.

These ten prototypes reduce database overhead, permit data compression, and retain consistent unit capabilities if and when forces change sided. In SWA demonstration database, we established between two and eight cases for each prototype. These cases define the varied capabilities of the 35 factions that are distributed over nine sides (one side is an empty side). In this example, the CIVILIAN faction is defined by the last case for each prototype.

Table 4: Faction Prototypes

	FACTION					
PROTOTYPE	US	UK	IT	TU	SA	CIV
Air Control	1	2	4	3	1	5
Cmd & Control	1	1	1	1	1	2
Communication	1	1	1	2	3	4
Fire & Lethality	1	1	2	1	2	3
Intel & Info	1	1	2	4	3	5
Maneuver	1	1	4	2	3	5
Mobility / CM	1	1	4	3	2	5
Survivability	1	1	1	2	2	3
Sustain & Log	1	3	4	3	2	5
Combat Systems	1	2	3	6	1	8

(Note: numbers represent case names and are not ranked.)

Prototypes detail most factions attributes but not all. Factions inherit color, a side name, and a side leader from their parent side. In addition, each faction has a specified leader who controls basic policy decisions, fight or quit, for the faction.

Two factions with identical prototypes can be merged into one. Conversely, factions cloned from single parents will initially have identical attributes. However, these data or prototype attributes are controller configurable.

All non-empty factions contain a subset of the units assigned to the parent faction. Upon controller direction, factions will change sides and carry their units along.

5.3.3 Units

Units represent aggregate collections of combat systems and supplies. In JTLS, unit objects are used to represent ground units, air squadrons, support bases/units, ships, air missions, and supply convoys. Each unit inherits the characteristics of its faction (direct) and side (via faction). In addition, each unit has a host of unit characteristics.

In an effort to reduce database preparation requirements, units obtain their objective mix of combat systems and supplies from a unit prototype. Ground units, support units, and squadrons organize and requisition equipment and supplies from a tactical unit prototype (TUP). TUPs function as a table of organization and equipment (TOE or TE). Ships use a ship unit prototype (SUP) to define their ship type. An unlimited number of unit prototypes can be made to define the range of units in the database. And, individual instances of a unit can be modified by the addition of select target systems.

In the SWA demonstration scenario, we employ about 100 unit prototypes to build over 1000 battalion-size units. Our ratio is skewed toward diversity over mass. This diversity ranges from civilian villages and city centers to forward support units, SOF teams, field headquarters, maneuver battalions, assault helicopter battalions, AWACS squadrons, aircraft carriers, mine sweepers, submarines... .

Each incidence of a unit object holds special attributes that define its true status, its perceived status (by all other sides), and its rules of engagement (ROE). Note that a unit's color, or side, is a perceived attribute.

Unit status tracks the exact real number of combat systems and supplies that are issued to each unit and the number that remain after each combat assessment or resupply action is complete. While units start the scenario with a full or partial fill of their prototype equipment, their running status of on-hand systems during a scenario run is not directly dependent on their basic prototype. However, supply stockage objectives and transport capability are established (initially) in the unit prototype and they can influence the status of a unit during the game.

Units have a special dynamic characteristic: its rules of engagement (ROE). Complex ROE are essential for coalition warfare simulations. We aligned ROE options with current doctrine, but have extended the flexibility within the model to support explicit refinements.

ROE can be set for each individual unit or air mission within the game. The doctrinal settings are WEAPONS FREE, WEAPONS TIGHT, and WEAPONS HOLD. These apply in each of three categories: Surface-to-Air, Surface-to-Surface, and Air-to-Air. Surface-to-Surface sets Surface-to-Subsurface rules and Air-to-Air doubles for Air-to-Surface.

We extended these doctrinal settings in three ways. First, we added explicit ranges to the FREE and TIGHT criteria. If non-friendly or hostile units (respectively) close to the specified range, engagement is authorized. Second, we permit a side-specific override that provides more fidelity that the standard KNOWN FRIENDLY and KNOWN HOSTILE discriminators. Thus separate ROE can be established for each of the other sides. Third, an additional NO FIRE setting is available. In this mode, units will not return fire...even in self defense.

Given these extensions, a tailored ROE matrix is supportable. Table 5 captures the results if BLUE has a general air-to-air ROE of FREE for aircraft within 50 kilometers that has been modified for NEUTRALS to HOLD (self defense); a surface-to-surface ROE of TIGHT for units/ships within 3 km (i.e. along a border) modified for WHITE non-combatants to NO FIRE; and a Surface-to-Air ROE of TIGHT within 15 km modified for allied GREEN to NO FIRE.

