Linguistic Geometry Tools

LG-PACKAGE

with
Demonstration DVD

STILMAN Advanced Strategies
1623 Blake Street, #200
Denver, CO 80202, USA

boris@stilman-strategies.com
Tel. (303) 717-2110
Fax (208) 279-2574

www.stilman-strategies.com

TABLE OF CONTENTS

A. Introduction to STILMAN and LG

A.1. About This Brochure

This brochure includes a brief description of the LG tools including their capabilities. We discuss scalability of the LG approach and its comparison with other gaming approaches. A chronological description of the LG-related projects is also included. Enclosed to this brochure you will find a Demonstration DVD, which includes a .pdf file of this brochure and several .avi files. These .avi movies are the actual recorded narrated runs of LG-PACKAGE for various scenarios. They are as follows:

- GDK, (17 min) experiments with new game construction employing Game Development Kit,
- LG-EXPERT, (13 min) experiments in training for urban operations (MOUT),
- LG-MOUT, (10 min) proof-of-concept experiments utilizing deception for MOUT,
- LG-ORBITAL, (6 min) experiments demonstrating effectiveness of repositionable satellites,
- LG-PROTECTOR, (15 min) experiments with Integrated Air Defense,
- LG-RAID, (10 min) experiments with presenting LG-based COA to a Blue Commander (MOUT),
- LG-SEAGUARD, (10 min) experiments for optimizing configuration of LCS (Littoral Combat Ship),
- LG-SHIELD, (10 min) experiments with Integrated Ballistic Missile Defense.

To watch movies on your computer, please, install a codec (TSCC.exe file) from the same DVD or from the STILMAN's web site – see READ ME FIRST.txt file on the DVD.

Information about various licensing options can be found in a different brochure "LG-PACKAGE: Price Structure" [19], which can be requested from STILMAN.

A.2. Paradigm Change

Defense Advanced Research Projects Agency - DARPA, Joint Forces Command – JFCOM, Space Missile Center - SMC, Air Force Research Laboratory – AFRL, Naval Surface Warfare Center - NSWC, Army Research Institute – ARI, The Boeing Company, Defence Science and Technology Laboratory - Dstl (Ministry of Defence, UK), BAE SYSTEMS (UK), and others, are taking advantage of STILMAN software systems.

STILMAN Advanced Strategies (STILMAN) is a high technology company based in Denver, CO, specializing in military decision aids, decision-making and Command and Control (C^2) systems. STILMAN's premier technology is based on *Linguistic Geometry* (LG) [48], a new type of game theory *revolutionizing* the paradigms of battle management and mission planning. In essence, LG-based tools automatically generate winning strategies, tactics, and courses of action (COA) and permit the warfighter to take advantage thereof for mission planning and execution. LG looks far into the future – it is "predictive". With unmatched scalability, LG provides a faithful model of an intelligent enemy and a unified conceptual model of joint military operations.

The word "linguistic" refers to the model of strategies formalized as a hierarchy of formal languages. The word "geometry" refers to the geometry of the game board as well as the abstract relations defining the movements and other actions of the game pieces as well as their mutual potential influence on each other. The game board represents the battlefield terrain including land, urban environment, sea, air space, near-Earth space, etc. The abstract relations represent movements of battlespace entities such as ships, tanks, fire teams, aircraft, missiles, etc., and their actions including applications of weapons, sensors, and communications.

When thinking about modern or future military operations, the *game metaphor* comes to mind right away. Indeed, the near-Earth space, the air space, the ground and seas may be viewed as a gigantic three-dimensional game board. The single entities and groups of ground vehicles, manned and unmanned aircraft, missiles, radars, etc. performing a joint task may be viewed as the friendly pieces, whereas the enemy assets may be viewed as the opponent's pieces. The mission commanders and warfighters on various levels have a place in this picture as game players. Presently, various game-based simulators and synthetic environments, with manual (i.e., operator and user-based) decision-making, are employed for training and other purposes. However, without an ability to automatically find the best strategies, tactics, and COA, the games serve mostly to display the current situation, rather than as a basis for automated decision-making with effective adversarial modeling. And that is precisely what LG algorithms do – generate such strategies, tactics, and COA. With LG, the games serve as models from which the solutions could be derived, rather than merely displayed.

The LG-based battlespace model stems from the concept of the LG *hypergame*. A hypergame [58, 59, 60] is a system of several abstract board games (ABG) of various resolutions and time frames (Figure 4, Figure 5). It may include a number of military and non-military concurrent games collectively called the hypergame components. The boards could be either completely separate or sharing common regions. For each local space of concern within the lower resolution games, we can define a mapping ("zoom in") into a higher-resolution game representing the local engagement in a greater detail. Doing this recursively, we create multiple game layers with increasing resolution. Intersecting or separate hypergame components on the same layer and with the same resolution are permitted as well. The games are "hyper-linked", whereby a move in one of the games may (or may not) change the state of the rest of the games included in the hypergame. The number of games in the LG hypergame may vary from several to thousands to represent the most sophisticated extremely large military operations.

A.3. Brief History of LG

Research on a new game theory started in 1972 in Moscow, Russia. For 16 years (since his graduation with M.S. in Mathematics from Moscow State University) Boris Stilman was involved in the advanced research project PIONEER led by a former World Chess Champion Professor Mikhail Botvinnik. The goal of the project was to discover and mathematically formalize methodology utilized by the most advanced chess experts (including Botvinnik himself) in solving chess problems almost without search. The following development showed that the power of the discovered approach goes far beyond original chess problem domain. Subsequent generalization led to a new theory for solving complex search problems from various problem domains. In the 80s, in Moscow, Dr. Stilman developed the foundations of the new approach. Some of these results were included in his Ph.D. Thesis defended in 1984 in Moscow.

In 1990-91, while doing research as Visiting Professor at McGill University, Montreal, Canada, Dr. Stilman coined the term *Linguistic Geometry* (LG) as a name for the new theory for solving abstract board games. LG is a type of game theory, which allows us to solve opposing games of practical scale and complexity. It is applicable to military decision aids, robotic manufacturing, software re-engineering, and traditional entertainment games. Unlike any other known gaming approach, LG provides extraordinarily fast and scalable algorithms finding best strategies for multi-agent discrete games and permit modeling a truly intelligent enemy. LG is applicable to the non-zero-sum games (the so-called asymmetric wargaming) and to the games with incomplete information (with imperfect sensors, weather effects, deception, etc.).

Since 1991, Dr. Stilman has been doing this research as Professor at the University of Colorado at Denver (Section I). In 1995, he has shown that LG is applicable to a wide class of games with concurrently moving agents [48]. Also, in 1997, he has proved that for several classes of games LG generates *optimal* strategies in polynomial time [48]. This groundbreaking result also suggests that for much wider class of games LG strategies are also optimal or close to optimal. The latest version of LG is dispensing with tree search altogether by defining a "projection" of the game tree on the board (by dropping the time axes). If considered in its entirety, this projection essentially forms the graph of the game such that each node in the graph represents multiple nodes of the game tree. However, even if the resultant graph is much smaller than the game tree, it could still be too large for a meaningful search. Within the LG approach, search through this graph is replaced with construction of the small portions of it and only those portions that represent meaningful flow of events, the so-called *trajectories*. Moreover, such "flows" are not constructed in isolation, but are intertwined together as action-reaction-counteraction constructs called LG zones. Essentially, in LG search is replaced by construction of strategies out of several types of constructs, an attack zone, a domination zone, a retreat zone, etc., whose combinations reflect the entire set of winning strategies in abstract board games. In other words, LG allowed us to discover the "genetic code" of abstract board games that provide a complete set of building blocks, "the monoacids", for construction of winning strategies.

A.4. Reviews

STILMAN has amassed considerable evidence, both theoretical and experimental ([48], Sections 0, I), that LG software tools provide highly effective scalable solutions and a faithful model of an intelligent enemy. The approach had been successfully applied to complex military and industrial problems and was recognized nationally. In particular, research on LG Wargaming was listed as one of the 25 most important projects directed against terrorism developed in the US engineering schools [3]. LG systems were successfully demonstrated to U.S. Air Force Scientific Advisory Board and to the U.S. Army Science Board. These boards define US national policy in the defense-related research and its transition to the US Armed Forces. Further recognition was achieved internationally ([20] and Section I).

An inter-departmental group of scientists, engineers and analysts composed of members K, G, and B departments of NSWC (Naval Surface Warfare Center, Dahlgren, VA) evaluated LG as follows: *STILMAN's LG software brings together many elements that are essential to the realization of Network-Centric Battle Management including course of action analysis, automatic allocation of resources, dynamic re-allocation of resources as the operational situation changes, and the coordinated deployment of both manned and unmanned systems. Integration of this software into a weapons control system that also incorporates situational awareness information regarding the deployment of friendly, unfriendly and neutral forces in the operational area will revolutionize the visualization of the battlespace and how the engagement is planned and executed. Through the use of the hypergame technology, the relevant operating picture can be presented to users at all levels of the command hierarchy with the scope and level of detail appropriate to their role. Because the software possesses knowledge of the current situation, including the capabilities of the deployed assets, it can quickly determine the most effective use of those assets to counter threats. This rapid course of action analysis will allow the user to quickly respond to the changing situation, and tasking orders can be automatically generated based upon the course of action selected*. (Section I, LG-SEAGUARD).

Defence Science and Technology Laboratory of the Ministry of Defence of UK (Dstl) hosted a

2-day workshop at Farnborough, UK in 2003. At this workshop, Dr. Tim Gardener (Dstl) evaluated LG as follows [20]: *The LG tool developed by Stilman Inc. uses game theoretic techniques to generate intelligent behavior in autonomous agents. This is a very difficult problem and a very important one. The computations required even in very simple games can easily become so large as to become unfeasible. A computer has no capability to distinguish between `sensible' and `stupid' game moves and no capacity to reason its way to such a distinction. Stilman claims that he has an algorithm which, in a large class of games, will detect and avoid unnecessary calculations. The reduction in computational time is dramatic: billions of calculations reduced to tens. This key reduction is then exploited through the rest of the tool. It is very likely that there is some breakthrough here …* The workshop concluded: *… LG … could be expanded to scenario preparation for … campaign models to assist in what is now a heavy, manpower intensive exercise requiring involvement of military experts … A primary attraction and interest in LG is its ability to automatically control multiple combat units in a coordinated fashion. A highly inventive and innovative application of LG is to develop the appropriate interface to enable its integration into a combat simulation thereby providing control of Computer Generated Forces (CGF), particularly for representing the threat.*

Out of the multiplicity of LG projects three projects with DARPA are in a class of their own (Section I). DARPA is the main research agency at the US Dept. of Defense and, certainly, the main defense research agency in the world. It funds development of technologies that may lead to revolutionary improvements, only. This is what DARPA program managers write about LG:

"… This is an intriguing technology; perhaps the breakthrough in applying game theoretic approaches to practical problems."

Dr. Alexander Kott, Program Manager, Information Exploitation Office (IXO), DARPA.

"*… LG is very prominent in all of Alex's presentations - mine too.*"

Dr. Robert Tenney, Deputy Director, Information Exploitation Office (IXO), DARPA.

A.5. DARPA RAID Project, 2004 - 2007

The most advanced so far application of LG is LG-RAID, an adversarial reasoning system developed for the large-scale DARPA RAID project (*Real-time Adversarial Intelligence and Decision-making*), 2004-07, http://dtsn.darpa.mil/ixo/programs.asp?id=43#. (See also LG-RAID Phase I, 2005 and Phase II, 2006, Section I of this brochure.) RAID demonstrated such progress in Phases 1 and 2 (2004-06) that its Phase 3 was converted into Transition Phase (2006-07) to the US Army DCGS-A Program of Record with subsequent employment in a battlefield starting from 2008. The team of DARPA contractors involved in the integration, experimentation and development includes Lockheed Martin, SAIC, STILMAN, Alion Science & Technology, NewVectors and subcontractors. LG serves as the "brain" behind the software oracle that predicts the future for human adversarial teams, Blue and Red, in an urban environment. As a part of such prediction, this oracle suggests the best courses of action for the Blue team against the actions of the unassisted Red team (also predicted by LG) in real time. Following these recommendations, the Blue team fights Red employing OTB (OneSAF Testbed Baseline, www.onesaf.org), a popular US Army simulation package. Blue and Red teams are physically separated. Both teams are staffed with retired and active Army, Navy and Special Operations Forces personnel.

The RAID validation experiments are conducted with three command and control cells (teams), an LG-assisted Blue Cell (Commander and LG software), entirely human Staff Blue Cell

(Commander and his Staff of five advisers), and an entirely human Red Cell (Commander and five advisers). The Blue Cells, by turns, are pitted against the Red Cell. The Cells control entities (fire-teams, vehicles) within MOUT (Military Operations in Urban Terrain) environment simulated via OTB employing teams of 5-6 puckers (operators). A model of a 4km×4km area of an actual city is utilized. The Blue Cells control a simulated force equivalent to a US company with about 30 to 35 infantry fire-teams, strykers and helicopters. The Red Cell controls several kinds of insurgents with about 30 teams of various sizes. In April 2005, July 2005 and February 2006, three experiment series of 20-25 simulated fights each have been completed. In comparison to the Staff Blue Cell, the LG-assisted Blue Cell demonstrated superior performance. Moreover, in the February 2006 series, RAID demonstrated super-intelligence by far exceeding human courses of actions.

Despite of the previous successes, DARPA RAID Experiment 4 that took place in July of 2006 is in a class of its own. For the purity of the experiment, the Blue Commander during the LG-assisted runs was obligated to utilize the LG-generated COA in his simulated fight against Red. More precisely, the Blue Commander would follow the LG-generated Blue COA and would observe the LG-generated Red COA as potential threats he has to counter to fulfill his mission. During the Staff runs (without LG), the Blue commander and his team did not see or receive any information regarding the LG-based COA, whereas those COA were available for the White Cell (the Umpires) for the comparison sake. The Red Cell had never had any access to the LG COA generated by the RAID tool. Moreover, the Red Cell has not known who they have been fighting with, an LG-assisted Blue Cell or an entirely human Staff Blue Cell.

Out of the 18 paired simulation runs (2 hours each) conducted in Experiment 4, the LG-assisted Blue Cell outperformed the Staff Blue Cell 14 times (78%). In 5 out of these 14 paired runs, the Staff Blue Cell had lost to the Red Cell, whereas the LG-assisted Blue Cell had won. In many other paired runs out of these 14, the difference in scores between Staff and LG-assisted was also significant although both teams had won over the Red. For all the 18 paired runs, on average, the RAID score exceeded the Staff score by about 10% - one standard deviation. Out of the 4 paired runs where the Staff outperformed RAID, only in one of the pairs the difference in scores was about 10%, for the other 3 pairs the difference was under 3%. Overall, the level of confidence in correctness of the RAID-generated COA was 98%.

Among voluminous statistical data collected by DARPA in the RAID experiments we would emphasize just one type of data collected in the July 2006 Experiment 4. After each simulated fight, DARPA requested the Red Commander to answer the question "With whom have you just fought?" (i.e., with Staff or RAID). In 16 out of 36 cases (44%), the Red Commander was wrong. One could say that RAID successfully passed an informal *Turing Test* (i.e., true Artificial Intelligence or not). It is interesting to notice that even when the Red Commander was guessing correctly, he demonstrated a very high opinion about RAID, albeit indirectly. Indeed, often, when he would correctly guess that he just fought with RAID, his reasoning for thinking that his opponent was RAID was that the opposition executed a particularly good strategy such as "very effective defensive posture", "effective shaping fires followed by careful maneuver to establish mid-field position", etc. Amazingly, the observing psychologist noticed that the Red team, the highly qualified military experts (retired colonels), have got so scared of the RAID power that close to the end of the experiments during simulated fights they stopped talking to each other and used the gesture language instead, being afraid that RAID is listening …

B. LG Software Products: LG-PACKAGE

B.1. Generic LG-PACKAGE

STILMAN's software tools include one or more of the following five components: GDK, GIK, GRT, GST and GNS. STILMAN designates the set of generic components as the generic LG-PACKAGE. Price structure for LG-PACKAGE is presented in [19].

- **Game Development Kit (GDK)** permits creation of battlespaces, missions, and campaigns. With GDK, the analysts may optionally develop *domains* (Air, Ground, Joint Operations, etc.) from which specific *campaigns* and *missions* may be developed with a significant level of automation. The domain development includes modeling military hardware (UAV, manned aircraft, tanks, SAM, ships, etc.) as LG piece-templates and automatic generation of battlespace/theater templates from elevation maps in the form of DTED files. Existing and *future military systems* and CONOPS can be modeled.

- **Game Integration Kit (GIK)** permits integration of LG-PACKAGE into a federation of other tools, such as control theory based tools like hybrid systems and discrete event systems, stochastic modeling tools, knowledge-based tools, external synthetic environments, etc. It works well and augments effects of attrition-based software tools, operations research tools, etc. These tools have to be "plugged-in" into the "sockets" provided by GIK in order to feed LG-PACKAGE with necessary information and, in turn, to receive feedback.

- **Game Resource Tool (GRT)** determines the start state of the game, i.e., resources needed for a side at the start of the game in order to win. It provides an optimal resource allocation for a given player (side) for every gaming template within the domain where the resources for all the other players are already specified. While allocating resources so that the designated side may fulfill its goals with a given overall probability of success, the GRT minimizes the total "opportunity cost" of the resources.

- **Game Solving Tool (GST)** simulates the engagement beginning from the start state selected manually, received from other simulation tools or generated by GRT. The engagement is executed by placing and moving the pieces on the board and by automatically, in real time, making decisions for one or more sides of a conflict. GST generates the best strategies, tactics, and COA for every battlespace within the domain. To provide various levels of automation, GST can be executed in several modes, automatic, interactive, and monitoring.

- **Game Network Services (GNS)** support automatic, massively parallel distributed execution of multiple components of LG-PACKAGE over the network of computers including local high-speed networks, Internet, or combinations of both. GNS support concurrent distributed construction and execution of the large-scale LG hypergames. GNS provide extreme robustness to the LG hypergame, so that various adverse hardware/software events (anywhere in the network) would not interrupt hypergame execution. In the worst case, they may reduce execution speed.

B.2. Customization of LG-PACKAGE

For a specific customer, depending on the customer needs, STILMAN may develop customized versions of LG-PACKAGE and assign it a name. A generic LG-PACKAGE for solving a diverse class of problems carries its original name LG-PACKAGE/customer's name. A problem-oriented LG-PACKAGE usually carries name reflecting its purpose, e.g., LG-SEAGUARD, LG-PROTECTOR, LG-SHIELD, LG-RAIDER, etc. (Section I). The customized versions are developed employing LG-FRAMEWORK (STILMAN's proprietary software) and generic components of LG-PACKAGE. LG-PACKAGE may be customized in various ways, either by removing some of the components, by restricting or extending the functionalities of the components, or both.

