Thought Experiments in Linguistic Geometry

Boris Stilman

STILMAN Advanced Strategies, Denver, CO, USA &
University of Colorado Denver, Denver, CO, USA
boris@stilman-strategies.com

***Abstract*—Linguistic Geometry (LG) is a type of game theory for extensive discrete games scalable to the level of real life defense systems. This scalability is based on changing the paradigm for game solving - from search to construction. LG was developed by generalizing experiences of the advanced chess players. LG is a formal model of human reasoning about armed conflict, a mental reality "hard-wired" in the human brain. This algorithm is an evolutionary product of millions of years of human warfare served in its turn as the principle mover for evolution of human intelligence. Its special role in human history and long coexistence with the Primary Language of the human brain (as introduced by J. Von Neumann) and the Algorithm of Discovery suggest utilizing LG for investigation of those two puzzles. This paper refining our experiences of discovering LG is the first step in this direction.**

Keywords—Linguistic Geometry; Primary Language; Artificial Intelligence; algorithm of discovery; game theory

I. Introduction

Linguistic Geometry (LG) [15] is a game-theoretic approach that has demonstrated a significant increase in size of problems solvable in real time (or near real time). This paper continues a series of papers [21], [22], [16], intended to discover role of LG in human culture.

Def. 1. LG is intended for solving Abstract Board Games (ABG) defined as follows (see full version in [15]):

$$< X, P, R_p, SPACE, val, S_0, S_t, TR>,$$

where

$X = \{x_i\}$ is a finite set of points (abstract board);

$P = \{p_i\}$ is a finite set of pieces; $P = P_1 \cup P_2$ called the opposing sides;

$R_p(x, y)$ is a set of binary relations of reachability in X ($x \in X$, $y \in X$, and $p \in P$);

val is a function on representing the values of pieces;

SPACE is the state space;

S_0 and S_t are the sets of start and target states. $S_t = S_t^1 \cup S_t^2 \cup S_t^3$, where all three are disjoint. S_t^1, S_t^2 are the subsets of target states for the opposing sides P_1 and P_2, respectively. S_t^3 is the subset of target draw states.

TR is a set of transitions (moves) of ABG between states.

The goal of each side is to reach a state from its subset of target states, S_t^1 or S_t^2, respectively, or, at least, a draw state from S_t^3. The problem of the optimal operation of ABG is considered as a problem of finding a sequence of transitions leading from a start state of S_0 to a target state of S_t assuming that each side makes only the best moves (if known), i.e., such moves that could lead ABG to the respective subset of target states. To solve ABG means to find a strategy (an algorithm to select moves) for one side, if it exists, that guarantees that the proper subset of target states will be reached assuming that the other side makes arbitrary moves.

The word *Linguistic* refers to the model of strategies formalized as a hierarchy of formal languages. These languages describe states of the game as well as moves from state to state. They utilize a powerful class of generating grammars, the controlled grammars [15], which employ formal semantics of the game to control generation of a string of symbols using mutual influence of the substring generated so far and the grammar's environment.

The word *Geometry* refers to the geometry of the game state space SPACE (Def. 1), which is a set of all the states resulting from all legal plays of ABG leading from a start state. Every state is an abstract board X with abstract pieces P, i.e., mobile entities, located on this board and acting upon each other. Thus, different states include the same board with different configurations of pieces resulting from the sequence of moves. In LG, the geometry of the state space is effectively reduced to the geometry of the board, which can also be called a game space. Thus, the state space is reduced to the projection of the "space-time" over "space", by introducing abstract relations defining the movements and other actions of the pieces as well as their mutual influence. This projection leads to efficient decomposition of the state space that permits replacing search by construction of strategies [15].

LG is a viable approach for solving board games such as the game of chess as well as practical problems such as mission planning and battle management. Historically, LG was developed, beginning from 1972, by generalizing experiences of the most advanced chess players including World Chess Champions and grandmasters [1], [15]. In the 70s and 80s this generalization resulted in the development of computer chess program PIONEER utilized successfully for solving chess endgames and complex chess positions with a number of variations considered in the order of 10^2 while the state spaces of those problems varied from 10^{10} to 15^{25}. The variations constructed by PIONEER were very close to those considered by the advanced chess experts when analyzing the same problems. Further generalization led to development of the new type of game theory, LG,

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-155-7

changing the paradigm for solving game problems: "From Search to Construction" [15]. An LG-based technology was applied to more than 30 real life defense related projects [6]. On multiple experiments, LG successfully demonstrated the ability to solve extremely complex modern military scenarios in real time. The efficacy and sophistication of the courses of action developed by the LG tools exceeded consistently those developed by the commanders and staff members [19]-[20].