Table 5: Extended Blue Unit ROE

	WHITE	GREEN	RED	unknown
Air / Air	Self Defense	Self Defense	Engage @ 50 km	Engage @ 50 km
Sur / Sur	No Fire	Self Defense	Engage @ 3 km	Self Defense
Sur / Air	Self Defense	No Fire	Engage @ 15 km	Self Defense

Note: Based on Table 3: Side Relationships

The No Fire mode may never be doctrine and may not be used in actual situations. But, in any simulation with potential automatic engagements based on ROE that, in turn, depend on side relationships and perception of the other unit's side, a NO FIRE ROE is essential.

Multiple perceptions of the color of an object effected by the ROE matrix drove the introduction of rudimentary IFF constructs. A simple IFF is defined by the faction air control prototype with correct identification of air missions being an absorbing state. (Kelleher, 1993)

5.3.4 Targets

Targets represent individual combat systems and key objects. Targets are divided into 20 target classes/types. Each target inherits characteristics of its faction (direct), its side (via faction), and -if applicable- its unit (direct). In addition, each has target-type specific characteristics.

Targets permit mixed resolution simulation. System-level and item-level objects model key capabilities and functions. These can be targeted and destroyed. Target classes include: surface-to-air missiles (SAM) and air defense artillery (ADA), sensors, jammers, surface-to-surface missiles (SSM and TBM), materiel handling equipment (MHE) and command posts... . SAM & ADA provide an air defense capability. Sensors locate targets. Jammers reduce detection of friendly targets and units. SSMs & TBMs attack surface targets with precision or area munitions. MHE enable transshipment of cargo at terminals. Other target types afford parallel capability while placing that capability at risk.

The 20 target types are augmented by unlimited target subclasses or categories. Target categories are defined by their target prototypes. In addition, each specific target is tagged with its own characteristics. This construction extends the flexibility of the entire target system while reducing database preparation requirements.

5.3.5 Conflict Management

JTLS models air battle, sea battle, deep battle, and rear battle at object or mission-level fidelity. Stochastic processes drive air missions, SAM engagements, sensor detections, precision and area weapons engagements, and SSM impacts. Multi-sided conflict management uses the extended ROE and IFF concepts. With exceptions, coordination isn't automated for precision engagements by units on two sides against objects on another.

In contrast, ground combat assessments are recurring scheduled events. JTLS aggregates most ground combat in the close battle. Given compliant ROE, Lanchester attrition will automatically occur between hostile units within range of their organic weapon systems. Forces on different sides contribute and risk proportional combat systems in multi-sided engagements against common hostile forces.

6 COALITION WARFARE SIMULATION

Simulation of coalition warfare requires more that just multi-sided conflict analysis. Integrated implementation of coalition requirements supports a focus on key issues. Given emerging doctrine, the design attempts to enable not restrict employment options. Policy and political decisions drive dynamic force alignment. Information and intelligence differences trigger information exchanges between coalition partners but permit compartmentation if desired. Logistical infrastructure for cross servicing and cross support is visible. Beddown of forces and host nation support is tied to policy decisions. Connections to external command and control systems are underway.

The ability to partition command of forces was built to provide control. In general, order input is restricted by side but coalition operations are enabled. Explicit liaison teams between sides and SOF link-up with other forces permits automated support for combat operations.

Coalition air, ground, and naval operations are possible. BLUE air liaison teams with GREEN forces will permit GREEN ground forces to receive apportioned BLUE close air support. Linked SOF teams will enhance information exchanges with GREEN supported forces and permit BLUE apportioned CAS to support the SOF-linked GREEN unit. Mixed air mission packages allow GREEN fighter CAPs to refuel from BLUE tankers and respond to BLUE or GREEN threat detections. Mixed naval formations permit coordinated ASW operations. All depend on flexible ROE and IFF.

Coalition logistics and host nation support are modeled. Airlift or sealift of non-hostile forces supports coalition force deployment and NEO. Logistical support between factions is permitted but must be explicit. Beddown of forces and joint use of facilities is dependent on the relationships established between the sides. And, port capabilities can be apportioned.

Combat system characteristics are retained by units regardless of their side and multiple sides or factions can use the same type systems. Data construction is eased by permitting any unit in the game to be assigned tactical units prototypes for generic type units. This also eases data classification requirements in a coalition setting.

A data filtering scheme has been adopted to limit data on subordinate nodes within the game to specific side perceptions. Other data security measures await trusted operating systems with multi-level, segmented security features.

A protocol to link to external command and control systems has been developed. Model links to OTH Gold and TADIL systems is undergoing final development and will be used to support an interactive exercise in 1994.

ACKNOWLEDGEMENTS

First and foremost, a special thanks to Ed Kelleher for his contributions to the design, the database, and the paper. In addition, we are indebted to the development team at Rolands & Associates for their ideas and support. Bill, Reenah, David, Beth: without your contributions, this design wouldn't have been implemented.

A Veda team, headed by Igor Axakowsky, built a ten-sided SWA database referenced in the examples and used in functional tests. Additional teams provided support in development and construction of basic coalition warfare concepts and technical skills for their implementation. Major results were provided by Majors Jim Rennie and Jim O'Connor at JWC, Mike Haraway at Veda, Rob Wittman at Mitre, and Dennis Brockway and Paula Stretz at Los Alamos National Laboratory.