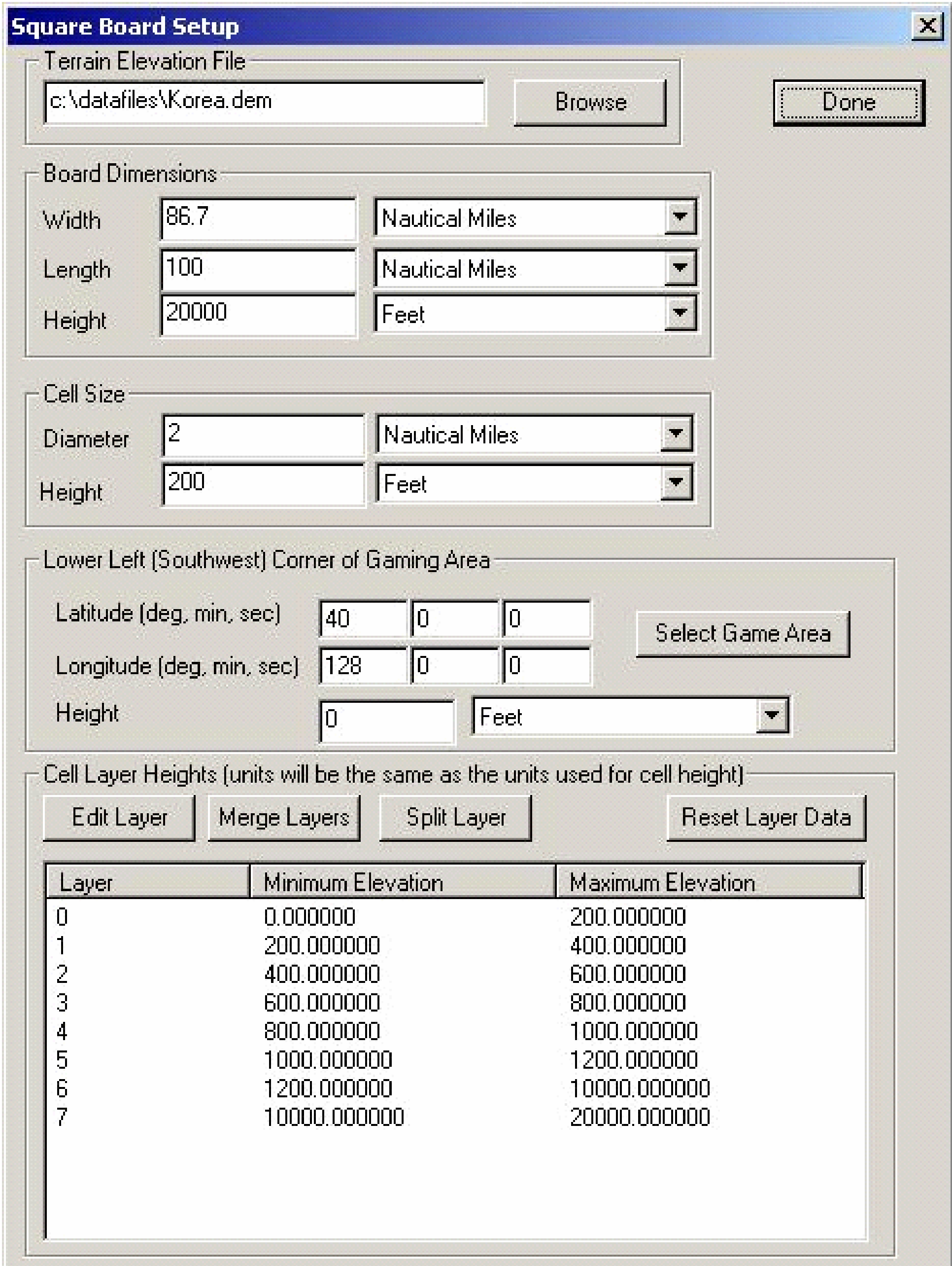

Figure 1. GDK: Construction of abstract board for Integrated Air Defenses

Depending on the type of the license granted to the customer, customization may involve all the components of LG-PACKAGE. For example, we offer customers several problem-oriented versions of GDK, depending on military applications. These versions may differ by the level of abstraction, that is, for strategic, operational, or tactical levels. They may combine some of the functionalities, e.g., strategic-operational or operational-tactical. The kits may also differ by the military scope. For instance, those who specialize in the anti-terrorist operations at any level of abstraction may not be interested in general naval operations, but would require plenty of specific anti-terrorist templates and techniques expressed in game board terms. On the contrary, those who work at CAOC-X (Combined Air Operation Center for Experiments) would want an ability to experiment with various Air Force doctrines and to produce description of various battlespaces amenable to LG solutions. Customized versions of GIK provide extended channels for communication optimized for classes of external simulation packages and other software tools.

The generic GRT and GST work perfectly for various campaigns and missions within the set of pre-developed domains. However, for best results, they may have to be fine-tuned to some of the new domains defined with GDK or imported through GIK. Using a new domain (developed with GDK), STILMAN can build domain-oriented GRT and GST such that for every campaign or mission within the domain GRT can select the best start state, i.e., allocate resources (with measures of effectiveness), while GST can generate the best strategies for all sides of a conflict.

B.3. Game Development Kit (GDK)

The GDK (Section B.1) included in LG-PACKAGE may capture the domains representing subsets of Air Force, Navy, Army, near-Earth Space, or Joint operations. If requested STILMAN may expand this list of domains. However, the power of GDK allows the user to do this expansion him/herself. GDK enables the user to

- capture a domain of battlespaces as a class of ABG and hypergames,
- define battlespace templates within the domain, and
- define specific battlespaces within the domain.

With GDK, prior to developing a campaign, the analysts may optionally develop the domain and/or several sub-domains, such as Joint operations, regional sub-domains (Middle East, Far East, Balkans, Korean Peninsula, etc.), etc. The game board creation (Figure 1) is completely automatic: GDK generates abstract boards from the elevation maps (e.g., *dted* files) and terrain data bases (e.g., CTDB for OneSAF). If the user desires to quickly create and execute a training scenario or plan an actual operation for an area without an existing terrain database, with GDK a faithful mock-up can be constructed employing commercially available satellite images. Domain development includes modeling military hardware (F-16, SAM, cruise missiles, etc.) encapsulated as game pieces, properties of game pieces (motion, weapons, and sensors), rules of engagement, etc. GDK employs most natural graphical point-and-click interface permitting military analysts to model solely based on their intuition, experience and knowledge of the equipment.

Domain construction should be performed prior to the commencement of a campaign, no later than during the campaign planning. It would require an experienced analyst and, depending on the type of the domain and desired level of detail, could take several weeks. Although some knowledge of how LG models the real world entities may be required, no knowledge of how LG solves the problems is necessary.

After the domain is constructed, a warfighter or a mission commander may use GDK to construct a new campaign, mission, or a battlespace. For this task, the operator need not be as experienced as a domain developer. No knowledge of LG, except understanding of the notion of the LG hypergame is required. Of course, understanding of the military objectives and procedures will be needed as well. GDK provides a significant level of automation in helping the operator to create new campaigns, missions, and battlespaces. Within GDK, each campaign, mission, or battlespace is represented as an LG hypergame. Employing input from the users, for each ABG included in the hypergame, GDK generates the abstract board with specific level of granularity based on the space-time scale chosen by the user (Figure 1). GDK allows the users to introduce the mobile and immobile entities, the pieces; various characteristics and capabilities of the entities such as mobility patterns, weapons, and sensors (called "reachabilities" - Figure 2), taking into account their ranges and probabilities of kill; the additional constraints on legal moves like rules of engagement and abort conditions; the winning conditions (based on the campaign goals); etc.

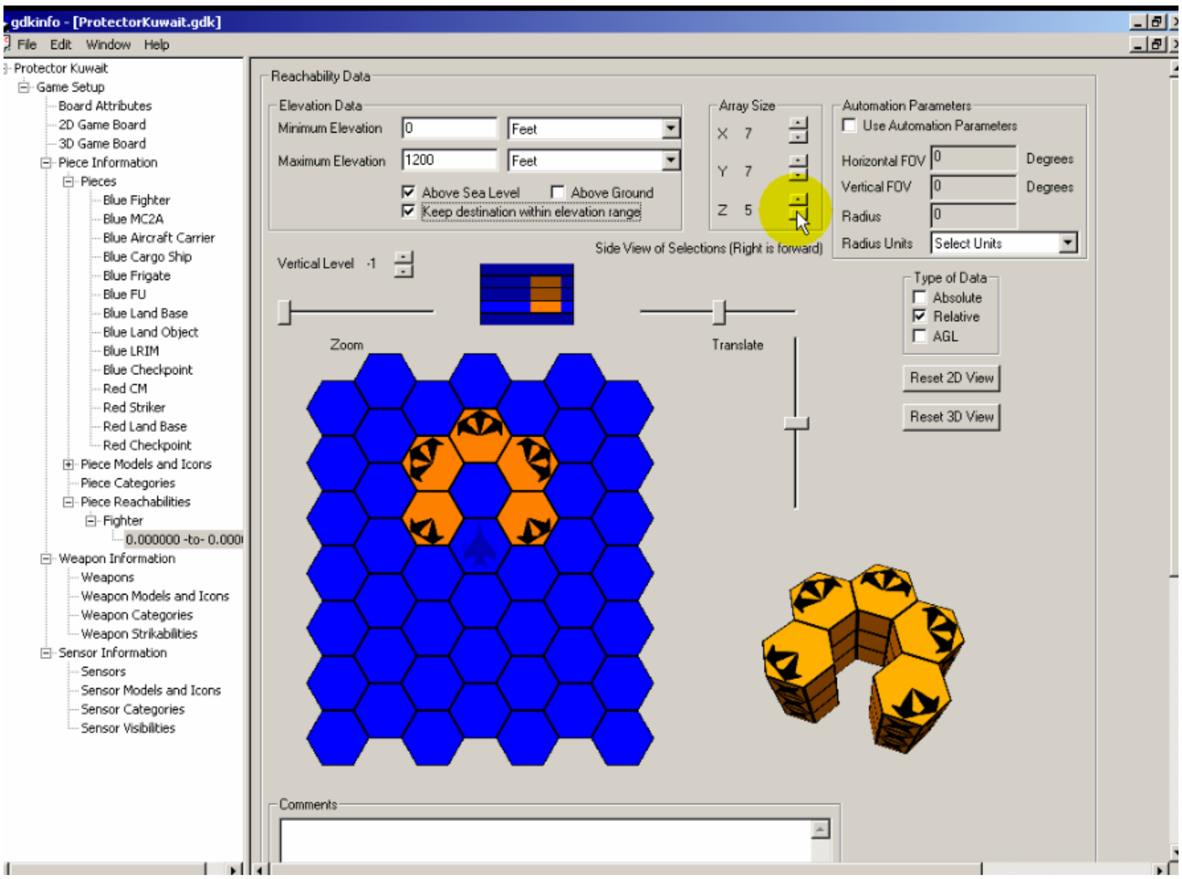

Figure 2. GDK: Defining the reachabilities of the aircraft

Hypergame construction can be accelerated employing the concept of LG templates and dynamic stepwise refinement. GDK stores a library of pre-developed template ABG and complete template hypergames. Library contents and configuration depend of the license requested by the customer [19]. Instead of creating the new campaign from scratch, military analysts may start from the template-hypergame most closely resembling the projected campaign, or they may start by combining several ABG templates resembling components of the projected campaign. More details about the game construction employing GDK are included in the GDK movie.

B.4. Game Integration Kit (GIK)

GIK includes integration modules to allow LG-PACKAGE to integrate with other products in a variety of ways – via exchange of input and output of simulation state data, missions, tactical & strategic calculations. Current interfaces support direct connection (client/server) or message oriented middleware (publish/subscribe) interfaces. Various versions of LG-PACKAGE have already been integrated with other software tools such as Rockwell DES and BBN Omar, Boeing tools (via InfoSphere and TotalDomain), Army OneSAF (via XML Blaster), Overwatch InterSCOPE, Ternion FLAMES, etc. A number of current STILMAN projects include integration with other simulation packages. Usually, integration with other modules through GIK requires tuning to specific customer needs to be performed by STILMAN.

GIK supports integration to third party products through various protocols desired by customer for every node of the "hypergame network" (Section A.1). For each hypergame, multiple LG applications are permitted to run in a networked environment to take advantage of concurrent distributed computing, forming a "hypergame network" (Sections A.1, D). For this network, each node may contain a standalone LG application or a third party application or both. An LG application should be assigned to a set of component ABG (of the hypergame) that are designated as "active" for the node. Within such "hypergame network", the LG applications and third party applications are interconnected using client/server approach. Each node could simultaneously be a "client" and a "server". There is no dedicated "server" version and a dedicated "client" version of LG-PACKAGE. Each LG standalone application in a "hypergame network" can be operated by a different user all taking part simultaneously in the same hypergame. In the future, other methods of the LG-PACKAGE deployment and integration may be possible.

GIK supports the concept of blackboard architecture. This concept requires designating a software module called "blackboard" responsible for communication. LG-PACKAGE as well as external environment communicate with the blackboard via publish/subscribe protocol. GIK supports its own blackboard module but may be adapted to the different types of blackboards. GIK provides conversion of commands, status information, and the initial domain/battlespace data between the internal format of LG-PACKAGE and the blackboard protocol. This protocol, based on XML messages, is a de-facto industry standard.

GIK supports various data interfaces. It can import terrain elevation data from DTED files in DEM and FIL formats (ARC/Info ASCII grid). Graphical overlays can be accepted in BMP, PNG, and JPEG formats. Additional data conversion modules can be added per specific customer needs. LG-PACKAGE can also process CTDB (compact terrain data base) format for elevations and MES (buildings) information. Additional real time data about the current status of simulation or real life operation can be imported via a XML-based blackboard module such as TotalDomain or XML Blaster.

The first preliminary version of GIK was developed for the DARPA JFACC project (Section I) for integration of LG-JEC with DES (Discrete Event System, developed by Rockwell Science Center [18]). This is a type of integration with a lower level entity, which provides information and services at the lower level of abstraction. Also, DARPA JFACC program office requested to integrate LG-JEC with independent Air Combat Simulation Plant developed by BBN. This is the type of integration that allows different technologies work at the same level through a higher-level entity.

One of the modern versions of GIK was employed for integration of LG-PROTECTOR (Boeing, Section I) with InfoSphere, a Boeing proprietary synthetic environment - a customer's blackboard module. This environment utilized LG-PROTECTOR output for the ISR, logistics and engagement modules supported by other software vendors. The subsequent versions of LG-PROTECTOR and comprehensive LG-PACKAGE/Boeing utilized GIK for integration with Total Domain, the next generation of Boeing synthetic environments. The same version of GIK permits accepting information from the IWARS simulation package.

Another version of GIK permits integration of LG-RAID (ARM-S) with the rest of the software packages within the DARPA RAID project (Section I). Employing a shareware blackboard module, XML Blaster, STILMAN's ARM-S, an LG-based Adversarial Reasoning Module, is integrated with ARM-A, responsible for generation of enemy's beliefs, desires & intentions (BDI) and with DRM, a Deception Reasoning Module. In addition, via the same blackboard, all three modules, ARM-S, ARM-A and DRM, are integrated with OneSAF, which simulates real world MOUT (military operations in urban terrain). The online component of GIK provides a reliable high-speed communication channel between LG-RAID (ARM-S) and XML Blaster, which in its turn communicates with OneSAF via double conversion of messages, first, to XML and then to DIS protocol (via SAIC DEM – Data Exchange Module). The offline component of GIK supports conversion of the domain data from the high fidelity terrain database (CTDB employed by OneSAF) into the LG-PACKAGE internal representation compatible with GDK (Section B.3). GIK allows integrating LG-RAID into a comprehensive federated human-computer wargaming system.

Yet another version of GIK was employed within the DARPA Force Multiplier Project (Section I). GIK integrated LG-COMMANDER, a urban warfare oriented LG-PACKAGE, with InterSCOPE, an advanced 2D/3D urban data visualization and sensor data collection environment (developed by Overwatch).

B.5. Game Resource Tool (GRT)

While a number of domains usually enclosed to the full LG-PACKAGE support a wide class of diverse military operations, GRT included in LG-PACKAGE has a more narrow scope. For example, it may be tuned for Land- and Land/Sea-based Integrated Air Defenses and for Ballistic Missile Defense, only. However, if requested, STILMAN may tune GRT to additional domains. The list of domains supported by GRT is being constantly expanded.

GRT determines resources needed for a side at the start of the game in order to win, i.e., GRT recommends how to start the campaign. GRT provides an initial resource allocation for a given side for every gaming template within the domain where the resources for all the *other* sides of a conflict are already specified. While allocating resources so that the designated side may fulfill its goals, GRT evaluates effectiveness of this allocation. Specifically, GRT makes optimization by attempting to achieve or exceed the threshold of probability of success for the side in the operation utilizing allocated resources. Simultaneously, GRT minimizes the total opportunity cost of the resources utilized.

GRT is a highly flexible system. It allows the analyst to conduct what-if analysis of various initial states. Indeed, a threshold for the probability of success chosen by the user may be non-realistic in a sense that it might be unachievable given the available stockpile of resources and the constraints of the terrain. In such case, GRT would still allocate resources, calculate the actual probability of success and the total opportunity cost. Moreover, employing a version of customized GST (Section B.6), the analyst will be able to observe the simulated engagement based on the initial state (resource allocation) just generated. This simulation is based on the best LG strategies for all sides of the conflict generated by GST. Such simulation usually reveals the impact of the imperfect stockpile of resources and chosen terrain. It also reveals the grounds for the predicted low probability of success (below threshold). The analyst can change the resource stockpile, the relative opportunity costs of the weapons and vehicles, or reconsider the place for engagement (if possible) and move it to the area with a better terrain. Given these changes to hypergame, GRT will reallocate resources and generate different initial state for the game, evaluate probability of success and the total opportunity cost of the resources utilized. Then GST could be invoked again to simulate engagement with the new start state. Such experimentation with GRT and GST will lead the analyst to the most thoughtful and well founded recommendation of how to start the campaign.

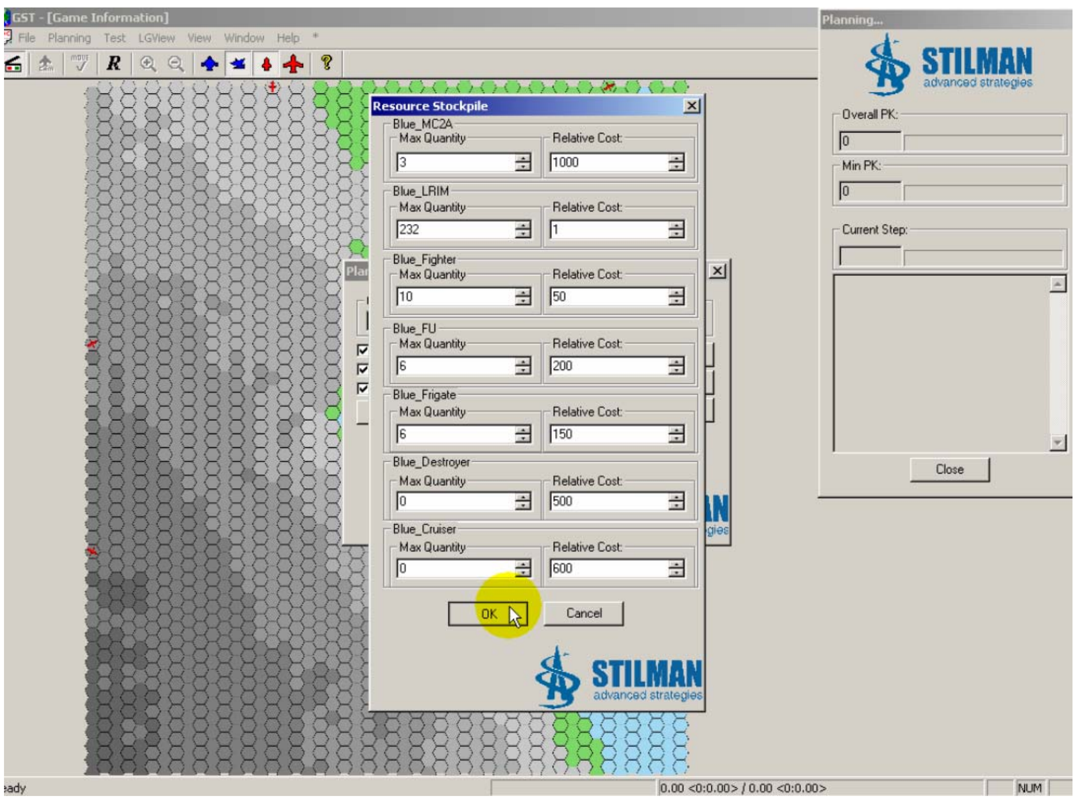

Figure 3. GRT: Stockpile of resources and opportunity costs

More details about the resource allocation employing GRT are included in the LG-PROTECTOR and LG-ORBITAL demonstration movies.