Almost forty years of development of LG including numerous successful applications to board games and, most importantly, to a highly diverse set of modern military operations [4]-[6], [20]-[22], from cruise missiles to military operations in urban terrain to ballistic missile defense to naval engagements, led us to believe that LG is something more fundamental than simply yet another mathematical model of efficient wargaming.

A universal applicability of LG in a variety of military domains, especially, in the domain of the ancient warfare, its total independence of nationality or country, its power in generating human-like strategies suggest that the algorithm of LG utilized by the human brain is "hard-wired" in the Primary Language (introduced by J. Von Neumann [23]). Moreover, the age of the Primary Language must be much greater than the age of human natural languages, and so the age of LG [16]. A highly intriguing and difficult issue is an algorithm of discovery, i.e., an algorithm of inventing new algorithms and new models. This algorithm should also be a major ancient item "recorded" in the Primary Language. In this paper, by investigating past discoveries, experiences of construction of various new algorithms, and the heritage of LG, we will make a step towards understanding of this puzzle.

II. BACK TO THE ORIGIN

In [21], [22], [16], we suggested that the game of chess served as a means for discovering LG. The original theory of LG was developed by generalizing algorithms implemented in the computer chess program PIONEER [1], [15]. Simultaneously, some of the similar algorithms were utilized for economic problems in the former USSR (programs PIONEER 2.0-4.0). Further development of LG, of course, took advantage of these new applications. However, the original major framework of LG, the hierarchy of three formal languages, was based exclusively on the chess domain, the only application available at that time. We must admit that over the following 30 years the structure of this framework has never changed.

By the end of the 80s, PIONEER solved a number of sophisticated endgames and positions but still could not play full games. It was clear for the developers that the main ideas are correct but further development for the chess domain was required. It was also expected that transferring LG to other domains, e.g., defense, should be tried only after the chess model would be completed. Besides incompleteness of this model, a number of other serious limitations based on the awkward nature of the game of chess (in comparison with real life) could have prevented from such transfer. These limitations were as follows.

- Totally discrete nature while the real world is mostly continuous (on macro level).
- Small number of agents while in the real world problems this number is often several orders of magnitude greater.
- Serial movement of agents in comparison with the real world agents such as humans, human-controlled vehicles, robots, etc. moving concurrently.
- Simplistic 2D mobility rules in comparison with sophisticated mobility patterns of the real world agents, which may require multi-dimensional phase space.
- Small, non-sophisticated 2D game board in comparison with real world 3D large-scale terrains with multiple different features.
- Awkward goals like checkmate in comparison with real life goals that vary significantly.

In addition, there was no theoretical evaluation of the accuracy of the LG solutions except for those experiments with chess positions approved by the chess experts.

The advanced version of LG, completed by Dr. Stilman by the end of the 90s [15], with contributions of Drs. V. Yakhnis and A. Yakhnis, had overcome some of the above limitations by further "wild" generalizations (Def. 1) and mathematical proofs of correctness of the algorithms and accuracy of the solutions. The major part of this research included thought experiments with applications of the new LG to the extended chess domain, such as the games with 3D board and concurrent movements. All the constructions of the old and new LG were tested in the thought experiments. Moreover, many of those constructions were conceived originally during such experiments. The constructions that successfully passed thought experiment (and some alternative constructions) were programmed and tested employing software applications [1], [15].

The new LG of the 90s definitely covered a number of different real life problem domains as many other mathematical theories do. But was it really an adequate model? In Physics, this means predicting results of experiments. In Computer Science, a requirement is similar. Software applications based on the new theory should yield plausible or satisfactory solutions for a new domain. In case of LG, this means consistently generating plans, i.e., military courses of action, comparable or even better than those of military experts. When the LG applications started consistently generate advanced courses of action for a number of defense sub-domains, the developers realized that the game of chess served the role of the eraser of particulars for the real world warfare. From the bird's eye view,

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-155-7

military operations are incomparably more complex than this game. Interestingly, this fact was pointed out by many reviewers of our papers with the first generalizations of the original LG in the 90s. All the above limitations that could have prevented us from transferring LG to the real world, in a sense, enabled us to see the essence behind numerous particulars. Of course, we still needed an advanced chess player like World Chess Champion Professor Botvinnik who was able to analyze the chess master's approach to solving problems. We could only guess if such a grandmaster-commander capable of doing the same for the military strategies would have ever appeared. With all the ingenuity of such an expert, a task of refining the military strategies down to trajectories and networks of trajectories would have been significantly more complex due to those particulars that mud the picture.