We acknowledge the collective contributions of the JTLS Configuration Control Board and the individual input of the members. Without their vision and faith, transition to a coalition warfare model couldn't have been accomplished. Without their review of the design, the end product would have omitted key functions.

Primary financial and logistical support was provided by the Joint Warfare Center. Additional logistical or financial support was provided by JCS J-7, JCS J-8, the United States Special Operations Command, the United States Atlantic Command, USAFE/USAREUR Warrior Preparation Center, the Defense Information Systems Agency, and the 5th Special Forces Group (Airborne).

A final acknowledgement to all agencies and personnel assisting in the functional tests. Special thanks to SAB.

APPENDIX

The Joint Theater Level Simulation is comprised of a system of simulation program modules and support tools. An Executive Program provides a menu-driven interface for model start-up and access to all modules.

The Scenario Preparation Program supports database development. External Database Programs maintain and modify scenario databases and support status checkpoints during model execution.

The Combat Events Program (CEP) tracks all forces within the wargame and adjudicates robust, integrated combat results. The Internode Process (INP) feeds data into the combat events program, to player terminals, and to other INPs to enable the simulation to be reliably distributed over local area or wide area networks.

The Data Receiver Program manages a coherent database of critical objects and attributes during each model run to support modules external to the CEP. Information Management Terminals provide realtime information feedback on the status of units and targets within the wargame while the Model Interface Program permits order input during model execution.

Two alternate graphics programs provide visual display of game data. Both displays now support multiple sides and multiple perceptions. The legacy GraphOver systems are directly tied into shared memory on a simulation node and only support display of units and orbit locations. The new GIAC displays, x-window based, support use of ADRG map backgrounds and display both air and surface tracks in addition to units.

In addition, GIAC provides a parallel data distribution system with flexible filtering schemes.

G-Protocol External Modules (GEMs) are under development to link simulation output to realtime command and control systems and other simulation tools.

REFERENCES

CACI Products Company. 1991 *Simscript II.5 Reference Handbook*. La Jolla, CA. CACI.

Defense Information Systems Agency, Defense System Support Organization and Joint Warfare Center. 1992. *Joint Theater Level Simulation Executive Overview: D-J-00009-H*. Washington D.C. DISA.

Defense Information Systems Agency, Defense System Support Organization and Joint Warfare Center. 1992. *Joint Theater Level Simulation Director Guide: D-J-00010-H*. Washington D.C. DISA.

Kelleher E.P. 1993. Coalition Warfare in the Joint Theater Level Simulation, presentation at 61st MORS Symposium. Dayton, OH.

Roland E.F. and Kelleher E.P. 1993. JTLS ECP Design Form: E-J-93001. Not Published.

AUTHOR BIOGRAPHIES

KEVIN L. BRANDT is a U.S. Army Special Forces Lieutenant Colonel assigned to the Joint Warfare Center as the Project Manager for the Joint Theater Level Simulation. Holds a BA Mathematics from UCLA and a MS in Industrial Engineering (Operations Research) from Kansas State University. Served in command and staff assignments in infantry and armored cavalry units of the 1st Infantry Division (Mech) and 2d Infantry Division. Served in detachment and company commands and as the Group warplans officer in the 5th Special Forces Group. Later served as Deputy Commander, 10th Special Forces Group during Desert Storm and Operation Provide Comfort. Operations Research assignments include duty with U.S. Army Materiel Command (MICOM) in Europe as a research engineer and at HQDA in the Army Initiatives Group, ODCSOPS.

ELLEN F. ROLAND is the co-founder and senior principal engineer at Rolands & Associates Corporation (R&A) where she leads the development team for the Joint Theater Level Simulation. Holds a BS (cum laude) in Mathematics from the University of Wisconsin and a MS in Operations Research from the Naval Postgraduate School. Retired as a Commander from the U.S. Navy after having served at the Joint Strategic Target Planning Staff, Offutt AFB where she developed interactive real-time models for contingency planning. Was technical manager for the Planning Alternatives for Interdicting National Terrorism (PAINT) model. Developed a Fleet Logistics model for evaluation of task force operations and contingency plan analysis at CINCPACFLT. Assigned as a faculty member at the Naval Postgraduate School (Department of Operations Research), where she focused on combat simulations, network theory, and voice recognition technology. At NPS she was a leader on the initial design team for the Joint Theater Level Simulation (JTLS) from 1982 to 1985 and engineered its development and fielding. She led the software design team for development of the item-level Combat Analysis and Sustainment Model (CASMO). In addition to her current work on JTLS, Mrs Roland is supporting the development of an open ocean combat simulation and has been selected to participate in a National Academy of Science study on Voice Recognition Technology.