B.6. Game Solving Tool (GST)

Usually, the GST included in LG-PACKAGE is tuned for all the domains enclosed to GDK. Specifically, it may support Land- and Land/Sea-based IAD (Integrated Air Defenses), Littoral operations, SEAD (Suppression of Enemy Air Defenses), TCT (Time Critical Targets), Joint Air/Ground operations, (BMD) Ballistic Missile Defense, CAV (Space/Global Strike - Common Aerial Vehicle) operations, MOUT (Military operations in urban terrain), etc. Figure 4 and Figure 5 depict GST screenshots with strategies generated for the combatants of a Joint Air/Ground operation. Special tuning to GST may be required for complex joint operations (modeled as

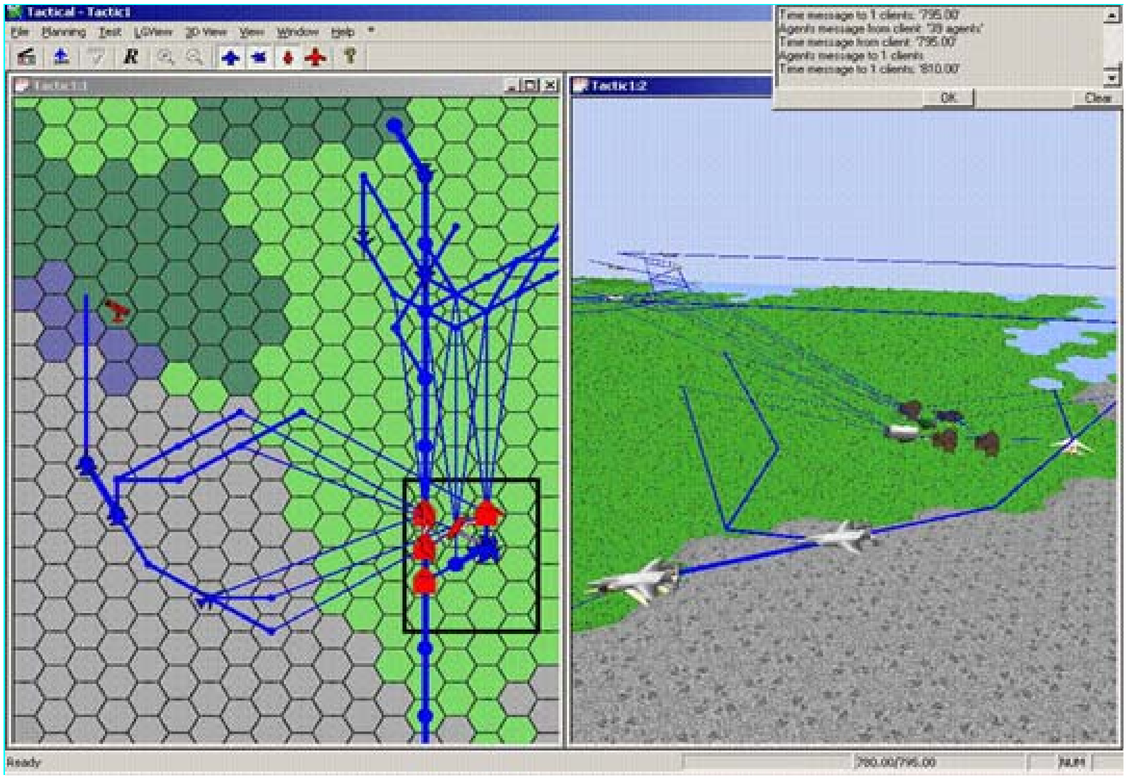

Figure 4. LG-AIR/LAND: the Air ABG of the Joint hypergame (☐ represents the Land ABG, Figure 5).

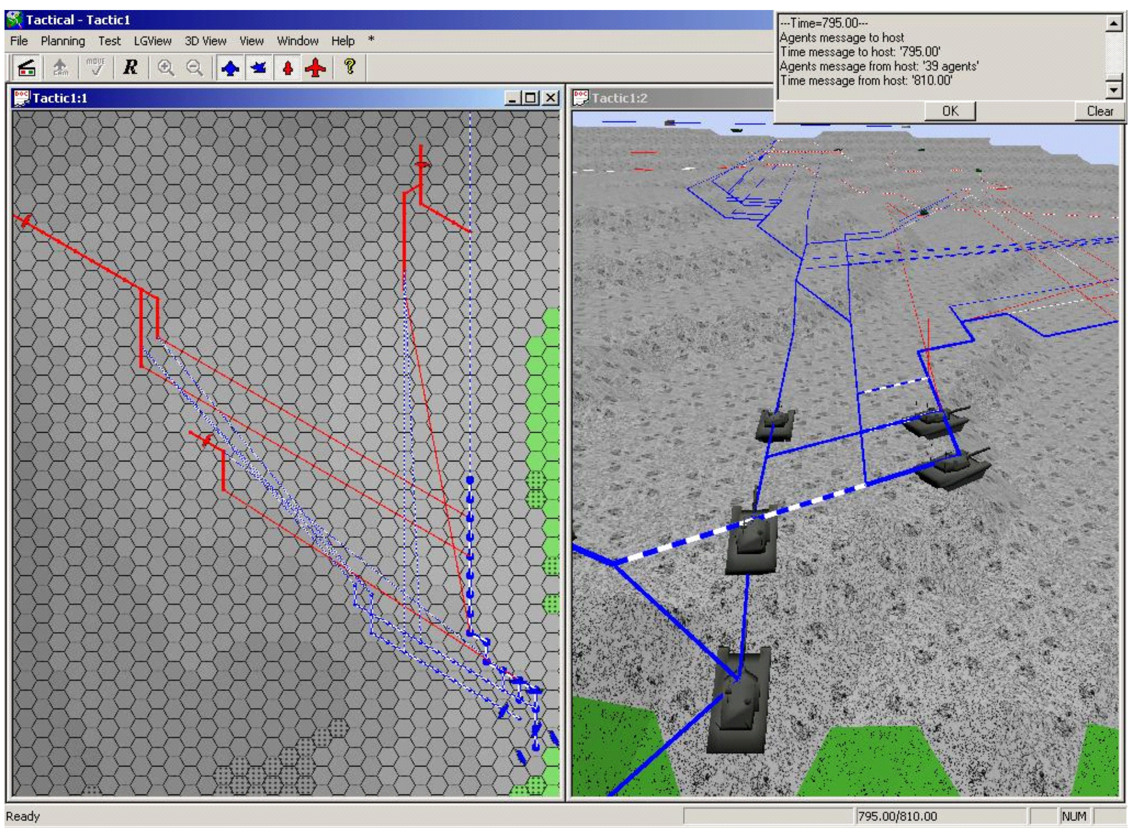

Figure 5. LG-AIR/LAND: the Land ABG of the Joint hypergame (the Air ABG is in Figure 4).

hypergames) that include all or some of the above operations unfolding simultaneously within one campaign. If requested an enhanced GST tuned for additional domains can be built by STILMAN. Moreover, the list of domains supported by the GST can be expanded.

GST simulates the wargame by placing and moving the pieces on the board and by automatically making decisions for one or more sides of a conflict. GST provides a solution to every specific battlespace and/or mission within the domain. Providing a solution means that the GST generates the best strategies and tactics to guide all the sides in the conflict.

GST is the core of LG-PACKAGE. While supporting construction of the LG hypergame (with GDK) and allocation of resources (with GRT), GST itself can serve as an ultimate tool for experimentation and extensive what-if analysis. For instance, experiments with GST may be conducted by varying the game rules, i.e., winning and abort conditions, rules of engagement, etc.

After the start state is selected (either with GDK by manual placing of pieces on the hypergame boards, or automatically with GRT), GST will generate an initial strategy to attain each task within the game. After the actual engagement starts, the mission execution control will be conducted as follows.

In the beginning, the initial LG strategy would be utilized to provide advice for the commander. As the mission progresses, the LG strategy would be updated by taking into account the actual advancement of agents, actual losses/gains, and changes of mobility, as well as the actual enemy actions. Feedback to the operational game from the tactical may cause re-allocation of teams to tasks in the mid-game.

A commander will observe the entire operation (including the logistics part) in the most effective mode as 3D interactive animated movie (running in compressed time) with full explanation of all the actions (provided on request). Visualization of the path planning strategies will provide full awareness and easy interaction between an operator and software. With GST, a commander will become an omnipresent ghost with a virtual "camera." He/she would be able to view the operation by "moving" along the generated path together with all the entities involved. A commander will observe the operation from the cockpit of a fighter flying on a SEAD mission, from the cabin of an amphibious vehicle, through the periscope of an attack submarine, or from a virtual AWACS flying over the entire battlefield. Even a normally invisible event, like damages to adversarial infrastructure or political changes, will be made "visible" together with the chain of events causing this effect. For every team and entity involved in the operation (a strike package, a ship, a submarine, an aircraft, a tank, etc.) and for the whole missions, GST is able to explain its course of actions by visualizing LG zones [48, 59, 60] representing actions, reactions, counter-actions, etc. GST will provide explanation for all the decisions made employing probabilities of kill, integrated probabilities of survival, threshold for retreat, etc.

The great variety of LG capabilities (Section F) is supported by various versions of GST. GST may be executed in several modes, *automatic*, *interactive*, and *watchdog* (Section F). In particular, in automatic mode, GST can control operation of autonomous inhabited and uninhabited vehicles.

More details about generating strategies employing GST are included in the demonstration movies.

B.7. Game Network Services (GNS)

GNS support automatic, massively parallel, distributed execution of multiple copies of various components of LG-PACKAGE over the network of computers including local high-speed networks, Internet, or combinations of both. Employing multiple copies of GDK, GNS support concurrent distributed construction and on-line real-time reconstruction of the large-scale LG hypergames. Employing multiple copies of GRT/GST and GIK, GNS support concurrent distributed execution of the large-scale LG hypergames and their communication with external networks providing, e.g., new intelligence and/or sensor data. GNS provide extreme robustness to the LG hypergame, so that various adverse hardware/software events (anywhere in the network) would not interrupt hypergame execution. In the worst case, they may reduce execution speed.

GNS enable self-organization of the LG hypergame. They keep track of all the games of the LG hypergame that are running on the network and their interconnections. Whenever and wherever an LG-PACKAGE component is started, it reports itself to GNS/Server. When a user or a component itself decides to connect to the active hypergame or send entity data to the hypergame, the component requests from GNS information about this hypergame, in particular, a list of currently running games. Further, employing GNS, multiple components find each other on a TCP/IP network in order to form teams and coalitions with user-defined chain of command.

Instead of many distinct games having separate views of the reality but achieving unity via talking through the network, GNS support *one* powerful LG-PACKAGE distributed over the network and "incarnated" within the multiple computers. The oneness is achieved via the common model of the entire area of operations embodied within the LG hypergame, a hierarchical system of interlinked games. The simplest and, probably, the most robust structure for the hypergame execution would be one-game-on-one-computer, though several games or the entire hypergame - on one computer are also possible. Moreover, this structure is flexible. GNS allow the LG hypergame to re-incarnate games from one computer to another during execution. The global coordination is achieved by the ability of each incarnation to generate advantageous strategies for one game, several games, or the entire hypergame, where a one-game strategy is just a component of the global hypergame strategy. While LG-PACKAGE is one entity, no central processor is required (though, it could be permitted if desired).

One can think of a virtual "octopus" spreading its "tentacles" through the network in separate geographical and functional areas. Only when communications are broken would the tentacles (temporarily) become independent LG-PACKAGEs or LG-PACKAGE components. The games of truncated tentacles are re-incarnated in the main body though with limited "currency". The octopus still keeps its coordinating and cognitive power (perhaps with some small reduction). When the communications are restored, the tentacles rejoin with the octopus, which restores its full power.

C. LG-PACKAGE: Design Guidelines

LG-PACKAGE is not just a problem-solving toolkit. It is a powerful design tool. It allows designing conceptual future battlespaces, missions and campaigns, which may include vehicles, weapons, and CONOPS limited only by imagination of the designer. LG-PACKAGE allows a user to model unplanned (by STILMAN) and even currently unforeseen scenarios by using various combinations of options.

Examples of tested domains include various air offensive missions as well as missions for suppression of enemy air defenses (SEAD). Another well tested domain includes resource allocation and execution of operations for the integrated defense against cruise missiles and enemy strikers. Yet another tested domain includes complex operations that involve various stages of integrated ballistic missile defense.

While the full list of tested domains is much longer and matches well the list of projects STILMAN has been involved in (Section I), it is not a comprehensive list. New domain development may require tuning of LG-PACKAGE by STILMAN software developers. However, it is often desirable for the user to quickly test new ideas, to experiment with proof-of-concept scenarios without STILMAN's involvement. Meeting these requirements, LG-PACKAGE allows rapid design and implementation of unplanned proof-of-concept scenarios without additional software development. Over the years the designing power of LG-PACKAGE was demonstrated on numerous occasions by STILMAN developers and users.

One of the first unplanned scenarios developed with LG-PACKAGE includes 3-game hypergame of tank combat with air support. It was developed within 4 days for demonstrating at DARPA the hypergame concept, which allows several games with pieces with incompatible mobility patterns (aircraft and tanks) to be included in one hypergame (Figure 4, Figure 5). More recent proof-of-concept scenario (developed for BAE SYSTEMS) involves on-the-fly testing of various configurations of the prospective aircraft carrier to optimize its defenses against incoming cruise missiles. It is worth to mention that in absence (at that time) of models of aircraft carrier in GDK it was modeled as a piece of rock (of different configurations) in the middle of the sea. Construction and experiments with this scenario required just half a day. Another unplanned proof-of-concept scenario was developed within 3 days for DARPA. This was the first large-scale LG-controlled military operation in urban terrain (MOUT) involving infantry fire teams.

Though at the time LG-PACKAGE did not have proper means for all the above scenarios and was not tuned for their execution, creative application of LG-PACKAGE allowed rapid implementation and complex wargaming experiments without additional software development.

To support this power in design, LG-PACKAGE has been developed as an *extremely flexible* software toolkit. This flexibility should be exercised with great care. Indeed, LG-PACKAGE includes a large number of user definable options. All possible *combinations* of options are available for the user including those foreseen by STILMAN developers as well as those totally unforeseen. Some of these combinations are well tested while others are untested. Billions of possible combinations of options could be used together. Some of them are useful while others are not plausible. It should be understood that every possible combination of options *could not* be possibly tested by STILMAN developers and some of these combinations may not work well together.

In working with LG-PACKAGE STILMAN recommends the following mode of operation. Initial scenario can be developed by the user even if it is far beyond the tested range. With LG-PACKAGE a user can develop a prototype scenario or even an advanced proof-of-concept scenario as described above. In many cases, a self-made scenario will be sufficient for initial experiments. If required, the fine tuning (including optimization) of the scenario will be made by STILMAN. On all stages of the development users should seek STILMAN's advice and assistance with employing unusual combinations of existing options of LG-PACKAGE. In addition, STILMAN developers can assist users with identifying required new functionality or tuning of LG-PACKAGE.

Our experience shows that often unplanned advanced scenarios may require only small software improvements. For example, this was the case for the scenario for the Air Force time critical targeting (LG-TCT) operations developed for the LG Workshop at Dstl, UK (Section I, [20]). Another example includes scenarios with non-integrated fire control for LG-PROTECTOR (for Boeing, Section I, [20], Figure 10), which originally included only integrated fire control radars.

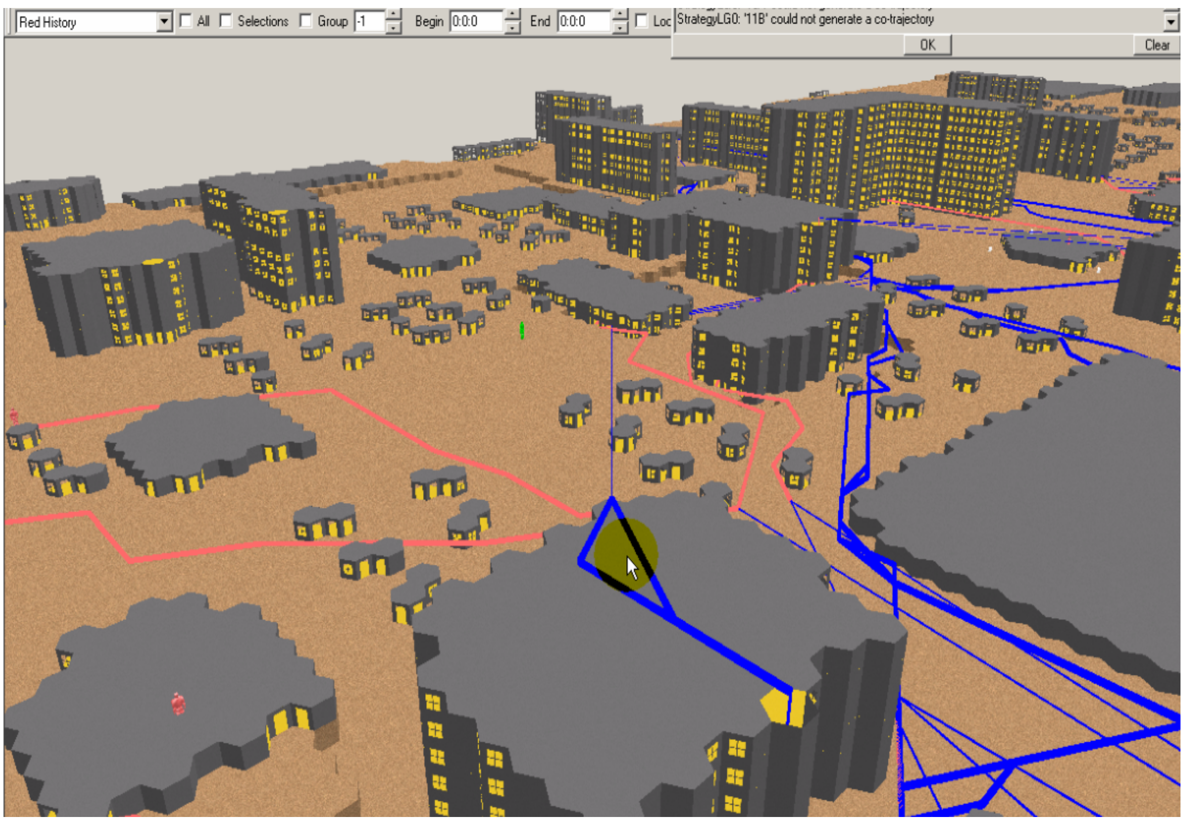

Figure 6. LG-COMMANDER: Military Operations in Urban Terrain

D. Distributed Computing with LG-PACKAGE

Consider a fairly complex military operation, which involves diverse types of units over a large geographical area. The types of forces may range from infantry to aircraft to ballistic missiles and satellites. Modeling infantry may require an abstract board with very small cells, while presence of satellites and ballistic missiles may require the board to include the entire surface of the Earth. However, it is usually computationally prohibitive to cover the entire surface of the planet with small cells, as well as to model movement of objects such as jet aircraft and ballistic missiles on such a board. Furthermore, in most cases this is simply not necessary. For example, high resolution required for infantry is only needed for the sub-region where infantry is present. It is natural then to represent the entire situation as a collection of games. However, these games are not independent and therefore should be solved as one large game.