III. Thought Experiments

Thought experiments allow us, by pure reflection, to draw conclusions about the laws of nature [2]. For example, Galileo before even starting dropping stones from the Tower in Pisa, used pure imaginative reasoning to conclude that two bodies of different masses fall at the same speed. The Albert Einstein's thought experiments inspiring his ideas of the special and general relativity are known even better. The efficiency and the very possibility of thought experiments show that our mind incorporates animated models of the reality, e.g., laws of physics, mathematics, human activities, etc. Scientists managed to decode some of the human mental images by visualizing their traces on the cortex [3]. It was shown that when we imagine a shape "in the mind's eye", the activity in the visual areas of the brain sketches the contours of the imagined object, thus, mental images have the analogical nature [2]. It appears that we simulate the laws of nature by physically reflecting the reality in our brain. The human species and even animals would have had difficulty to survive without even minimal "understanding" of the laws of environment. Over the course of evolution and during development of every organism, our nervous system learns to comprehend its environment, i.e., to "literally take it into ourselves" in the form of mental images, which is a small scale reproduction of the laws of nature. Neuropsychologists discovered that "we carry within ourselves a universe of mental objects whose laws imitate those of physics and geometry" [2]. In [16], we suggested that we also carry the laws of the major human relations including the laws of optimal warfighting. The laws of nature and human relations manifest themselves in many different ways. However, the clearest manifestation is in perception and in action. For example, we can say that the sensorimotor system of the human brain "understands kinematics" when it anticipates the trajectories of objects. It is really fascinating that these same "laws continue to be applicable in the absence of any action or perception when we merely imagine a moving object or a trajectory on a map" [2]. This observation, of course, covers actions of all kinds of objects, natural and artificial. Scientists have shown that the time needed to rotate or explore these mental images follows a linear function of the angle or distance traveled as if we really traveled with a constant speed. They concluded that "mental *trajectory* imitates that of a physical object" [2]. Further, we will consider mechanics of the thought experiments in the development of LG and, specifically, in the development of the advanced LG applicable to the defense problems.

IV. 2D/4A Experiment

The typical thought experiments in LG focused on several chess problems and variations of those problems. The major problem of this kind is the so-called 2D/4A problem [15]. This is a problem of simplified "air combat" with four aircraft and 2D operational district. It was simple enough to be used for the development of the original chess LG, for the first demonstration of the LG approach, and for generalization. Nevertheless, the 2D/4A is not trivial and requires approximately 9^{12} move search tree to be solved employing brute force search. This problem is an alteration of the famous Reti endgame for the game of chess. This endgame was compiled by Richard Reti in 1921. Program PIONEER solved this endgame in 1977 employing the search tree that includes 54 moves [1], [15].

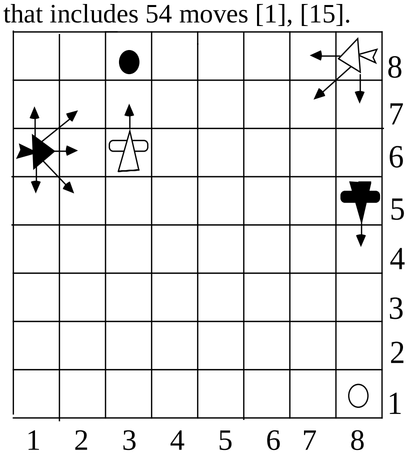

Figure 1. The 2D/4A serial problem with 8×8 district

The major components of the 2D/4A ABG (Def. 1) are as follows (Fig. 1). The abstract board X represents the area of combat operation, a 2D grid of 8×8. P is the set of robots or autonomous vehicles. It is partitioned into two subsets P_1 and P_2 (White and Black) with opposing interests. Relation $R_p(x, y)$ represents the moving abilities of robots, i.e., robot p can move from point x to point y if $R_p(x, y)$ holds.

Robot W-FIGHTER (White Fighter), standing on 88, can move to any adjacent square (shown by arrows). Thus, robot W-FIGHTER on 88 can reach any of the points y $\in$ {87, 77, 78} in one step, i.e., $R_{W\text{-}FIGHTER}(88, y)$ holds. The other robot, B-BOMBER (Black Bomber) from 85, can move only straight ahead, one square at a time, e.g., from 85 to 84, from 84 to 83, etc. Robot B-FIGHTER (Black Fighter) standing on 16, can move to any adjacent square similarly to robot W-FIGHTER. Robot W-BOMBER standing on 36 is analogous with the robot B-BOMBER; it can move only straight ahead but in the opposite direction (it can reach only 37 in one step).

Assume that robots W-FIGHTER and W-BOMBER belong to the White side, while B-FIGHTER and B-BOMBER belong to the opposing Black side: W-FIGHTER, W-BOMBER $\in P_1$, B-FIGHTER, B-BOMBER $\in P_2$. Also, assume that two more robots, W-TARGET and B-TARGET, (immobile devices or target areas) are located at 81 and 38, respectively: W-TARGET $\in P_1$, B-TARGET $\in P_2$. Each of the BOMBERs can destroy immobile TARGET ahead of its course; it also has powerful weapons able to destroy opposing FIGHTERs on the adjacent diagonal squares ahead of its course. For example, W-BOMBER from 36 can destroy opposing FIGHTERs on 27 and 47. Each of the FIGHTERs is capable of destroying an opposing BOMBER by approaching its location and moving there. But, it is also able to protect its friendly BOMBER on the adjacent locations. In the latter case, the joint protective power of the combined weapons of the friendly BOMBER and FIGHTER can protect the BOMBER from an interception. For example, the W-FIGHTER located at 46 can protect W-BOMBER on 36 and 37. Assume that the moves of the opposing sides *alternate* and *only one piece at a time* can move.

The combat considered can be broken down into two local operations. The first operation is as follows: robot B-BOMBER should reach point 81 to destroy the W-TARGET, while W-FIGHTER will try to intercept the B-BOMBER. The second operation is similar: robot W-BOMBER should reach point 38 to destroy the B-TARGET, while B-FIGHTER will try to intercept the W-BOMBER. Interception is impossible after a BOMBER hits a TARGET and stays safe for at least one time interval. After destroying the opposing TARGET and saving its BOMBER, the attacking side is considered a winner of the local operation. The only chance for the opposing side to avenge is to hit its TARGET, save its BOMBER for one time interval after that, and, this way, end the battle in a draw.

Let S_t^1 (the set of winning target states for White) be the set of states where B-BOMBER is destroyed, and W-BOMBER hit B-TARGET and has been safe for at least one time interval. Let S_t^2 (the set of winning target states for Black) be the set of states where W-BOMBER is destroyed, and B-BOMBER hit W-TARGET and has been safe for at least one time interval. Let S_t^3 (the set of draw states) be the set of states where both BOMBERs hit their targets and stay safe for at least one time interval, or both BOMBERs are destroyed before they hit their targets or immediately after that. Start State S_0 is shown in (Fig. 1). *Is there a strategy for White to force a draw*?

The draw strategy generated by the LG algorithm is to move W-FIGHTER along the diagonal to 77, 66 and, in some cases, to 55. It should deviate from this diagonal movement in response to the activities of the Black. In all cases employing such strategy it would have enough time either to approach W-BOMBER at 36 and support its safe attack of W-TARGET or to intercept B-BOMBER before it hits its target (or immediately after that).

The initial set of thought experiments with Reti endgame and 2D/4A problem consisted in mental execution of various versions of the LG algorithm and subsequent verification of those employing program PIONEER. The purpose of the next experiment was to investigate the impact of an increase in "dimension" of the abstract board and sophistication of the reachability relations.

V. 3D/4A EXPERIMENT

The 3D/4A thought experiment was constructed by morphing the start state S_0 of the 2D/4A problem (Fig. 1) into the start state of 3D/4A (Fig. 2) [15]. The key constraints for this morphing were based on the preservation of the R. Reti's idea for W-INTERCEPTOR to be able to either protect W-STATION or to intercept B-STATION. As was the case with the 2D/4A problem, all the considerations were based on the Euclidean distances on the abstract 3D board (not in the state space). Recall that the definition of ABG (Def. 1) does not impose any constraints on the abstract board X – the board simply expands from 64 to 512 points. However, all the sophistication of the 3D/4A problem is built into the reachability relations of the INTERCEPTORs. They are able to move to the adjacent cubes in three layers, current, top and bottom.

The operational district X is the 3D grid of 8×8×8. Robot W-INTERCEPTOR (White Interceptor), located at 118 ($x = 1$, $y = 1$, $z = 8$), can move to any adjacent location, i.e., 117, 217, 218, 228, 227, 128, 127. Robot B-STATION (double-ring shape in Fig. 2) at 416, can move only straight ahead towards the goal area 816 (shaded), one cube area at a time, e.g., from 416 to 516, from 516 to 616, etc. Robot B-INTERCEPTOR (Black Interceptor), located at 186, can move to any adjacent square, just as robot W-INTERCEPTOR. Robotic vehicle W-STATION, located at 266, is analogous with robotic B-STATION; it can move only straight ahead towards the goal area 268 (shaded in (Fig. 2). Thus, robot W-INTERCEPTOR at 118 can reach any of the points $y \in \{117, 217, 218, 228, 227, 128, 127\}$ in one step, i.e., relation $R_{W\text{-}INTERCEPTOR}(118, y)$ holds, while W-STATION can reach only 267 in one step.