LG employs the concept of LG hypergames to model complex real-world operations. A hypergame is a system of multiple ABG, which are linked together using hypergame inter-linking mappings. This method allows a complex game to be decomposed into multi-layered games which can be played and solved in an integrated manner. It should be noted that hypergames could also model games for scenarios other than force-on-force engagements such as asymmetric operations, political or economic games. Employing hypergames LG tools generate strategies, tactics, and COA for all the sides of a conflict not only for each game but for the entire system of games, the hypergame.

The first obvious benefit of the LG hypergame approach is that large problems are decomposed into sub-problems. The size of the individual sub-problem is therefore reduced and it can be solved easier while still maintaining connections to the overall problem. The second benefit is the possibility of distributed computing for these sub-problems. Each of the individual games can be executed on a separate processor or computer in parallel. The strategies can still be computed for the entire hypergame as a whole due to strategy exchange and synchronization methods, which are part of the inter-linking mappings. Due to parallel processing of the games, some of them can be introduced specifically for the benefit of distributed processing. The displays of the games used for that purpose only can be suppressed and the results observed on the higher level game.

LG-PACKAGE allows users to define a hierarchical structure of hypergames. Each game has one higher-level parent or host game (except for the top-level host) and any number of lower-level child games. During simulation, individual games are executed as separate processes, which communicate between each other using STILMAN's proprietary protocols built on top of the TCP/IP protocol. This allows different games to *be run on a single processor, on several processors on the same computer, on different computers over a local network or over internet, or any combination thereof.*

The hierarchical structure of hypergames is defined recursively so that there is no special treatment of the top-level or any other games. This allows for a high level of flexibility since the games are only aware of their immediate neighbors (in the hierarchy), while any communication to games further away is possible due to the recursive structure. It is not necessary to execute the entire hypergame tree in each simulation. Subtrees consisting of any cluster of the games can be executed by themselves with the top game of the cluster to become automatically the top-level host.

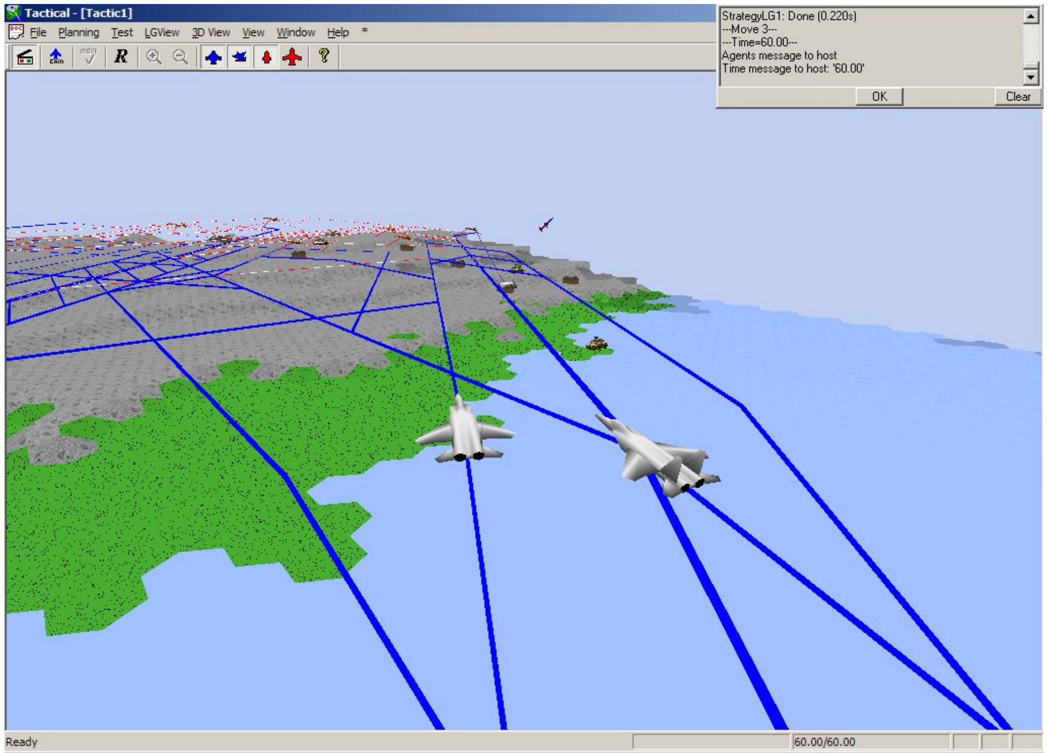

Figure 7. LG-SHIELD: A Local ABG: An air attack on the ballistic missile launches and other targets in N. Korea; a red ballistic missile has been launched. (The Global ABG is in Figure 8.)

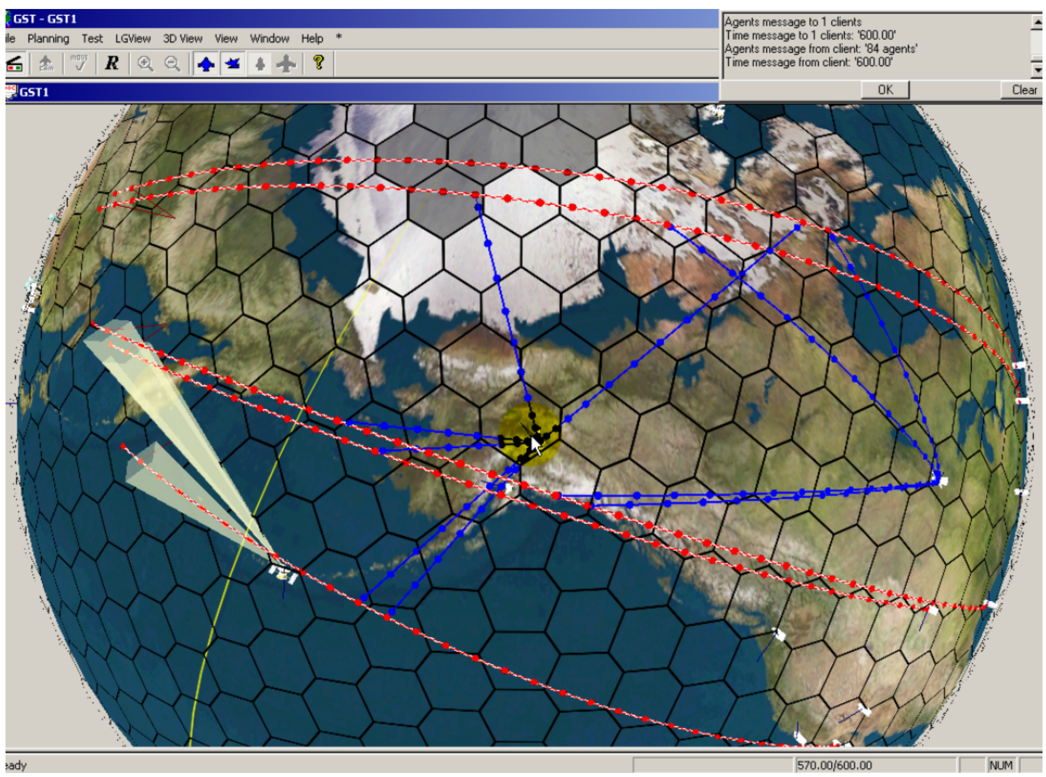

Figure 8. LG-SHIELD: A Global ABG: X-band radar from Shemia Island is tracking red ballistic missiles; blue interceptors from Ft. Greeley, Alaska have been launched. (The Local ABG is in Figure 7.)

To achieve even more flexibility, the protocols allow the games (or entire branches of games) to connect and disconnect from each other at any time of the simulation without violating the overall synchronization scheme. The execution is also not tied to any specific network (or single computer) configuration so as to allow ease of portability. The information about the hypergame structure and interconnections is stored in a GDK data file. Deployment of the simulation on a specific network requires the user to simply provide the IP address of the parent or host game for each process that is started.

Current implementation of a hypergame employs two stage synchronization for strategic and action information exchange.

- During the first stage, all of the games perform LG strategy computations for the pieces within their control. Each individual game's strategy information can be shared across the entire tree of games without requiring other games to know any details about the source game. This allows the strategy of a game to be affected by the events (current and future) of any other game.
- Once the computations for a current time step are finished in all of the games, the top level host begins the second stage of the synchronization. The actions (movements, shootings, explosions, radar illuminations, etc.) from all games are collected and redistributed. All of the actions are then executed across the entire hypergame structure. The master time is then advanced and the two stages repeated for the next time step.

Note also, that the game *time steps do not have to be identical* for all the games. The master time is always advanced to the next time mark needed by at least one of the games; and only those games that need that time mark will perform strategy computations. In addition, the protocols permit out-of-order communications to allow for special events such as connecting or disconnecting games during simulation, human interaction, and others.

STILMAN implemented various hypergames (Figure 4 and Figure 5, Figure 7 and Figure 8, Figure 11 and Figure 12, and Section I). The largest hypergame so far, the 8-game hypergame, was developed in 2003-04 within the scope of the Simulation Based Acquisition LG-CAV project (Section I). There is no theoretical limit to the size of the LG hypergames. The current status of LG-PACKAGE permits implementing hypergames that include hundreds and thousands of games. More details about LG hypergames and their implementation is included in the GDK, LG-PROTECTOR, LG-SHIELD, LG-ORBITAL, and LG-SEAGUARD demonstration movies.

E. Hardware for LG-PACKAGE

LG-PACKAGE, exclusive of the graphics, is executable on any computer system (desktop or laptop) running MS Windows 2000/XP. The minimal requirements are 2 GHz CPU and 1GB RAM and 32MB of video RAM. For more efficient execution we recommend 3 GHz CPU and 2GB RAM with at least 128MB of video RAM. For correct execution of the graphics, the video card must be compatible with OpenGL (NVIDIA preferred). For construction and execution of large-scale hypergames with hundreds and thousands of ABG we recommend a network of PCs, local or global.

F. Capabilities of LG-PACKAGE

- Real time generation of potential COA and strategies
- Modeling intelligent adversaries and their reasoning
- Modeling military campaigns at various levels of resolution

- Situational awareness and predictive analysis
- Managing uncertainty, incomplete information, and deception
- Level 2/3 information fusion

- Resource allocation
- Distributed collaborative planning and execution
- Real time C^2 and decision aids
- Uninhabited vehicles

- Post-mission analysis
- Training and mission rehearsal
- Rapid scenario generation

- Joint Operations
- Effect-Based Operations (EBO)
- Asymmetric Operations
- Military Operations in Urban Terrain (MOUT)
- Network-Centric Operations (NCO)

- Simulation Based Acquisition (SBA)

The first capability of real time generation of potential COA and strategies is discussed throughout the entire brochure. This is the basis for the rest of the capabilities, which are discussed below in Sections F.1 – F.18.

F.1. Modeling intelligent adversaries and their reasoning.

Accurate adversarial reasoning is the key to modeling intelligent adversaries, specifically, friendly COA should be assessed versus enemy COA (eCOA), as an integrated process. The major shortcoming of the present day COA development processes (either manual or automated) is that the COA for the opponents are developed in sequence, i.e., one side attempts to counter a previously developed COA for the opposition. As such, they fail to address the fact that each side's COA is inexorably linked to what the other side is doing, one move at a time [26]. Multiple friendly COA should be assessed against multiple possible eCOA, so that each pair of the COA/eCOA assessment is intertwined into one chain of events that constitutes the interplay between the two combatants. A direct and natural way to adversarial reasoning is to employ the game-theoretical approach (Section 0). Following this approach, one would have to introduce a game that represents a conflict including several opposing and neutral sides. Further, one would have to represent formally the actions of all the sides involved in the game, movements, application of weapons and sensors, communications, goals of each side, etc. Various game-theory approaches can be employed to implement the above representations. Unfortunately, all the conventional gaming approaches, continuous and discrete, fail to provide solutions in real time (Section 0). To make matters worse, usually, the time required for computations exceeds the lifetime of our universe (Sections 0, H). The LG tools simultaneously, in real time, assess Red and Blue behavior by generating LG zones (LG centerpiece action-reaction-counteraction construct [48]) where actions and counteractions of all the sides are taken into account. The LG tools enable the commander to see the “interplay between the two combatants” behind the multitude of the details. LG algorithm implemented in GST and GRT generates COA/eCOA pairs parameterized by probability of success for the Blue side, losses for both sides in terms of the opportunity cost, and other parameters.

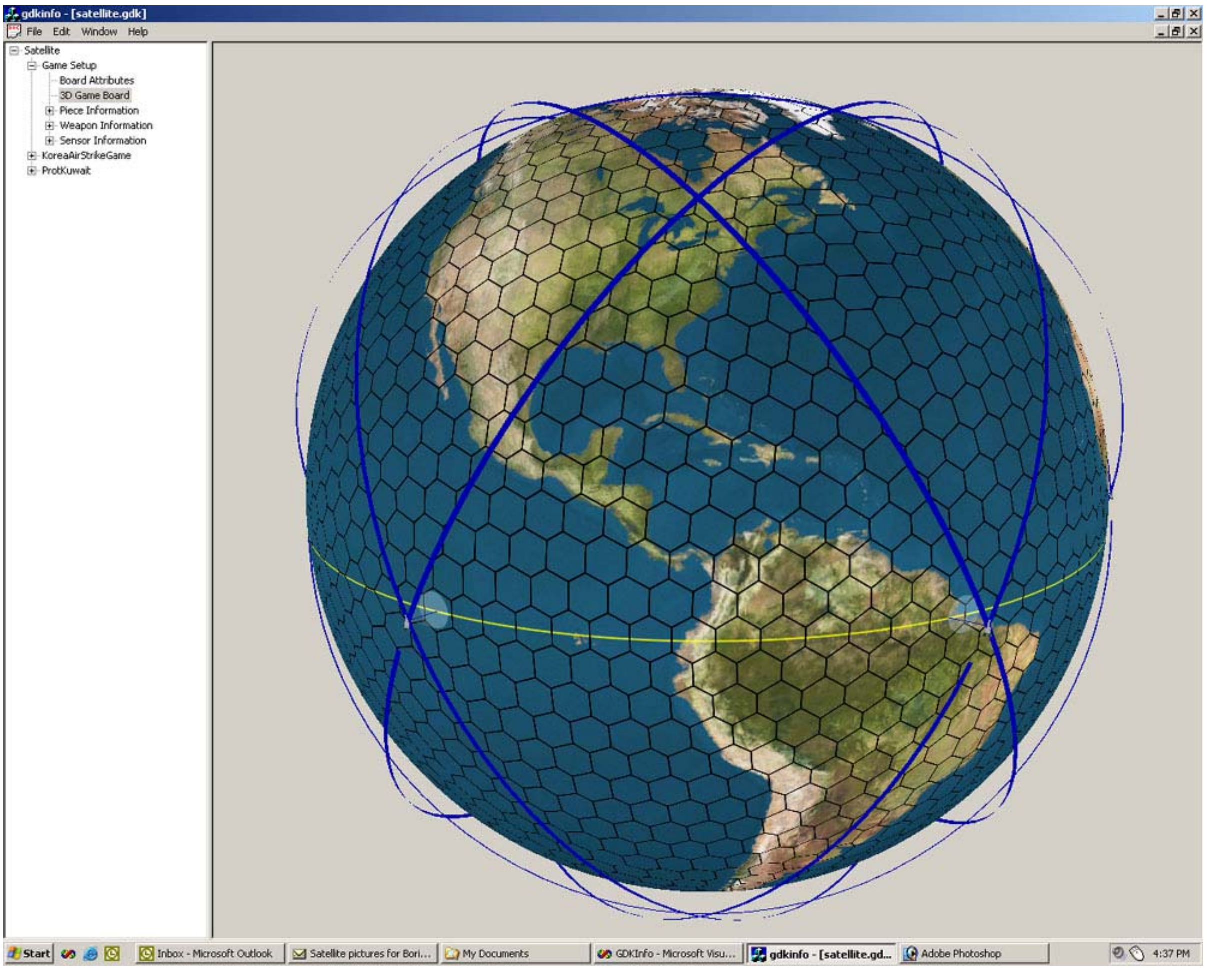

Figure 9. LG-ORBITAL: Constellation of 6 repositionable satellites

F.2. Modeling military campaigns at various levels of resolution

Employing the LG hypergame mechanism LG-PACKAGE permits capture of military operations at all levels, from strategic to operational to tactical. At the top (strategic) level, the lowest resolution models capture the global campaign-size operations, as well as the largest possible groups of military mobile, inhabited and/or uninhabited vehicles. In the LG terms, the abstract board is determined via a low-resolution grid covering the physical domain of the campaign (i.e., oceans, land, air, and near-planet space). The pieces are groups of Air, Land, Sea surface, or Undersea battle units intended to fulfill a uniform goal. The LG motion and weapon reachability relations permit us to encapsulate their mobility and military strength into the ABG. At the lower levels of the hierarchy, high-resolution grids covering relatively small areas are employed. High-resolution ABG capture small groups of vehicles or infantry, as well as individual entities. This capability requires employment of GDK. Examples of the ABG of various resolutions are included in all the movies. In addition, LG-SHIELD, LG-ORBITAL, LG-PROTECTOR and

LG-SEAGUARD movies demonstrate multi-resolution LG hypergames (Section A.1). Construction of multi-resolution hypergames is briefly demonstrated in the GDK movie.

F.3. Situational awareness and predictive analysis

Mission commanders will be able to observe the entire operation in the most effective mode as 3D interactive animated movie (running in compressed time) with full explanation of all the actions. With LG tools, a mission commander will become an omnipresent ghost freely moving within the entire operational theater. He/she would be able to view the operation from the captain's bridge of a cutter, the cockpit of a fighter, from the "virtual cockpit" of an UAV flying on a combat or surveillance mission, or from a "virtual AWACS" flying over the entire operational theater. For every entity involved in the operation and for the whole mission, LG-PACKAGE will explain its COA by providing, if desired, comments for every step including most critical ones like mission abort, engaging the target of opportunity, saving crews of endangered craft, etc. The LG tools fully embody the principle: "know yourself, know your enemy, one hundred battles – one hundred victories" (Sun Tzu). All five tools, GDK, GIK, GRT, GST and GNS provide different facets of the common operation picture (COP).

F.4. Managing uncertainty, incomplete information, and deception

Each LG game piece may possess game sensors representing real-world sensors or even naked eye. A game sensor has a sensing reachability reflecting limitations of real world sensors, limited horizon, line-of-sight detection, viewing angle, etc. Thus the piece does not possess a complete knowledge of the battlespace, which is somewhat alleviated by the abilities of pieces of the same player to exchange information. In conjunction with this, the separate player's worldviews structure of the LG simulation space provides the commander with highly effective means to deal with incomplete information including deception. For instance, the LG tool has means to automatically generate deceptive tactics for the Blue, thus increasing the probability to attain the effects desired by them. It also identifies possible deceptive tactics for the Red, as well as supplies the Blue with counter-measures. For instance, decoys or false attacks undertaken by the Blue may cause the Red to move their forces away from the direction of the actual attack intended by the Blues. Exploration of various uncertainties and deception requires a complete LG-PACKAGE. The LG approach to modeling deception in demonstrated in the LG-MOUT movie (Section A.1).

F.5. Level 2/3 information fusion

LG tools provide Level 2/3 information fusion capabilities and save the warfighter from a devastating flow of massive amounts of data. Instead, the data are automatically converted into logically organized and understandable (through visualization and GUI) segments and layers. This is done by extracting semantically meaningful information. LG-PACKAGE employs LG-based game-theoretical reasoning about objects and events in the battlespace, knowledge of the commander's intent and other relevant contextual information such as environment, doctrine, past behavior, and force capability. Warfighters are given understanding of the past and current battlespaces and the ability to anticipate best options of the battlespace activities in the future. LG tools indicate most desirable targets based on the mission goals and the commander's intent. They recognize enemy activities by generating best strategies and COA options for the enemy including enemy COA most damaging for the Blue side and the COA that the enemy would most likely undertake. LG tools are able to infer relationships of objects in the scene based on their identities or coordinated behaviors and historical analyses. LG tools allow us to detect possible misidentification of the enemy objects by sensors due to either sensor errors or enemy deceptions when the enemy side disguises its entities. They determine situations when human assistance or additional information from sensors or databases is needed. In most of the situations, LG tools automatically resolve ambiguities by analyzing LG-based dynamic semantic model of the situation.

F.6. Resource allocation

The LG tools allocate resources by solving the "inverse strategic problem". While the "forward strategic problem" might be described as "find a strategy for one of the conflicting sides to win the conflict against the adversary", the inverse strategic problem is interpreted by LG as follows. Given the knowledge about the adversary, the desirable threshold probability of success, and the available resources stockpiles for the friendly side, *find* the *best* initial allocation of the friendly resources while minimizing total cost of resources and attaining or exceeding the desired probability of success. The *best* means that for this allocation a winning strategy exists for the friendly side (with probability of success above the threshold). The LG solution of the inverse problem provides the planning capability for mission training and mission preparation in a most natural way. Prior to and/or in parallel with the development of engagement plans, the planners run the LG *resource allocation* games. The winning condition for such games would be achievement of the mission goals with minimal resources. While game construction and experiments with strategies require GDK and GST, respectively, the actual inverse gaming is accomplished by the GRT component of LG-PACKAGE. Scenarios involving resource allocation are demonstrated in the LG-PROTECTOR and LG-ORBITAL movies.

F.7. Distributed collaborative planning and execution

LG-PACKAGE can provide planning, operation monitoring, and dynamic re-planning across geographically separated echelons and across security enclaves. Within *minutes*, employing network of PCs, LG tools will provide commanders and individual combatants with collaborative planning, COA analysis, resource management, and mission execution. LG tools have an ability to share and dynamically update commander's intent and plans, to simulate and assess alternative courses of actions (COA) on the fly, in minutes. All the plans, alternative COA, intentions, resource and assets allocations are presented as 3D interactive animated digital movies, which reflect best warfighters' strategies. LG-based collaborative planning and execution may employ multiple copies of LG-PACKAGE located on multiple computers and handheld devices. This collaborative network could be integrated with database. Multiple copies of LG-PACKAGE will be located in key positions for the Ground operation, on the aircraft and at the Air Operation Center for the Air Force operation, and on the ships for the Naval operation. The top-level strategic computer in the headquarters will plan global strategy and pass on smaller operational tasks to the lower level operational computers in each theater of engagement. Those computers will calculate the strategies for their regions to accomplish specified tasks and pass on their targets to tactical computers for individual battles. In turn, the tactical plan generated on the flagship will be passed on to the individual ships, vehicles, battalions, fire teems. Each higher level accepts feedback from the lower levels on feasibility of tasks that it tries to assign to it, as well as feedback on the actual outcomes. Higher-level computers will plan campaign with lower level of detail, and subsequent levels will refine the details for smaller parts of the problem, break it up again, and pass it to next lower levels. This allows distributing computational complexity between multiple locations using hierarchical scaling. Furthermore, lower levels can request extra resources based on the estimated probabilities of success calculated by LG-

PACKAGE. Higher-level copies of LG-PACKAGE would be able to advise on re-distribution of resources by asking lower-level LG-PACKAGE for an estimate of success if they give up those resources. This decision-making will be based on the LG-PACKAGE capability to provide feasibility and probability of success calculations at the planning stage as well as during mission execution. Collaborative planning and execution can involve multiple copies of partial as well as complete LG-PACKAGE.

F.8. Real time C2 and decision aids

We assume that LG-PACKAGE providing real time decision aids to the mission commander is receiving continuous automatic feed of the current intelligence and sensor data. LG may be utilized in three modes, automatic, advisory, and monitoring. In the *automatic* mode, LG is most suitable for intelligent control of the uninhabited vehicles, either Air, Ground, Sea surface, or Undersea (Section F.9). Various degrees of control are possible, from completely *autonomous* (with a copy of GST on board of the vehicle) to *automatic* (with a human supervisor with an override function), to a *partial control* (with some actuators and sensors controlled by the GST and some by humans). A human commander can derive immense benefits from LG-PACKAGE in an *advisory* mode. This mode is highly interactive. The LG tool displays several Blue COA options parameterized with probabilities of success vs. assumptions about the enemy as well as several possible Red COA most harmful for the Blue together with several COA that the Red would most likely undertake. In addition, the commander can provide the tool with his/her current assumptions about the enemy and may request a "what if" analysis. In case of exercises or training, the LG tool may be switched into the *monitoring* (watchdog) mode. In this mode, it will continuously generate COA while the troops and vehicles are controlled by the unaided operators and commanders going through intense training or exercises. In this mode, the tool would not bother the operator, until it would determine that a disaster will occur unless certain actions are taken. The threat threshold and the intensity of the warnings may be set by the commander. Real time decision aids may be provided on several levels, including *tactical, operational, and strategic*. Autonomous, automatic, partial control and monitoring modes require a GST, while an advisory mode may require a complete LG-PACKAGE. Partial control and monitoring modes have been tested during the DARPA RAID project for MOUT. In multiple experiments, the LG tools demonstrated high efficiency and quality of decisions (generated COA) that often exceeded those suggested by human SME (Sections A.4, I - RAID Phases I, II).

F.9. Uninhabited vehicles

The LG-based Predictive Controller (LG-PC) for uninhabited vehicles will include Rapid Battlespace Constructor, Global Predictive Controller and Local Predictive Controller. These tools will be based on the standard components of LG-PACKAGE. Employing LG-PC for unmanned systems would permit to overturn the existing trend when the number of human operators for one UV is growing making it difficult if not impossible to coordinate swarms of such vehicles. With LG-PC, the robotic vehicles would become truly autonomous. With the multitude of the routine details taken care via LG-PC and aided by its predictive power, the operator would be able to concentrate on the crucial command role – enforcing the high level policy. Thus, the operator would be able to handle many vehicles, instead of the other way around. Moreover, this role could be assumed by the commander of the joint human/robotic forces in the theater or by his/her aids. While controlling actions of the Blue robotic systems and predicting Red actions, LG-PC would provide the common operational picture (COP), including joint operations. This dynamic picture would demonstrate not only the current status of the operation but the dynamically unfolding potential futures. By freeing the commander from tedious control and by providing an ultimate situational awareness, LG-PC would empower his/her operational and strategic thinking. It would give the commander an opportunity to conduct what-if analysis in real time by exploring various unorthodox maneuvers and employing LG-PC for demonstrating their outcome. At any time the commander would be able to impose his/her will by introducing his preferences of COA for the swarms of vehicles, for one vehicle, or even by assuming direct control of a specific vehicle or specific actuators. LG-based situational awareness and prediction of the future would allow for a dramatic extension of the employment of robotic systems – from reconnaissance and point attack missions to global combat missions involving combinations of swarms of UV and human forces, including manned vehicles and dismounted infantry. Human drivers and dismounts would no longer be wary and overcautious of participating in combat operations together with robots because their joint COA and strategies will be transparently displayed on their computers and hand-held devices with safeguards from friendly fire by the UV included. Recognizing a lack of expertise in operations employing swarms of UV separately or jointly with manned vehicles and dismounts, LG-PC would generate a set of training engagements with strategies and COA explained, which would allow the commander and his staff to steadily develop and improve such expertise.

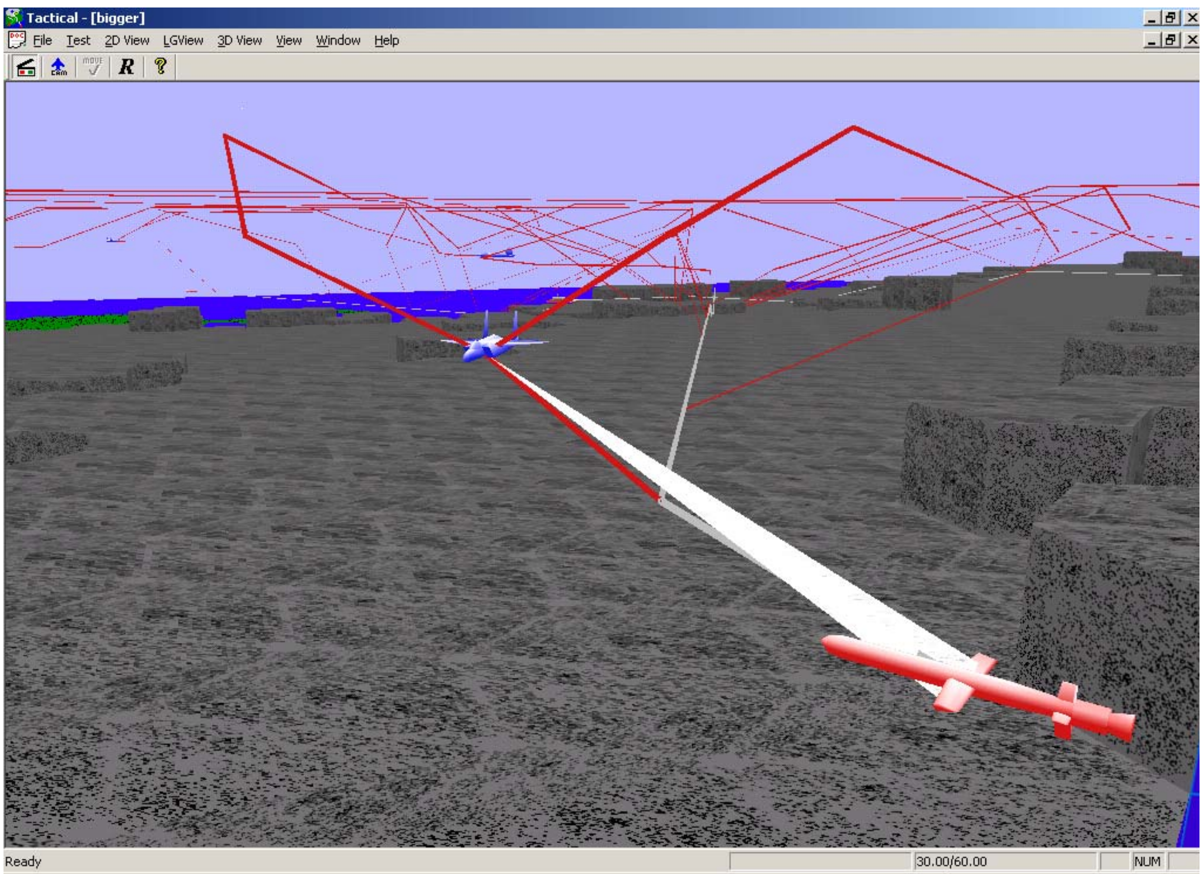

Figure 10. LG-PROTECTOR: Blue aircraft illuminating Red cruise missile.

F.10. Post-mission analysis

After the mission is completed, a commander would be able to replay the mission as a simulation with the final actual information. LG-PACKAGE will analyze mistakes of both sides, reveal their causes, and teach a lesson for the future. LG-PACKAGE will take advantage of strategic patterns developed beforehand by the military experts (either LG-assisted or not) and stored in a database. These retrieved strategies and patterns would allow the analysts to utilize the historical experts' knowledge by identifying strategies leading to familiar patterns of successful operations and by avoiding strategies leading to known failures. The completed mission could be analyzed on the presence of the new patterns and they will be included in the database. Post-mission analysis requires a complete LG-PACKAGE.

F.11. Training and mission rehearsal

A wargaming simulator based on LG-PACKAGE provides highly effective training environment for mission commanders and staff officers by letting them construct and run tactical and strategic scenarios that closely capture real world situations. Training and mission rehearsal with LG-RAID was well tested during DARPA RAID project. The LG-EXPERT instructive movie demonstrates advantages of training and mission rehearsal with LG tools.

F.12. Rapid scenario generation

Employing point-and-click interface, GDK allows a user to create rapidly various battlespaces and wargames, ranging from the urban environment to near-Earth space, from different data sources. GDK also allows users to introduce human teams, platforms, weapons, and sensors. If the user wishes to execute a scenario for an area with an elevation map and other characteristics available via a simulation database, this data can be imported into GDK employing GIK's offline components. For instance, LG-RAID employed a CTDB terrain database for OneSAF (OTB) developed by SAIC for the US Army. This database contains terrain elevation data as well as full buildings information including footprints, doors, windows, and staircases. As GDK imports this information, it is automatically transformed into an LG abstract board of 1.5 million cells, represented as multiple layers of hex prisms, 9m across and 3m tall. GDK's abstract board corresponds directly to the external simulation database. If the user is going to utilize an external simulator (OTB, FLAMES, etc.), this feature allows a scenario constructed employing LG tools to be linked directly to the corresponding scenario being executed externally. If the user desires to quickly create and execute a training scenario or plan an actual operation for an area without an existing terrain database, a reasonably faithful mock-up can be constructed. For instance, to develop such mock-up for the city of Baghdad, first, the user obtains a digital satellite image of the desired region at a half-meter pixel resolution. (Such images are commercially available for almost any area on Earth.) Then, employing GDK, the user creates an overlay of this image that highlights individual buildings. Next, GDK allows him/her to color code this overlay to indicate desired building heights. (If the building heights information is not available for the area, it can be estimated.) Then the overlay is imported into GDK, which automatically creates an LG abstract board – a 3D representation of the city similar to what could have been created from a simulation database. This option allows users to quickly design battlespaces for areas of their choice rather than be restricted to scenarios for areas with existing simulation databases, or to canned scenarios. More details about rapid scenario construction are included in the GDK and LG-EXPERT movies.

F.13. Joint Operations

The difficulty of modeling Joint Operations (and, consequentially, of generating strategies and COA for the Joint Forces) lies in modeling entities with vastly different characteristics within the same framework []. Examples of such extremes are satellites vs. infantry or air vehicles vs. land vehicles or naval units. For instance, if the speed of a soldier in urban terrain may be measured in feet per second whereas the satellite speed is measured in miles per second. The conventional approach to wargaming would result in creation of game cell structure reflecting the common denominator between the speeds resulting in billions of cells covering the Earth size board. This would be computationally untenable. In contrast, the LG hypergame mechanism (Sections A.2, D) permits such entities to coexist within the same framework without creating a huge common denominator game. In several sample scenarios included in the LG-ORBITAL, LG-SHIELD (Section A.1) and other movies, satellites and ballistic missiles coexist with the aircraft, cruise missiles, land vehicles, naval ships, and infantry. There are no limits on the variety of entities and the scale of the operations. Each group of pieces with similar characteristics exists in its own game, a hypergame component, while communicating with the other non-compatible entities via the hypergame links. This permits LG to create advantageous strategies and COA for the Joint Operations with unmatched scalability.

F.14. Effect-Based Operations (EBO)

The EBO approach to planning and execution of military operations is one of the most complex and desirable at the same time. The LG hypergame concept captures all the major elements of EBO, such as effects, causes, direct and indirect effects, and effect indicators. For example, this concept permits to implement the most important aspect of EBO, *effect development*, as follows:

- Attaining desired effects, i.e., given a causal event, generate a behavior of the Blue side that could result in the desired effect despite opposing actions of the Red side (or conclude that it is unattainable).
- Avoiding undesired effects, i.e., given a causal event, generate a behavior of the Red side that could result in the undesired effect despite opposing actions of the Blue side; then add Blue resources or change Blue missions and generate new (not possible before) Blue actions that would allow to avoid this effect.

LG allows us to achieve effect development via *effect inference*. To infer an effect from a causal event, we need to show that after the causal event occurred, the effect will occur at a future time with actions of one of the sides. Employing the LG hypergame concept (Sections A.2, D), we can distinguish between two kinds of inferences, inter- and intra-game. With respect to the inter-game inference, the effect occurs in a different game component than that for the causal event and can be inferred by tracking down through hyperlinks between the games. With respect to the inter-game inference, the effect occurs in the same game component. Specifically, we infer an effect *beta* from a causal event *alpha* with respect to a player Omega if for the initial state satisfying *alpha*, the player Omega has an *LG strategy* achieving *beta*. This approach allows us to represent and assess in real time complex types of EBO with sophisticated effects seemingly unrelated or distant from their causes (and thus untraceable via logical inference or Bayesian nets).

- To attain the desired effect from the cause, LG first builds an inference chain from the cause to the effect with respect to the actions of the Blue side. Essentially, this is a Blue strategy propagated through several games (ABG) to achieve the effect. Then LG would recommend the commander to execute the strategy (to be recomputed at every time interval).
- To avoid an undesired effect, the same inference chain is built, but with respect to the actions of the Red side. Then a Red strategy (in the form of the LG zones [48]) is obtained. After that the algorithm increases the size/efficiency of the Blue forces (via mission reassign and/or

using reserves) to make the LG zones for Red action impassable. This effectively negates the Red strategy to achieve the undesired effects.

A pilot implementation of the effect development was included in the LG-EBO project ([58], 2001, Section I).

F.15. Asymmetric Operations

Asymmetric Operations require modeling at least two sides with (a) vastly different goals; and (b) vastly different means (i.e., force structure, weapons, ROE, etc.) to attain the goals. The most obvious example is the US forces in Iraq or Afghanistan vs. the insurgents, terrorists, and suicide bombers. The LG easily captures both aspects. Whereas most conventional approaches require modeling via zero-sum games essentially limiting the goals of the opposition to directly negating the goals of the Blue side (thereby not permitting substantially different goals for the Red), the LG approach permits independent assignment of missions to the opposing sides, i.e., without them being negations of each other. The other aspect, vastly different means, is handled via the LG hypergame mechanism, as described in Section F.13. In essence, all the scenarios included in the demonstration movies have elements of asymmetric operations (Section A.1).

F.16. Military Operations in Urban Terrain (MOUT)

There are several aspects creating difficulties in modeling MOUT. Joint Forces are usually required to achieve success in MOUT (Section F.13). The sides in an urban conflict are usually asymmetric (Section F.15). The precise modeling of MOUT may lead to combinatorial explosion of the required computations, which makes this type of problems intractable. Indeed, the specifics of the urban environment require modeling urban infrastructure including buildings with their internals, roads, transportation, etc. In LG terms, this requires construction of a sophisticated very high-resolution 3D abstract board (1.5 million cells in RAID!) with the so-called egg-shell cellular structure (Section I, RAID Phases I). In addition, the urban specifics may lead to the difficulties in defining the game pieces with high-resolution reachabilities for motion, weapons, and sensors operating in such environment. While the definition of the MOUT ABG does not pose a problem it may easily lead to the abstract game of such complexity that even LG with its polynomial run time (Section H) would require enormous time to generate strategies. STILMAN successfully overcome all those scalability problems and demonstrated capabilities of LG in MOUT in the DARPA RAID project (Sections A.4 and I, RAID Phases I, II). LG solutions to MOUT are also demonstrated in the LG-MOUT, LG-RAID and LG-EXPERT movies.