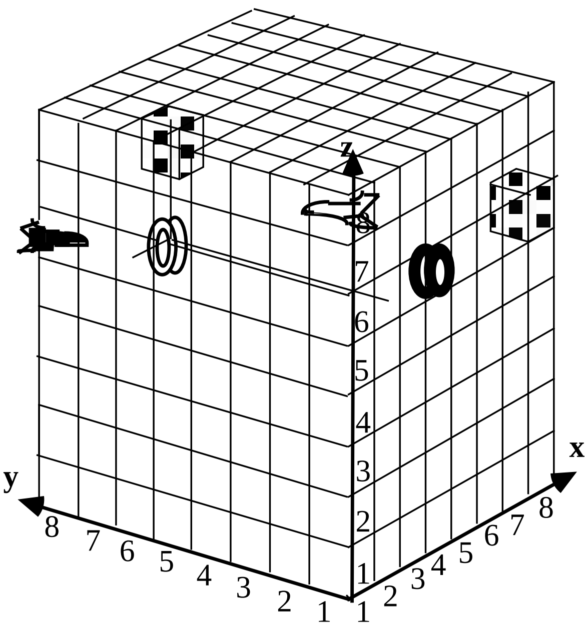

Figure 2. The 3D/4A serial problem

Assume that robots W-INTERCEPTOR and W-STATION belong to one side, P_1, while B-INTERCEPTOR and B-STATION belong to the opposing side, P_2. Also assume that both goal areas, 816 and 268, are the safe areas for B-STATION and W-STATION, respectively, if they reach the area and stay there for more than one time interval. Each of the STATIONs has weapons capable of destroying opposing INTERCEPTORs at the forward adjacent diagonal locations. For example, W-STATION at 266 can destroy opposing INTERCEPTORs at 157, 257, 357, 367, 377, 277, 177, 167. Each of the INTERCEPTORs is able to destroy an opposing STATION approaching its location from any direction, but it is also capable of protecting its friendly STATION. In the latter case, the joint protective power of the combined weapons of the friendly STATION and INTERCEPTOR (from any area adjacent to the STATION) can protect the STATION from an interception. For example, W-INTERCEPTOR located at 156 can protect W-STATION on 266 and 267. As in the 2D case, we assume that the moves of the opposing sides *alternate* and *only one piece at a time* can move.

The 3D combat can be broken into two local operations. The first operation is as follows: B-STATION should reach strategic point 816 safely, while W-INTERCEPTOR will try to intercept B-STATION. The second operation is similar: W-STATION should reach point 268, while B-INTERCEPTOR will try to intercept W-STATION. Interception is impossible after a STATION reaches the strategic point and stays safe for at least one time interval. After reaching safely its strategic point, the (attack) side is considered a winner of the local operation. The only chance for the opposing side to avenge is to reach safely its own strategic area and, this way, end the battle in a draw.

Let S_t^1 be the set of states where B-STATION is destroyed, and W-STATION reached strategic point 268 and has been safe for at least one time interval. Let S_t^2 be the set of states where W-STATION is destroyed, and B-STATION reached strategic point 816 and has been safe for at least one time interval. Let S_t^3 be the set of states where both STATIONs reached their strategic points and stay safe for at least one time interval, or both STATIONs are destroyed before they reached their targets, or immediately thereafter. The Start State S_0 is shown in Fig. 2.

As in the 2D problem, it seems that local operations are independent, because they are located far from each other. Moreover, the operation of B-STATION from 418 looks like an unconditionally winning operation, and, consequently, the global battle can be easily won by Black. *Is there a strategy for White to force a draw*?

The draw strategy generated by the LG algorithm is to move W-INTERCEPTOR along the main diagonal of the cube from 118 to 227 to 336, and, in some cases, to 445. There are several optional draw strategies. They include also moves "around" the main diagonal of the cube. In contrast to the 2D/4A problem, where W-FIGHTER has to follow exactly the diagonal of the square (Section IV), here, W-INTERCEPTOR has a number of options in moving around the main diagonal such as locations 337 and 338 (from 227). At some moment, following the response of Black, White should deviate from the diagonal by approaching either W-STATION or B-STATION.

VI. Experiment with Total Concurrency

The next thought experiment was constructed by morphing the 2D/4A problem (Section IV) to achieve total concurrency (TC) and variable size district, [15]. This morphing was made in two stages, from serial to alternating concurrent (AC, i.e., both sides alternate but pieces can move concurrently for each side) to totally concurrent (TC). As was the case for the 3D/4A problem, the key constraints for the morphing were based on the preservation of the R. Reti's idea for W-FIGHTER to be able to either protect W-BOMBER or to intercept B-BOMBER. It appears that this idea is inherent to the serial motion. To preserve it in the concurrent environment we introduced the awkward condition of "remote destruction" of the armed BOMBERs.

The operational district X is a 2D n×n square grid, $n > 7$. W-FIGHTER located at 11, can move to any adjacent square. It can reach any of the points $y \in \{12, 22, 21\}$ in one step, i.e., $R_{\text{W-FIGHTER}}(11, y)$ holds. B-BOMBER from 12 can move only straight ahead, one square at a time, e.g., from 12 to 13, from 13 to 14, etc. B-FIGHTER located at 83 can move to any adjacent square. W-BOMBER located at 63 is analogous with the robot B-BOMBER; it can move only straight ahead but in opposite direction. It can reach only 62 in one step.