F.17. Network-Centric Operations (NCO)

The essence of Network-Centric Operations is providing desperately needed services to the US military forces via a network. The LG hypergame ideology and the LG game-solving capabilities match this approach top-down and bottom-up. LG provides most of the needed services. They are distributed planning, resource allocation, predictive Red/Blue COA generation. The nodes of the NCO network are mission-oriented and the node connectivity follows both C2 hierarchy and communication links. LG models both types of the NCO network via the LG hypergame mechanism. Each of the nodes corresponds to the game, the top hypergame component, assigned to the mission. The command hierarchy is reflected by passing information about the missions back and forth. The downward direction corresponds to generating Blue COA in the mission commander's game (hypergame component) resulting in the set of required actions for the units subordinate to the commander. Each such action is, in fact, a mission assigned to the commander of the subordinate unit within the game of the subordinate unit (which is a subordinate node in the NCO network). Thus, the flow of missions down the C2 hierarchy along the network is defined. The explicit communication links are modeled via the hyperlinks between the games and via explicit communication pieces such as relay towers, radio stations, or power plants supporting communications. The NCO network maintains subsets of itself that may dynamically separate and rejoin due to changing cohesion of the network. The NCO network cohesion is dynamically changing due to appearance/disappearance of the nodes (as governed by dynamically formed missions), communications failures/restorations, and/or radio silences dynamically imposed upon or lifted from missions or battlespace regions. Thus, in the hypergame, the LG hyperlinks not supported via explicit communication links can be dynamically severed or restored following the dynamics of the communication pieces (to be damaged or repaired during the engagement). The COA or other services generated for each mission reflect the information passed through the network nodes along the permitted communication links. The entire network is dynamically following the ever-changing active mission structure captured via LG hypergame. The hypergame mechanism permits the services to be distributed from the commanding generals (at the top) to the squad leaders (at the bottom). The LG supported network permits integration with other technologies providing additional operational and traffic services such as optimization of the information flow or safeguards from information losses or network self-protection.

F.18. Simulation Based Acquisition (SBA)

LG-PACKAGE permits modeling and evaluation of new conceptual military hardware in terms of its functionalities before actually building it [69]. Using LG-PACKAGE, the analysts can create a gaming environment populated with the Blue forces armed with the new conceptual hardware as well as with appropriate existing weapons and equipment. This environment will also contain the intelligent enemy with appropriate weaponry and, if desired, with a conceptual counters to the new Blue weapons. Within such LG gaming environment, the analyst can run various what-ifs with the LG tools providing the simulated combatants with strategies and tactics solving their goals with minimal resources spent. If the new hardware functionality has hidden flaws, the simulated enemy guided by the LG strategies would be able to exploit them providing the hardware evaluators with hands-on proofs of failure. Contrariwise, if the new hardware functionality has spectacular advantages, the Blue forces guided by the LG strategies would be able to convincingly demonstrate how these advantages could be translated into victory for the Blue forces. This not only helps the evaluators to assess the hardware's advantages, it will help to convince the funding agencies, such as US Congress, to fund the prototype construction. In similar fashion, several alternative functionalities could be compared using the ultimate criteria – how well the conceptual weapons and/or equipment will do against an intelligent adversary fully simulated by the LG tools. This is especially important with respect to "constellations" of air and space assets. The constellation concept includes multiple software and hardware elements requiring a significant level of coordination for successful applications. Experimentation within the LG simulated environment may provide an inexpensive alternative to the live exercises designed to catch the bugs in the coordination while facing the intelligent enemy.

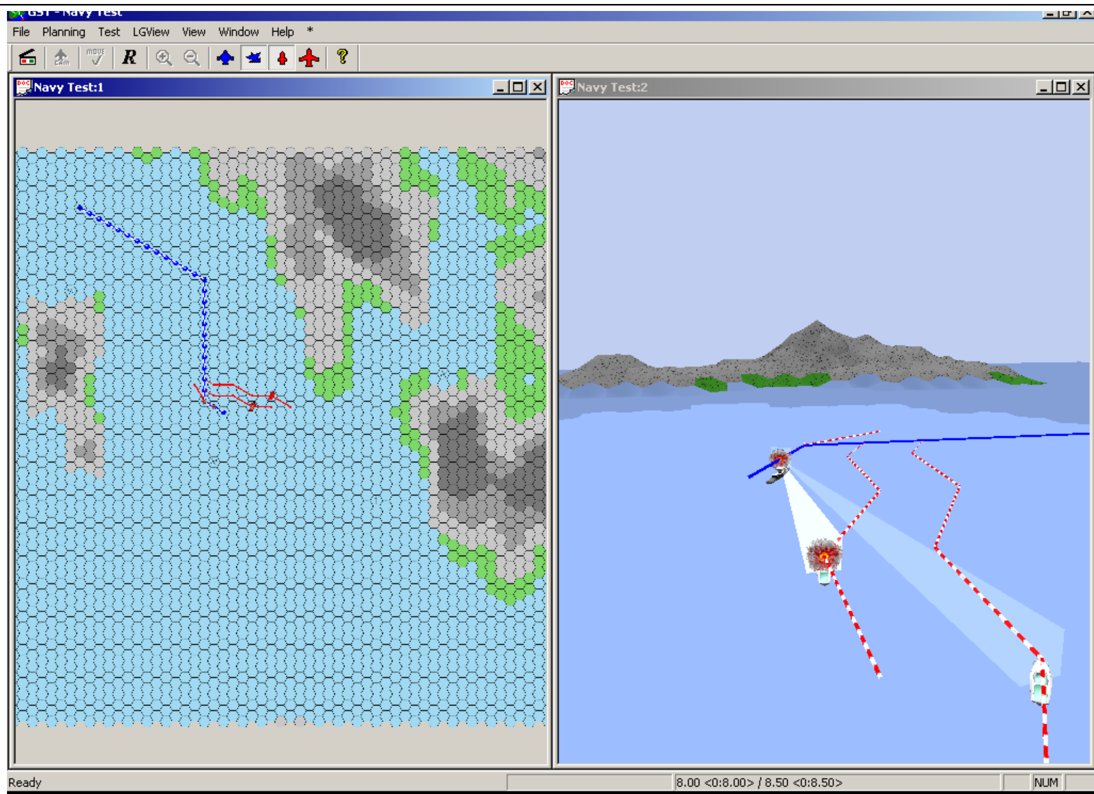

Figure 11. LG-SEAGUARD: LCS defense fails when only one phalanx system is available.

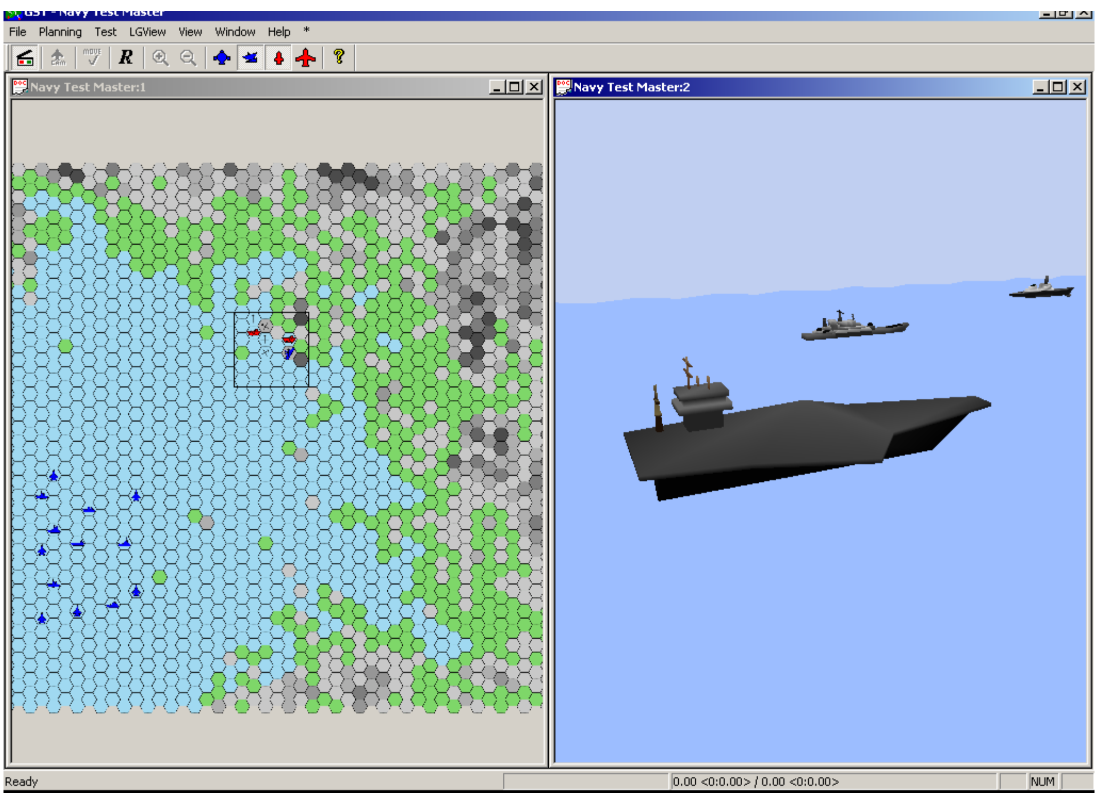

Figure 12. LG-SEAGUARD: LG hypergame capturing an aircraft carrier group.

With respect to the space assets, it is extremely expensive to build any hardware prototype for experimentation. This makes it even more important to evaluate and debug the concepts in a simulated environment before starting to build the hardware prototype. STILMAN has been involved in a number of SBA-related projects including LG-ORBITAL, LG-CAV and LG-SEAGUARD (Section I). More details about the LG-based SBA are included in the LG-ORBITAL and LG-SEAGUARD demonstration movies.

G. LG Approach vs. Other Gaming Approaches

To be successful LG-PACKAGE has to overcome two major technical barriers. The first barrier is related to adequate representation of an active intelligent adversary. The second barrier is the so-called "curse" of dimension or scalability, which often makes all the theoretical constructions impractical. Both barriers have been attacked in the past.

The only theoretical approach that allows introduction of the full-scale intelligent adversary is the gaming approach. Gaming has frequently been applied to military C^2. The games used by many game-based approaches are *continuous and discrete, strategic and extensive*.

- *Continuous games* are usually described mathematically in the form of pursuit-evasion *differential games*. The classic approach based on the conventional theory of differential games [11] is insufficient, especially in case of dynamic, multi-agent models [18, 9]. It is well known that there exist a small number of differential games for which exact analytical solutions are available. There are a few more differential games for which numerical solutions can be computed in a reasonable amount of time, albeit under rather restrictive conditions. However, each of these games must be one-to-one, which is very far from the real world combat scenarios. They are also of the "zero-sum" type which does not allow the enemy to have goals other than diametrically opposing to those of the friend. Other difficulties arise from the requirements of the 3D modeling, limitation of the lifetime of the agents, or simultaneous participation of the heterogeneous agents such as on-surface and aerospace vehicles.
- *Discrete strategic* games were introduced and investigated by Von Neumann and Morgenstern [72] half a century ago and later developed by multiple followers [29]. This approach allows analyzing full game strategies, representing entire games. It does not allow breaking a game into separate moves and comparing them. Only full strategies, the entire courses of behavior of players can be compared. Each player chooses his/her plan of action once and for all and is not informed about of the plan chosen by another player. This significant limitation makes discrete strategic games inadequate for real world C^2 problems.
- *Discrete extensive* games specify the possible orders of events; each player can consider his/her plan of action not only at the beginning of the game but also whenever he/she has to make a decision [29]. Extensive games are usually represented as trees, which include every alternative move of every strategy of every player. Application of this class of games to real world problems requires discretization of the domain, which can be done with various levels of granularity. In addition, in the real world problems, moves of all the pieces (aircraft, tanks) and players (Red and Blue) are concurrent, and this can be represented within extensive (not strategic) games. Thus, the extensive games allow us to adequately represent numerous problem domains including military C^2. However, the classic game theory considers real extensive games (like chess) trivial for "rational" players because an algorithm exists that can be used to "solve" the game [29]. This algorithm defines a pair of strategies, one for each player that leads to an "equilibrium" outcome: a player who follows his/her strategy can be sure that the outcome will be at least as good as the equilibrium no matter what strategy the other player uses. Classic theory of extensive games is not interested in the actual *tractability* of this algorithm, which in practice is not feasible.
- *Practical gaming* approaches try to solve games by searching through the game tree. The main difficulty for any practical gaming approach is scalability, i.e., "the curse of dimension."

Even for a small-scale combat, an extensive game would be represented by a game tree of astronomic size, which makes this game intractable employing conventional approaches.

A number of research groups relied on the *hardware processing power* in solving games and game-related problems. Consider, for example, a small concurrent game with 10 pieces total so that each can make 10 distinct moves at a time. If the game lasts for at least 50 moves (not unusual for battlefield examples), the size of the game tree would be about $(10^{10})^{50} = 10^{500}$ nodes. To be more specific, the JFACC Game (Figure 13 and LG-JEC project, Section I) includes 30 mobile pieces with 18 moves each, while the game lasts 70 moves. No computer can search such trees in a lifetime.

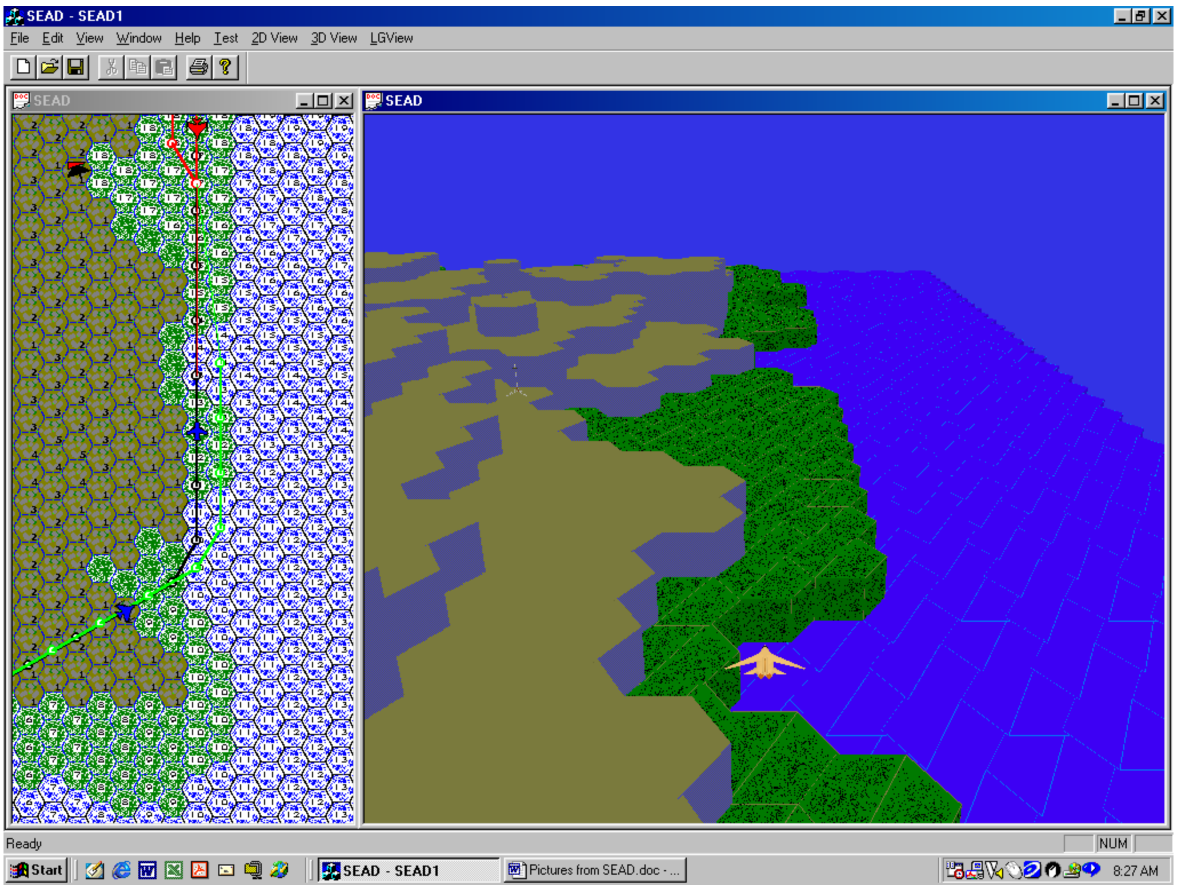

Figure 13. LG-JEC: A SEAD mission around the island

Even the Deep Blue-type hardware-software system cannot make this leap [10, 27, 28]. This massively parallel system of special-purpose chess chips with a processing speed of two hundred million positions per second falls short in an attempt to overcome the exponential growth that comes with a high branching factor. The most presently promising search algorithms on the game trees, those that utilize alpha-beta pruning, would result in insufficient search reduction. Even in the best case, the number of moves to be searched, employing alpha-beta algorithms, grows exponentially with the power of this exponent divided by two with respect to the original game tree [16]. In the above example the reduced tree would have $(10^{500})^{1/2} = 10^{250}$ nodes, which is just as impossible to search as the unreduced tree. Moreover, the alpha-beta pruning method is applicable to sequential alternating games only (Blue-Red-Blue-…) with one-entity-at-a-time movement, whereas most of the real world games, including military operations are concurrent.

For the games with concurrent actions, the number of moves to be searched “explodes” more dramatically than for the sequential games. This is because all the possible combinations of moves for different pieces can be included in one concurrent move. With conventional non-LG approaches, the question of practical solvability of extensive concurrent games could not even be raised. Even future super-computers will not be able to handle this amount of computations employing conventional (non-LG) search procedures.

In contrast, the LG-based models are *scalable* (Section H). With this approach the controlled systems and their environments are modeled as multiagent higher-dimensional ABG with concurrent moves. This methodology allows dramatic search reduction alleviating the huge state spaces characteristic to the problems of dynamic mission planning for military C^2.

H. Scalability of the LG Approach

The major difficulty of employing predictive analysis tools with adversarial reasoning is related to the issue of *scalability*. It means that even *modest* increases in problem complexity, such as adding several tanks, aircraft or platoons, could cause *exponential* increase in computation time to generate plans or make decisions. This is called combinatorial explosion. This is a common problem of all the tools utilizing "look-ahead", that is, an ability to make plans or decisions to achieve certain goals in the future. The problem is considerably aggravated by the fact that the real world military domains are immensely (not modestly) larger than those upon which the majority of the "look-ahead" tools (including those described in other chapters) are being tested.

As a consequence, a number of non-LG wargaming tools that are currently used for planning and control of military operations, do not employ look-ahead but provide only a display of the conflict environment. For such tools, the planning of possible courses of actions is either totally scripted or performed by the human experts.

The LG approach overcomes the combinatorial explosion on two levels.

- The first level is *theoretical*. There is a mathematical proof that LG approach has a low degree polynomial run time [48]. In contrast, for majority of other approaches the complexity is exponential. It means that, unlike LG, the combinatorial explosion is inherent to such approaches and cannot be avoided within the approach itself. As a consequence, many such approaches either employ ad-hoc forward pruning to keep the computations within the required time limits (resulting in generating ad-hoc plans), or employ alternative technologies such as rule-based systems and/or neural networks, which have their own disadvantages. The essence of the contrast between the LG and non-LG approaches to the gaming problems is that LG changes the paradigm *from search to construction*:
    - ➢ The *dynamic hierarchical decomposition* within a hypergame and component games is one of the main principles of LG leading to reduction of dimensionality. For example, with LG large-scale real world problems including those related to warfare are decomposed, via the hypergame approach (Sections A, D), into a hierarchy of smaller, homogeneous abstract board games (ABG) of various resolution and time scales. Moreover, a hypergame with its hyperlinks between the component games permits us to avoid a Cartesian product representation. Such a product could be thought as a gigantic game where every move (also called a multi-move [76]), is a vector including the individual moves from every component ABG. Although convenient mathematically due to simplicity of its definition, such a gigantic product could be an un-scalable obstacle to implementing true concurrency.
    - ➢ The geometrical *relations of reachability* on the abstract board permit to encode the game description in a highly efficient way since most of the game rules (representing movement, application of weapons and sensors) are formalized via relations of reachability (Section B.3). In addition, these relations permit to efficiently generate *trajectories*, as sequences of steps along the optional planning paths;
    - ➢ The geometrical *relations of connectivity* on the abstract board [48] permit to define a hierarchy of constructs used to generate desirable strategies. These constructs include trajectories, zones, and complex zones, where each subsequent construct is defined as a collection of objects of the preceding construct, linked to each other by certain

relation of connectivity. Zones and complex zones represent optional local skirmishes built out of several well organized actions, reactions, counteractions, retreats, etc.