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-155-7

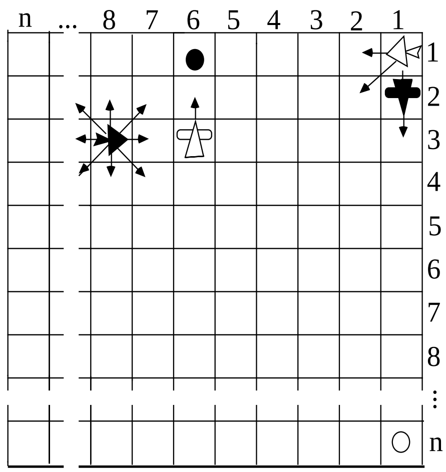

Figure 3. The 2D/4A TC problem for the n×n district

Assume that robots W-FIGHTER and W-BOMBER belong to one side, P_1, while B-FIGHTER and B-BOMBER belong to the opposing side, P_2. Also, assume that two robots, W-TARGET and B-TARGET, (immobile devices or target areas) are located at 1n and 61, respectively. W-TARGET $\in P_1$, while B-TARGET $\in P_2$. Each of the BOMBERs can destroy its immobile TARGET ahead. Each of the FIGHTERs is able to destroy the opposing BOMBER by moving to its location, but it is also able to destroy an opposing BOMBER if this BOMBER itself arrives at the current FIGHTER's location or at the location where the FIGHTER arrives simultaneously with the opposing BOMBER. For example, if B-FIGHTER is at location 61 and W-BOMBER arrives there (unprotected) then during the same time interval it destroys B-TARGET and is destroyed itself by B-FIGHTER. BOMBERs cannot destroy FIGHTERs. Each BOMBER can be protected by its friendly FIGHTER if it is at the location adjacent to the BOMBER. In this case, the joint protective power of the weapons of the friendly BOMBER and FIGHTER can protect the BOMBER from an interception. For example, W-FIGHTER located at 53 can protect W-BOMBER at 63 and 62.

Assume that *all the robots* can move *simultaneously* and there is no alternation of turns. This means that during the current time interval, all four vehicles, W-BOMBER, W-FIGHTER, B-BOMBER, and B-FIGHTER, three of them, two, one, or none of them, can move. Thus, every concurrent move could be considered as a 4-move, a 3-move, a 2-move or a 1-move.

As in all the TC systems [15], this is a model with incomplete information about the current move (before it is made). When moving, each side does not know the opposing side's component of the concurrent move, i.e., the immediate moves of the opposing side, if they are not limited down to the specific one or zero moves and, thus, can be predicted. Moreover, even after developing a deterministic strategy a side cannot follow it, because of the uncertainty about the concurrent moves of the opposing side. However, if the strategy resulted in the variants of concurrent moves with a single "universal" component (group of moves) for one side, which is good for all possible components of the other side, this strategy *can be* implemented.

As discussed at the beginning of this Section, to preserve the Reti's idea, we introduced the condition of remote destruction as follows. Each of the BOMBERs is vulnerable not only to a FIGHTER's attack, but also to the explosion of another BOMBER. If W-FIGHTER hits B-BOMBER while the latter is fully armed, i.e., it is not at its final destination – square 1n, and W-BOMBER is moving during the same time interval, it will be destroyed as a result of the B-BOMBER's explosion. If W-BOMBER is not moving at this moment, it is safe. Similar condition holds for B-BOMBER: it should not move at the moment when W-BOMBER is being destroyed (excluding 61). Therefore, under certain conditions, destruction of one of the BOMBERs triggers the explosion of the other one. This may be, for example, a result of a sudden change of atmospheric conditions or an electromagnetic field caused by a nuclear explosion.

Let S_t^1 be the set of states where the B-BOMBER is destroyed and W-BOMBER hit B-TARGET and has been safe during the hit. Let S_t^2 be the set of states where the W-BOMBER is destroyed, and B-BOMBER hit W-TARGET and has been safe during the hit. Let S_t^3 be the set of states where both BOMBERs hit their targets and stay safe, or both BOMBERs are destroyed before they hit their targets or during these hits. Start state S_0 is shown in Fig. 3.

The combat considered can be broken down into two local operations. The first operation is as follows: robot B-BOMBER should reach location 1n to destroy the W-TARGET, while the W-FIGHTER will try to intercept this movement. The second operation is similar: robot W-BOMBER should reach location 61 to destroy the B-TARGET, while B-FIGHTER will try to intercept this movement. Interception is impossible after a BOMBER has hit a TARGET and stayed safe during this hit. After destroying the opposing TARGET and keeping its BOMBER safe, the attacking side is considered a winner of the local operation. The only chance for the opposing side to avenge is to do the same: to hit its TARGET and keep its BOMBER safe. This will end the battle in a draw.

Is there a strategy for White to force a draw, i.e., a strategy that provides one of the following: both BOMBERs hit their targets and none of the BOMBERs is destroyed at the moment of strike, or both BOMBERs are destroyed before they hit their targets or at the moment of strike?

The conclusive draw strategy generated by LG is to move W-FIGHTER from 11 along the diagonal to 22, 33, 44 and keep W-BOMBER at 63. From there, W-BOMBER and W-FIGHTER should move as a pair, simultaneously, to 62 and 53, then to 61 and 52, respectively. This variant leads to the safe attack of both targets. All other variants of draw are inconclusive – those strategies could be implemented with probability of 50% only.

VII. Experiment with Direct Construction

For decades the accuracy of the solutions generated by prototypes and applications of LG was evaluated with respect to the solutions obtained from the most advanced experts in the field. In particular, these were chess experts, authorities in power maintenance, vehicles routing, etc. Usually, the solutions generated by the LG systems were approved by domain experts or matched those published in the domain literature (like chess). However, nobody claimed that these published solutions are provably optimal. They are solutions which experts agreed upon.

The same approach was used for the development of the thought experiments described in Sections IV-VI and respective software applications. Our assumption was that the solution sought by the LG systems should be good, satisfactory, but not necessarily optimal. In our attempt to evaluate accuracy of the solution of the 2D/4A problem we followed the same pattern. By running thought experiments for solving 2D/4A (as well as other problems) by executing the LG algorithm we tried to evaluate the accuracy of the solutions. This means that we would have to evaluate the error, i.e., the "distance" between two solutions, the solution generated by LG and the optimal solution. Both solutions could be represented as variants of moves in the state space leading from the Start state. It is extremely difficult to understand the meaning of "distance" between two variant-solutions in the state space, especially, if the problem is an opposing game. By morphing the 2D/4A thought experiment in several directions and trying to visualize the notion of "distance" I suddenly realized that we can measure distances between subspaces of the state space (sets of states) by projecting these subspaces on the abstract board. This realization did not help us in evaluating the error, i.e., the distance between solutions. However, it helped to evaluate classes of potential solutions, i.e., potential strategies. Indeed, from the LG algorithm it was known that two areas of the abstract board, the upper left and the bottom right networks (Fig. 4, left), are highly desirable for White. This means that arrival of W-FIGHTER in at least one of them may lead to the draw, i.e., the draw strategy. Such an arrival might happen through the entry points, the so-called zone gateways (double circles in Fig. 4).

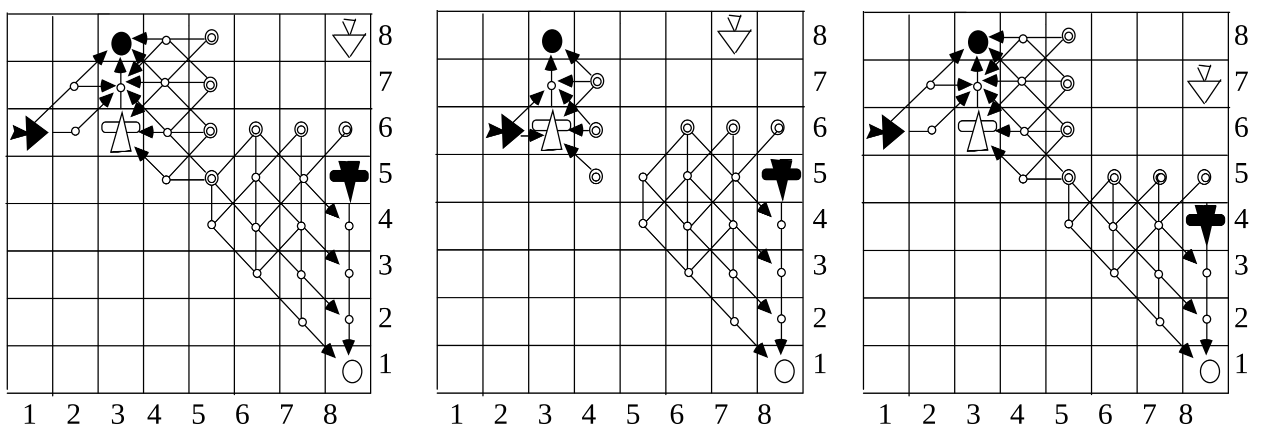

Figure 4. Projections of Draw Subspaces for three different states for the 2D/4A problem

It appears that these two networks represent projections (on the board) of the 2D/4A subspaces where the draw strategy exists. This means that the LG algorithm can identify the subspaces such that for every state from those there is a strategy leading to the draw. We can think about these subspaces as "black holes" – once you have got there you would not get out, i.e., the draw is guaranteed.