- The dynamic *hierarchy of formal languages* [48] effectively "redefines" the LG game in a way that every game move represents a *translation* from one hierarchy of languages to another. The hierarchy with translations provides an efficient representation of the hierarchy of constructs, which permits to *translate* (i.e., slightly update and reuse) this hierarchy instead of regenerating it from scratch when moving from state to state during strategy generation.
- Essentially, the hierarchy of languages permits to project the "game space-time" (the game tree) onto the space (on the abstract board), *construct a solution* within the board without searching through the "space-time" and elevate the solution back into the "space-time". For many classes of problems including a variety of defense systems, LG constructs are sufficient to solve the game by constructing advantageous strategies *without employing the search tree*. The rest of the problems are usually those that require highly precise solution; the game of chess is one of such problems. For these problems, construction may lead to a *tiny search tree* in the order of a hundred moves [48];

- The second level is experimental. Software implementations could be inefficient, leading to exponential run times despite the theoretical results. Thus the claim of scalability for the LG based software systems must be confirmed experimentally. There have been several lengthy experimental feasibility studies conducted jointly by AFRL, Boeing, Rockwell, and STILMAN in 2000-03 that included hundreds of experiments [61]. These studies concluded that the LG based software tools of mission planning and execution, resource allocation, COA generation and assessment have polynomial run time while several of those tools demonstrated even better, linear, run time growth. Further, multiple experiments with various other LG-based systems, including real world systems (Section I), demonstrated exceptional scalability [20, 61].

I. LG/STILMAN Projects

Though our company is young, STILMAN was founded in 1999, it has already been involved in several large-scale defense-related projects including development of advanced problem-oriented and generic LG-PACKAGEs (Section B). However, the main experience and expertise come from our employees. STILMAN scientists and software engineers are the world leading developers of the LG theory and applications, including the originator of LG Dr. Boris Stilman. Below we listed some of the major events and projects that involved LG/STILMAN.

Year	Project/Event Description
1972–1990	Research on LG started in 1972 in Moscow, USSR. For 16 years Boris Stilman was involved in the advanced research project PIONEER led by a former World Chess Champion Professor Mikhail Botvinnik [1, 48]. The goal of the project was, to discover, formalize, and implement methodologies utilized by the most advanced chess experts (including Botvinnik) in solving chess problems almost without search. Dr. Stilman developed mathematical foundations of the new approach and co-developed software.
1991	The term “Linguistic Geometry” (LG) was coined by Dr. Stilman as a name for the new theory for solving abstract board games. At that time he was a Visiting Professor at McGill University, Montreal, Canada.
	Research on LG continued in the USA, at the University of Colorado at Denver, where Dr. Stilman was accepted as Associate Professor (Professor – since 1994).
1994	LG-based Prototype for Optimal Routing of Emergency Vehicles for the City of Aurora (for Lockheed Martin, GIS Solutions, and University of Colorado Denver).
1995	Demonstration of applicability of LG to a wide class of multi-dimensional, multi-agent games with *concurrently* moving agents.
	Demonstration of applicability to UAV Control (for AFOSR, USAF Phillips Lab at Kirtland AFB, Albuquerque).
1996	LG-based Prototype for Robot Control in Industrial Environment (for CU Denver).
1997	Optimality Proof. For several classes of games LG generates *optimal strategies* in *polynomial time* [43, 48]. This groundbreaking result also suggests that for much wider classes of games LG strategies are also optimal or close to optimal. By that time, 100 papers on LG had been published.
1998	LG-based Prototype for Air Combat Planning with 2D Concurrent Games (University of Denver and University of Colorado at Denver).
1999	*Linguistic Geometry: From Search to Construction*, Dr. Stilman finished manuscript of the first book on LG published by Kluwer in February of 2000 [48].
	STILMAN Advanced Strategies (STILMAN) was founded in Denver, Colorado.
	LG-JEC, for DARPA JFACC project (1999-2001). Contacts: Dr. Alex Kott, DARPA, tel. 571-218-4649, Mike Ownby, Solers, 571-218-4272, Carl DeFranco, AFRL, tel. 315-330-3096. STILMAN, teamed with Rockwell Science Center, University of Colorado, and Wayne State University, has become a major participant of the DARPA JFACC project for the development of the intelligent adviser to the Joint Chiefs of Staff. STILMAN developed LG-JEC (JFACC Experiment Commander), an advanced software prototype for a system supporting SEAD (suppression of enemy air defenses) operations. STILMAN gained substantial experience of integrating its software with external entities including DES (Discrete Event System) by Rockwell and OMAR by BBN. LG-JEC was the very first problem-oriented LG-PACKAGE.

2001	**LG-EBO**, for Boeing, Rockwell, and AFRL (Rome). Dr. Jeff Albert (253-773-9097, jeffrey.h.albert@boeing.com) and Paul Parks (253-773-9042, paul.parks@boeing.com). AFRL SME (subject matter expert): Dr. Maris "Buster" McCrabb (757-508-8735, Buster@DMMVentures.com). Boeing contracted STILMAN to develop an LG foundation for reasoning about Effects Based Operations (EBO) as a part of the Boeing-AFRL (Rome) CRADA agreement. STILMAN developed LG-based formalization of fundamental EBO notions (cascade effects, COGs, effect inference, etc.) based on the LG hypergame theory. STILMAN also developed a demonstration scenario and a preliminary prototype for EBO (LG-EBO) utilizing scenario "Thunder from Space". The approach to EBO based on LG hypergames has been recognized as a highly promising conceptual framework by the leading national experts in EBO including Dr. B. McCrabb (USAF Col., ret.), the Chief Adviser on EBO to AFRL (Rome).
	LG-PROTECTOR for Integrated Air Defenses (IAD). Ten licenses were purchased by Boeing so far with more purchases planned. Contacts: John Hearing (253-657-2135, e-mail: **john.d.hearing@boeing.com**), Paul Bloch (253-773-0376, **paul.bloch@pss.boeing.com**). STILMAN developed LG-PROTECTOR [61], a prototype of decision-making/C2 system providing minimal cost resource allocation, as well as best engagement strategies, tactics, and COA for IAD against cruise missiles and manned/unmanned aircraft. In a Gulf-war-like situation, the Blue Forces established several air bases and supply depots in Saudi Arabia and Kuwait. The Blue stockpiles of resources include MC2A/AWACS, ground radars, airborne interceptor aircraft, and long-range ground-to-air interceptor missiles – all with "opportunity costs" to be varied during experiments. Advanced versions of LG-PROTECTOR include also Blue naval components, such as Aegis ships. LG-PROTECTOR includes a full implementation of advanced fire control by dynamic preemptive control of sensor-to-shooter and shooter-to-target pairing. See LG-PROTECTOR demonstration movie on the DVD (enclosed to this brochure) or download from STILMAN web site [21].
2002	**LG-PROTECTOR (TRL 6)**. Demonstrated to USAF SAB and Gen. Jumper, USAF Chief of Staff. Major advancements to LG-PROTECTOR, up to TRL 6 (Technology Readiness Level), led Boeing to the decision to demonstrate LG-PROTECTOR 1.3 to the Air Force Scientific Advisory Board (USAF SAB) in Mesa, AZ on May 14, 2002 as a part of Boeing's demonstration of Network Centric Warfare management. On Aug. 28, 2002, Boeing demonstrated LG-PROTECTOR 1.4 to Gen. Jumper. Due to success of these demonstrations, Gen. Jumper requested Boeing and STILMAN to develop a proposal for installation of an advanced LG-PROTECTOR 2.0 at CAOC-X (Langley, VA) radio-linked to AWACS within the large-scale project "Transformational Air and Space Expeditionary Force" (TAEF) for USAF. STILMAN is a part of the Boeing TAEF team.
2003	**LG-TCT** for Time Critical Targets; Workshop on LG. After a number of presentations of LG tools in UK and NATO Headquarters (Brussels and The Hague) in 2002-03, MOD (UK) organized two-day *International Workshop on LG*. It was held in London on February 25-26, 2003 [20]. This Workshop was aimed to discuss LG theory and familiarize the British government and major defense contractors with new opportunities in defense that are made possible by the LG applications. During this workshop, multiple experiments with LG-TCT employing various Iraq-Kuwait scenarios were related to destroying *time critical targets* (SCUD missile lunches) and protecting American forces from Iraqi's *cruise missiles*. In addition, LG tools demonstrated real time wargame construction and game solving. A panel of Dstl/MoD scientists, military experts and industry representatives evaluated LG approach as scientific breakthrough [20]. MOD allocated funds for purchase of the LG-PACKAGE license for conducting experiments related to two advanced MOD projects (see LG-PACKAGE/Dstl project in 2004).
	LG-AIR/LAND for Joint Operations. Demonstrated at DARPA. In March 2003 STILMAN was invited by DARPA to demonstrate its new application of LG to Joint Operations, the AIR/LAND hypergame. This hypergame includes two games unfolding concurrently in different space-time resolutions. LG-AIR/LAND generates strategies which involve reasoning about all the sides of the conflict within the entire hypergame. If necessary, one game utilizes resources of

2003 (cont.)	the other game. Indeed, tanks in the LAND game call on the fighters from the AIR game to destroy enemy tanks in the LAND game. LG-AIR/LAND demonstrates, in particular, that fighters respond to this call by destroying enemy tanks while "staying in the AIR game".
	LG-SHIELD for Ballistic Missile Defense (MDA SBIR Phase I), invited for SBIR Phase II. MDA; TPOC: Dr. Larry Altgilbers (256-955-1488, Larry.altgilbers@smdc.army.mil); CPOC: Gladys Erskine (256-955-4102). TPOC: William Strickland (256-955-2746); CPOC: Linda B. Gray (256-955-1897) *Two* consecutive projects on LG, *LG Techniques for Missile Defense* and *Linguistic Geometry Concepts for Advanced Engagement Planning* related to dynamic planning of midcourse defense have been successfully completed in 2003. STILMAN developed specifications for software prototypes and demonstrated their feasibility via experiments with LG-SHIELD, a software prototype of reconfigurable Integrated Ballistic Missile Defense (IBMD). LG-SHIELD allowed us to do experiments on the optimal configuration real time re-configuration of IBMD including sensors, interceptors, launch sites, etc., assuming that IBMD is under attack itself. STILMAN has given three invited presentations of LG approach (with software demonstrations) at MDA, Arlington, VA. See LG-SHIELD demonstration movie on the DVD enclosed to this brochure or download from STILMAN web site [21].
	LG-ORBITAL. Repositionable Satellite Employment for Boeing. Contact: Margaret Ryan (714-896-3014, **margaret.a.ryan@boeing.com**). Employing LG-FRAMEWORK and generic LG-PACKAGE, STILMAN applied LG to *Simulation Based Acquisition* (SBA) within the Boeing/DARPA program ORBITAL EXPRESS. STILMAN demonstrated effectiveness of repositionable satellite employment. Models of satellite constellation within the LG Space domain were developed. On a series of experiments, STILMAN established feasibility of constellation of repositionable satellites by demonstrating improvement of the results of the Joint Air/Ground operations based on the improved ISR provided by the repositionable satellites (in comparison with non-repositionable satellites). See LG-ORBITAL demonstration movie on the DVD enclosed to this brochure or download from STILMAN web site [21].
2004	**LG-CAV**. Measures of Effectiveness (MOE) of LG tools; Feasibility Assessment of the Common Aero Vehicle (CAV) for Boeing. Contacts: Ted Ralston (714-896-3312, **ted.ralston-iii@boeing.com**), Keith McIver (714-317-2203, keith.l.mciver@boeing.com). Employing LG-FRAMEWORK and generic LG-PACKAGE, STILMAN investigated feasibility of applying LG tools for SBA on example of the Boeing/DARPA program FALCON/CAV ("Hypersonic Bomber") for Special Operation Forces (SOF). On a series of experiments, STILMAN demonstrated feasibility of employing CAV for rapid response in case of possible international crisis related to the launch of ballistic missiles from North Korea. LG-CAV includes an 8-game hierarchical hypergame which can be distributed between 8 computers, can be executed on a single computer, or any option in between. The strategies for all the games and for the entire hypergame are computed in parallel on every move; however, they are still interconnected as resources from one game assist the resources from other games in their missions. The details of each component or a hierarchy of components of this engagement can be seen simultaneously on 8 individual 2D/3D displays of each game. The top-level host is the All-Earth game, which models defense against ballistic missiles and flight of CAV (Common Aero Vehicle – a future hypersonic exo-atmospheric bomber). It has 3 child games for modeling movement of Special Forces teams needed to provide illumination of the targets for the CAV. The 5th game models descent of the CAV into the atmosphere and delivery of its payload of cruise missiles, while this game itself has a more detailed child game, the 6th game, for modeling the terrain-following approach of the cruise missiles to their targets through the enemy air defense. The 7th game models long-distance flight of a strike package from an aircraft carrier to the area of interest, with another 8th higher resolution game modeling the air-combat around the target area.
	LG-CHALLENGER for COA Generation and Analysis for US Army (SBIR Phase I), invited for SBIR Phase II. CECOM, RDEC, Myer Center, Fort Monmouth, NJ 07703; TPOC: Edward Dawidowicz (732-427-4122, **Edward.Dawidowicz@us.army.mil).** The purpose was to

2004 (cont.)	develop specs and a demo of the LG-based decision aid system for the Army units. LG-CHALLENGER will provide Common Operating Picture (COP) as well as potential consequences and alternatives to commands expressed in BML (battle management language). This leads to verification of semantics behind BML, elimination of ambiguities in commands and objectives, to elevating GDK to the level close to the natural language.
	LG-SEAGUARD for Human-Centric Combat System Automation for US Navy (SBIR Phase I). Naval Surface Warfare Center (NSWC), Dahlgren, VA 22448-5100; TPOC: John Kimball (Phone: 540-653-0783, email: **KimballJD@NSWC.navy.mil**). Additional contacts: John Canning, Code G07, Tel.: (540) 653-5275, email: CanningJS@nswc.navy.mil; and Carolyn Blakelock, NSWC K63, Tel.: (540) 653-5885, email: **BlakelockCJ@nswc.navy.mil**. The purpose was to develop specs for a prototype LG system for the naval combat units. The system was intended to provide best COA and support predictive situational awareness. Employing LG-SEAGUARD, STILMAN demonstrated pilot experiments of LG-based Simulation Based Acquisition for the Naval project of Littoral Combat Ship (LCS). These experiments demonstrated selection of the optimal configuration of the future LCS with respect to defensive weapons and sensors on board the ship in order to successfully withstand attack by multiple small boats. STILMAN has given two invited day-long presentations of the LG approach (with software demonstrations) at NSWC, Dahlgren, VA. In addition, two day-long demonstrations for program managers from NSWC took place in Denver at STILMAN's offices. See LG-SEAGUARD demonstration movie on the DVD enclosed to this brochure or download from the STILMAN's web site [21].
	LG-PACKAGE/Dstl for "Scenario Preparation for Synthetic Environments" & "Control of Computer Generated Forces in Synthetic Environments" for Dstl MoD, UK (Defense Science and Technology Laboratory of MoD, UK); Contact: Dan Tilley, DATILLEY@mail.dstl.gov.uk. After more than two years of mutual visits, extensive studies and demonstrations [20], Dstl purchased a 1-year license for the experiments with a comprehensive generic LG-PACKAGE/Dstl, which includes three types of LG tools, GDK, GRT and GST. Dstl plans to expand utilization of LG tools.
	LG-PACKAGE/BAE for investigation of capabilities of LG for various projects, for BAE SYSTEMS, UK. Contacts: Peter Collins (peter.v.collins@baesystems.com, +44(0)1252-384573), Pavel Grossmann (pavel.grossmann@baesystems.com, +44(0) 1173-028183**).** BAe SYSTEMS purchased a 1-year license from STILMAN for the experiments with a comprehensive LG-PACKAGE/BAE, which includes three types of LG tools, GDK, GRT and GST. This is the largest so far STILMAN's international project. Within one year BAE evaluated capabilities of LG-PACKAGE for various applications including applications to Systems Engineering, Simulation Based Acquisition and Design.
	LG-PACKAGE/Boeing, Phase I for Network Centric Operations for Boeing. Contacts: Dave Manser (714-762-4978, david.b.manser@boeing.com), Leigh Gustafson (714-762-5368, leigh.l.gustafson@boeing.com**)** The Boeing Company purchased a 1-year license from STILMAN for the experiments with a comprehensive generic LG-PACKAGE/Boeing at Boeing Integration Center (BIC West) in Los Angeles. LG-PACKAGE/Boeing includes the most advanced versions of GDK, GRT and GST linked to the Boeing's Total Domain 2.1 software environment. This project manifested change of the past Boeing's attitude to STILMAN's software when Boeing purchased separate software tools expecting only minor improvements with respect other packages because this kind of performance was usually delivered by other software vendors. Boeing finally realized that LG tools provide not just performance improvement - they lead to the revolutionary paradigm change in military C2. In 2005, after evaluating capabilities of LG-PACKAGE for various projects, Boeing expanded our collaboration into various large-scale projects related to the Network Centric Operations (NCO). LG-PACKAGE/Boeing is a centerpiece of the highly prestigious Boeing NCO demonstrations to US Military Forces. Boeing has purchased of multiple copies of LG-PACKAGE/Boeing for various departments including BIC East in Washington DC.

2005	**LG-RAID**, Phase I, for Adversarial Reasoning and Deception for the **DARPA RAID** Project. This project started in Sept. of 2004. Contacts: Dr. Alex Kott, DARPA, 571-218-4649, Alexander.Kott@darpa.mil; Michael Ownby, Solers, michael.ownby.ctr@darpa.mil, 571-218-4272,. This is the largest ever and the most challenging project for STILMAN. DARPA RAID (Real-time Adversarial Intelligence and Deception reasoning) is a 3-year project where STILMAN serves as one of the 5 prime contractors. The rest include Altarum, Lockheed Martin, Alion (Experimentor) and SAIC (Systems Integrator). STILMAN is applying the most advanced capabilities of LG to real time generation of strategies and tactics for all sides of a conflict. An internal name for LG-RAID is ARM-S (Adversarial Reasoning Module from STILMAN). LG-RAID will assist US Army in predicting adversarial behavior and defeating enemies in military operations in urban terrain (MOUT). See proof-of-concept LG-MOUT, LG-EXPERT and LG-RAID demonstration movies on the DVD enclosed to this brochure [21]. Challenges of RAID required major advancements of LG-FRAMEWORK, STILMAN's proprietary set of tools that are used for all the STILMAN's projects. One of such advancements is an "egg-shell" cellular 3D abstract board for modeling internal structure of buildings. Another major advancement allows LG-based systems to generate very long strategies (long-term plans) lasting for up to 180 moves. Employing these plans (for 15-second moves) LG-RAID makes detailed predictions for 1 hour into the future. Yet another advancement is related to the dynamic evaluation of the current state of the abstract board, which leads to the dynamic terrain analysis. This analysis allows LG-RAID to generate strategies avoiding dangerous areas and attacking enemy in the most vulnerable spots. In Sept 2005, after successful completion of Experiments 1 and 2 in April and July, DARPA RAID passed Phase I Gateway and moved into Phase II (see 2006 projects below).
	LG-COMMANDER for Automated Decision Support for Urban Operations for DARPA. Contacts: Chris Ramming, DARPA, jramming@darpa.mil. In this project STILMAN collaborates with TAG (The Analysis Group) and Overwatch Systems. Employing GIK STILMAN integrated LG tools with InterSCOPE, an advanced smart 2D/3D urban data visualization and decision support environment. LG-COMMANDER will be deployed at the Tactical Command Post/Tactical Operations Center (TAC/TOC).
	LG-TRAINER for Operational Training of the Joint Forces for Joint Warfighting Center (JWFC). Contact: CAPT Ray Rodriguez, USJFCOM, **raymond.rodriguez@jfcom.mil.** In this project STILMAN collaborates with TAG (The Analysis Group) and Overwatch Systems. Employing GIK, STILMAN integrated LG tools with InterSCOPE to address current Joint training simulation gaps and deficiencies. Major advancement was achieved in modeling Joint Opposing Forces (OPFOR) and Joint live-virtual-constructive (LVC) environment. It is expected that integrated tools will be used not only for training but for real-time operational planning and execution.
	LG-ADVERSARY for Modeling Asymmetric Adversaries for US Air Force (SBIR Phase I). Invited for Phase II. Contact: William McQuay, AFRL, 937-904-9214, william.mcquay@wpafb.af.mil. Major goal of this project is integration of LG tools with SEAS, an advanced agent-based simulation system to address current simulation gaps and deficiencies. LG-ADVERSARY will permit the commander/analyst to generate high probability alternative futures and to perform predictive analysis of the adversarial courses of actions.
	LG-EXPERT for Distributed Interactive Training for US Army (STTR Phase I). Contact: Dr. Scott Shadrick, Army Research Institute – Fort Knox, KY, **Scott.Shadrick@knox.army.mil**, 502-624-2613. LG-EXPERT is intended for experiments of applying LG software for training and adaptable, embedded battlefield decision-making training exercises. LG-EXPERT is intended to create the ultimate learning environment for warfighters. See LG-EXPERT movie on the DVD enclosed to this brochure.

2006	**LG-RAID, Phase II** for DARPA RAID project. Phase II started in Sept of 2005 (see Phase I description in 2005, above). Contacts: Dr. Alex Kott, DARPA, 571-218-4649, Alexander.Kott@darpa.mil; Michael Ownby, Solers, michael.ownby.ctr@darpa.mil, 571-218-4272. LG-RAID is a tool for predictive analysis: its job is to predict the upcoming actions of the enemy, and do so not just before, but also during the unfolding battle, in near real time. In addition, LG-RAID generates best COA for the blue team. To stress this emerging capability, the RAID program focuses on a particularly challenging environment: a fluid urban fight against a dismounted insurgent force, reminiscent of events in places like Mogadishu, Najaf, Fallujah, etc. In February 2006, DARPA executed Experiment 3 in Ft. Huachuca, AZ. Two dozens of free-play wargames involved complex urban terrain, Red's rapid movement in the familiar city, concealment, deceptions, ambushes, IEDs, RPGs, heavy machine guns, infiltration and civilian spies. Remarkably, RAID predictions were significantly more accurate than those of a very competent staff. In particular, RAID was more accurate in pinpointing the likely locations of concealed insurgent teams and estimating their future re-positioning. Even more interesting that strategies generated by LG-RAID were sophisticated, sometimes counterintuitive, and by far exceeded those suggested the human staff. These results were supported by major advances in further development of STILMAN's LG-FRAMEWORK. One of such advances is the generalization of several types of behaviors like sensor-weapon pairing, suppressive fire or "bound overwatch", reconnaissance teams, etc. into the new general type of LG zones with paired/prerequisite trajectories. Another advance is related to LG zones with dynamically changing restricted areas that support command and control hierarchical structure (like company-platoon), no-go zones, etc. Yet another new type of LG zones with synchronized intercepting trajectories allowed LG-RAID to accomplish the required intricate level of entity synchronization in the MOUT conditions. RAID project is now approaching the 3d phase of the program focused on practical extensions, integration with fielded systems, and transition to warfighters.
	LG-PACKAGE/Boeing, Phase II is an expansion of the Phase I (2004-05). Boeing is planning to purchase full unlimited-term licenses for several successive versions of the new advanced LG-PACKAGE to be used within Boeing NETWORK COMMANDER. Contact: Dave Manser (714-762-4978, david.b.manser@boeing.com). In particular, the new LG-PACKAGE includes dynamic data exchange between all its distributed components, including multiple copies of GDK, GRT and GST as well as several LG hypergames running concurrently on a large-scale computer network (thousands of nodes). The new LG-PACKAGE supports a server-supervised LG hypergame that is capable of self-organization, which provides extreme robustness. If computing environment shrinks, e.g., computers crash, communications are terminated, etc., LG-PACKAGE automatically shrinks as well by recreating the entire LG hypergame on the available equipment and continues execution, possibly, with reduced speed. If the network is restored, LG-PACKAGE automatically expands and restores performance. More details about the networking capabilities of the new LG-PACKAGE are given in Section B.7. In addition to advanced networking, the new LG-PACKAGE supports realistic communications, sensors and hierarchical C2 organizational structure and enables an independent "Umpire" worldview with access to the entire LG hypergame. The new LG-PACKAGE allows users to create teams, coalitions and introduce various types of collaboration within the LG hypergame.

J. References

1. Botvinnik, M.M., *Computers in Chess: Solving Inexact Search Problems*, Springer-Verlag, 1984.
2. Brewer, G.D., Shubik, M., *The War Game: A Critique of Military Problem Solvin*g, Harvard University Press, 1979.
3. Cohen, Warren, 25 Ways to Fight Terrorism: PLAYING WAR GAMES, *ASEE Prism Magazine*, p.31, Washington, DC, Feb. 2002.
4. DARPA RAID program, 2004-2007, http://dtsn.darpa.mil/ixo/programdetail.asp?progid=57.
5. Dunnigan, J., *The Complete Wargames Handbook*, William Morrow & Co., 1992.
6. DII COE I&RTS: Rev 4.1, Compliance Checklists, B-1 – B41, August 2000.
7. DII COE Data Emporium, (2000) SHADE, http://diides.ncr.disa.mil/shade/index.cfm
8. DII COE CM (2000) Documentation, http://dod-ead.mont.disa.mil/cm/dii_docs.html
9. Garcia-Ortiz, A., Wootton, J., Rodin, E.Y., et al. Application of Semantic Control to a Class of Pursuer-Evader Problems, *Int. J. Computers and Mathematics with Applications*, pp. 97-124, 26(6), 1993.
10. Hsu, F-h., Anantharaman, T.S., Campbell M.S., Nowatzyk, A. (1990). Deep Thought, in *Computers*, *Chess*, *and Cognition*, (55-78), Springer-Verlag, New York.
11. Isaacs, R. *Differential Games*, Wiley, New York, NY, 1965.
12. Joint Tactics, Techniques, and Procedures for Antiterrorism (1998), Joint Pub 3-07.2 at http://www.fas.org/man/dod-101/ops/docs/jp3_07_2.pdf
13. Joint Tactics, Techniques, and Procedures for Peace Operations (1999), Joint Pub 3-07.3 at http://www.fas.org/man/dod-101/ops/docs/jp3_07_3.pdf
14. Jones, R.M. Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P., & Koss, F. V. Automated Intelligent Pilots for Combat Flight Simulation. *AI Magazine*. Spring, 1999.
15. Khoo, N., Chen, D., (1995) *The Evolution Of Intelligent Agent And Game Theory: Towards The Future of Intelligent Automation*, www.dse.doc.ic.ac.uk/~nd/surprise_95/journal/vol4/jjc1/report.html
16. Knuth, D., Moore, R., An Analysis of Alpha-Beta Pruning, *Artificial Intelligence*, 293-326, 6(4), 1975.
17. Kuhn, H. W., Tucker, A. W., eds. *Contributions to the Theory of Games*, Volume II (Annals of Mathematics Studies, 28), Princeton University Press, 1953.
18. Lee, J., Chen, Y-L., Yakhnis, V., Stilman, B., and Lin, F., Discrete Event Control and Hostile Counteraction Strategies, *DARPA JFACC Final Report*, Rockwell Science Center, Feb. 2001.
19. *LG-PACKAGE: Price Structure*, 11 pp., STILMAN, 2006.
20. *Linguistic Geometry Workshop, with STILMAN's Comments*, 17 pp., REPORT, Dstl, Ministry of Defence, Farnborough, UK, Feb. 25-26, 2003.
21. *Linguistic Geometry Tools: LG-PACKAGE*, with Demo DVD, 48 pp., STILMAN, 2006. This brochure and recorded demonstrations are available at http://www.stilman-strategies.com
22. Lirov, Y., Rodin, E.Y., McElhaney, B.G., Wilbur, L.W. Artificial Intelligence Modeling of Control Systems, *Simulation*, pp.12-24, 1988.
23. McCrabb, M., Uncertainty, Expeditionary Air Force and Effects-Based Operations, AFRL, 2000.
24. McCrabb, M., Concept of Operations for Effect-Based Operations, v.2, AFRL, June 2000.
25. McCrabb, M., Effects-Based Operations: Examples & and Operational Requirements, June 2000.
26. McQuay, W., Stilman, B., Yakhnis, V., Distributed Collaborative Decision Support Environments for Predictive Awareness, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science*

IX", Orlando, FL, USA, 2005.
27. Newborn, M. (1996). *Computer Chess Comes of Age*, Springer-Verlag, New York, NY.
28. Newborn, M. (1997). *History of Chess Table,* Guest Essay, www.chess.ibm.com/learn
29. Osborn, M., Rubinstein, A., (1994) *A Course in Game Theory*, MIT Press, Cambridge, MA, 1994.
30. *Operations in a Low-Intensity Conflict* (1992), www.adtdl.army.mil/cgi-bin/atdl.dll/fm/7-98/f798.htm
31. Pugh, David M., Captain, USAF, *A Validation Assessment of the STORM Air-to-Air Prototype Algorithm*, AFIT. Also see http://www.s3i.com/STORM/overview.htm.
32. Stability and Support Operations (SASO), (2000) FAS, Military Analysis Network, at http://www.fas.org/man/dod-101/ops/saso.htm
33. Stilman, B., A Formal Language for Hierarchical Systems Control, *Int. J. Languages of Design*, Vol. 1, No.4, pp. 333-356, 1993.
34. Stilman, B., A Linguistic Approach to Geometric Reasoning, *Int. J. of Computers & Mathematics with Applications*, Vol. 26, No. 7, pp. 29-58, 1993,
35. Stilman, B., A Syntactic Hierarchy for Robotic Systems, *Int. J. Integrated Computer-Aided Engineering*, Vol. 1, No. 1, pp. 57-82, 1993.
36. Stilman, B., Network Languages for Complex Systems, *Int. J. of Computers & Mathematics with Applications*, Vol. 26, No. 8, pp. 51-80, 1993.
37. Stilman, B., Linguistic Geometry for Control Systems Design, *Int. J. of Computers and Their Applications*, 1(2): 89-110, 1994.
38. Stilman, B., Translations of Network Languages, *Int. J. of Computers & Mathematics with Applications*, Vol. 27, No. 2, pp. 65-98, 1994.
39. Stilman, B., Heuristic Networks for Space Exploration, Telematics and Informatics, Int. J. on Telecommunications & Information Technology, Vol. 11, No. 4, pp. 403-428, 1994.
40. Stilman, B., Linguistic Geometry of the Chess Model, *Advances in Computer Chess 7*, pp. 91-117, 1994.
41. Stilman, B., A Linguistic Geometry: Methodology and Techniques, *Cybernetics and Systems – An International Journal*, Vol. 26, No. 5, pp. 343-405, Sept. 1995.
42. Stilman, B., Network Languages for Intelligent Control, *Int. J. of Computers & Mathematics with Applications*, Vol. 31, No. 3, pp. 91-118, 1996.
43. Stilman, B., Managing Search Complexity in Linguistic Geometry, *IEEE Transactions on Systems, Man, and Cybernetics*, 27(6): 978-998, 1997.
44. Stilman, B., Network Languages for Concurrent Multi-agent Systems, *Intl. J. of Computers & Mathematics with Applications*, 34(1): 103-136, 1997.
45. Stilman, B., What is Linguistic Geometry?, *Advances in Systems Science and Applications*, Special Issue, pp. 154-163, Jan. 1997.
46. Stilman, B., Winning Strategies in Linguistic Geometry: A Formal Approach, *Advances in Systems Science and Applications*, Special Issue, pp. 630-644, Jan. 1997.
47. Stilman, B., Emergent Synthesis of Strategies in Linguistic Geometry, pp 27-36, *Proc. of the Int. Workshop on Emergent Synthesis*, Dec. 6-7, 1999, Kobe University, Kobe, Japan.
48. Stilman, B., Linguistic Geometry: From Search to Construction. Kluwer Acad. Publishers, 2000.
49. Stilman, B., Linguistic Geometry for Symmetric and Asymmetric War Games, pp. 431-449, Proc. of the 2[nd] Int. Conference on Computers and Games – CG'2000, Oct. 26-28, 2000, Hamamatsu, Japan.
50. Stilman, B., From Games to Intelligent Systems, pp. 779-786, *Proc. of the 2[nd] ICSC Int. Symp. on Engineering of Intelligent Systems* – EIS'2000, June 27-30, 2000, University of Paisley, UK.
51. Stilman, B., All in One: Fighting Adversaries, Complexity, Time: Advances in Linguistic Geometry, pp. 8-16, *Proc. of the Int. Conference on Artificial Intelligence in Science and Technology* – AISAT'2000, Dec. 17-20, 2000, The Univ. of Tasmania, Hobart, Australia.

52. Stilman, B., Air Traffic Management with LG Strategies, STILMAN Advanced Strategies, *Tech. REPORT to Boeing*, 23 pp., Dec. 2000.
53. Stilman, B., Application of Linguistic Geometry for Mid-Air Collision Avoidance, STILMAN Advanced Strategies, *Tech. REPORT to Boeing*, 28 pp., Dec. 2000.
54. Stilman, B., Dyer, D., Linguistic Geometry for aerospace combat simulation: serial and concurrent agents. *In Proc. 5th Int. Conf. on Human-Machine Interaction and Artificial Intelligence in Aerospace* (HMI-AI-AS'95), Toulouse, France, 1995.
55. Stilman, B., Fletcher, C., Systems Modeling in Linguistic Geometry: Natural and Artificial Conflicts, Intl. J.: Systems Analysis, Modeling, Simulation, pp. 57-97, vol. 33, 1998.
56. Stilman, B., Yakhnis, V., Solving Adversarial Control Problems with Abstract Board Games and Linguistic Geometry (LG) Strategies, pp. 11-23, *Proc. of the 1st Symp.: Advances in Enterprise Control*, JFACC Program, DARPA, ISO, Nov. 15-16, 1999, San Diego, CA, USA.
57. Stilman, B., Yakhnis, V., Adapting the Linguistic Geometry – Abstract Board Games Approach to Air Operations, pp. 219-234, *Proc. of the 2nd Symp.: Advances in Enterprise Control*, JFACC Program, DARPA, ISO, July 10-11, 2000, Minneapolis, MN, USA.
58. Stilman, B., Yakhnis, V., LG War Gaming for Effects Based Operations, STILMAN Advanced Strategies, *Tech. Report to Rockwell and Boeing*, 47 pp, July 2001.
59. Stilman, B., Yakhnis, V., Umanskiy, O., Winning Strategies for Robotic Wars: Defense Applications of Linguistic Geometry, *Artificial Life and Robotics*, pp. 148-155, Vol. 4, No. 3, 2000.
60. Stilman, B., Yakhnis, V., Umanskiy, O., Knowledge Acquisition and Strategy Generation with LG Wargaming Tools, *Int. J. of Computational Intelligence and Applications*, pp. 1-26, Vol.19(2), 2002.
61. Stilman, B., Yakhnis, V., Umanskiy, O., Hearing, J. Operational Level Decision Aids with LG-based Tools, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science VI"*, Orlando, FL, USA, 2002.
62. Stilman, B., Yakhnis, V., Umanskiy, O., Boyd, R., Deception Discovery and Employment with Linguistic Geometry, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28 – April 1, 2005, pp. 189-200, Orlando, FL, USA.
63. Stilman, B., Yakhnis, V., Umanskiy, O., Linguistic Geometry for Technology Procurement, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28 – April 1, 2005, pp. 201-211, Orlando, FL, USA.
64. Stilman, B., Yakhnis, V., Umanskiy, O., Boyd, R., Adversarial Reasoning and Resource Allocation: the LG Approach, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28 – April 1, 2005, pp. 177-188, Orlando, FL, USA.
65. McQuay, W., Stilman, B., Yakhnis, V., Distributed Collaborative Decision Support Environments for Predictive Awareness, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28 – April 1, 2005, pp. 212-223, Orlando, FL, USA.
66. Blakelock, C., Stilman, B., Yakhnis, V., Umanskiy, O., Boyd, R., LG Based Decision Aid for Naval Tactical Action Officer's (TAO) Workstation, *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28–April 1, 2005, pp. 236-246, Orlando, FL, USA.
67. Weber, R., Stilman, B., Yakhnis, V., Extension to the LG Hypergame to "Inner Game" played over topology of competing "mind nets", *Proc. of the SPIE Conference "Enabling Technology for Simulation Science IX"*, March 28 – April 1, 2005, pp. 224-235, Orlando, FL, USA.
68. Stilman, B., Yakhnis, V., Umanskiy, O., Boyd, R., LG-PACKAGE, The Ultimate Wargaming Environment, *Proc. of the Int. Conference on AI in Science and Technology* – AISAT'2004, Nov. 21-24, 2004, pp.1-7, UTAS, Hobart, Australia.
69. Stilman, B., Yakhnis, V., Umanskiy, O., Boyd, R., LG Wargaming for Simulation Based Acquisition, *Proc. of the Int. Conference on AI in Science and Technology* – AISAT'2004, Nov. 21-24, 2004, pp. 289-294, UTAS, Hobart, Australia.

70. Tucker, J., (1999) Asymmetric Warfare, FORUM for Applied Research and Public Policy, http://forum.ra.utk.edu/1999summer/asymmetric.htm
71. Turek, R. The Application of Linguistic Geometry to the Routing of Emergency Vehicles, Advances in Systems Science and Applications, (186-195), Jan. 1997.
72. Von Neumann, J., Morgenstern, O. *Theory of Games & Economic Behavior*, Princeton U. Press, 1944
73. Walker, P. (2001) *A Chronology of Game Theory*, at http://www.econ.canterbery.ac.nz/hist.htm
74. Williams, J.D., The Complete Strategist, Dover Publications, 1986.
75. Yakhnis, A., Yakhnis, V. Gurevich-Harrington's Games Defined by Finite Automata, *Annals of Pure and Applied Logic*, Vol. 62, pp. 265-294, 1993.
76. Yakhnis, V., Stilman, B., A Multi-Agent Graph-Game Approach to Theoretical Foundations of Linguistic Geometry. *Proc. of the Second World Conference on the Fundamentals of Artificial Intelligence* (WOCFAI 95), Paris, France, July, 1995.
77. Yakhnis, V., Yakhnis, A., and Stilman, B., Linguistic Geometry and Board Games Approach to Automated Generation of Schedules, *Proc. 15th IMACS World Congress on Scientific Computation, Modeling and Applied Mathematics*, Berlin, Germany, 1997.
78. Zermelo, E. (1913), Uber eine Anwendung der Mengenlehre auf die Theorie des Schachspiels, pp. 501-504, *Proc. of the Fifth International Congress of Mathematicians*, Vol. II, Cambridge Univ. Press.