Unfortunately, these "black holes" depend on the location of the observer, i.e., on the current state. When the game moves to another state, the draw subspaces may shrink, expand, or even disappear. Fig. 4 shows projections of the "black holes" for 3 different states of the 2D/4A problem. Another difficulty is related to the presence of adversarial pieces – they may interfere. If W-FIGHTER gets into one of those areas on the board this does not assure that the game would actually get into the proper subspace – it is an effect of projection. However, it is known that in order for the game to eventually get to the draw target state W-FIGHTER must cross the boundaries of one of those areas. A simple analogy is for a point moving in 3D space in an attempt to get into the complex 3D shape. Consider a stationary 2D plain. If an orthogonal projection of this point on this plain gets inside the projection of the 3D shape (on the same plain), does it mean that our point actually reached inside the shape? Certainly, it is not the case. However, this is a necessary condition. Analogously, for W-FIGHTER to reach one of those areas is a necessary condition to achieve

Copyright (c) IARIA, 2011. ISBN: 978-1-61208-155-7

a draw. Consequently, if W-FIGHTER cannot reach those projections, there is no way the system can get into the draw subspaces from the start state (Fig. 4, left) - the "black holes" would be unreachable.

Now, we can come back to the notion of state distance. Consider the "board distances" (number of steps for W-FIGHTER) between the W-FIGHTER's current location at 81 and the entry points of the networks (the double circles) as the bottom values of the respective state distances. This means that the distance between the start state and the draw subspaces cannot be less than the board distances just described. Let us check these board distances (Fig. 4, left). The board distances from 88 to the entry points of the upper left network, 55, 56, 57 and 58, are the same and equal 3 steps. The board distances from 88 to entry points of the bottom right network are equal to 2 steps.

As I already mentioned the notion of state distances discovered from the thought experiment could be used to evaluate classes of potential strategies. Indeed, consider all the strategies leading to one of the draw subspaces. As we just realized, all of them must include the movement of W-FIGHTER approaching at least one of the areas, top or bottom (as a necessary condition). This means that the total of board distances must shrink. If W-FIGHTER moves 88-78, then Black "gets the message" about the specific draw subspace White is trying to approach. Based on this message, Black responds with B-FIGHTER 16-26; the draw subspace and its projection shrink (Fig. 4, middle) while both of the board distances stay the same, 2 and 3. If W-FIGHTER moves 88-87, then Black also gets the message about the specific draw subspace White is trying to approach. Based on this message, Black responds with B-BOMBER 85-84; the draw subspace and its projection shrink (Fig. 4, right) while both of the board distances stay the same, 2 and 3. In both cases W-FIGHTER approaches nothing because the board distances do not shrink. The only potential strategy that may lead to a draw must include diagonal movement of W-FIGHTER. For example, if W-FIGHTER moves 88-77, then Black gets the mixed message about a target of this movement because it is not clear which subspace is being approached. It does not matter how Black responds, i.e., which projection shrinks. In all cases, at least one of the distances will be reduced. This way we can eliminate all the potential draw strategies that do not include diagonal movement of W-FIGHTER as non-implementable.

The thought experiment described in this Section allowed us to develop the no-search approach in LG [13], [15], which permits generating solutions by direct construction (without search at all). Simultaneously, it includes proof of optimality of the constructed solution.

VIII. Conclusion and Future Works

This paper is the first step in our research, discovering the algorithm for inventing new algorithms. For this research we employed several thought experiments utilized over the years for developing LG. The preliminary conclusion (to be verified in our future research) is that these inventions never included the "search per se". Instead we morphed under certain constraints visual images of the existing dynamic objects into the new objects.

References

[1] M.M. Botvinnik, *Computers in Chess: Solving Inexact Search Problems*, Springer-Verlag, 1984.

[2] S. Deheaene, A Few Steps Toward a Science of Mental Life, *Mind, Brain and Education*, Vol. 1, No. 1, pp. 28-47, 2007.

[3] S. Kosslyn, W. Thompson, I. Kim, and N. Alpert, Representations of mental images in primary visual cortex, *Nature*, 378 , 496–498, 1995.

[4] A. Kott (editor), *Advanced Technology Concepts for Command and Control*, Xlibris Corporation, 2004.

[5] A. Kott and W. McEneaney (editors), *Adversarial Reasoning: Computational Approaches to Reading the Opponent's Mind*, by, Chapman & Hall/CRC, 2007.

[6] *Linguistic Geometry Tools: LG-PACKAGE*, with Demo DVD, 60 pp., STILMAN Advanced Strategies, 2010. This brochure and 8 recorded demonstrations are also available at www.stilman-strategies.com

[7] B. Stilman, Formation of the Set of Trajectory Bundles, Appendix 1 to the book: *On the Cybernetic Goal of Games*, by Botvinnik, M. M., Soviet Radio, Moscow (in Russian), pp. 70–77, 1975.

[8] B. Stilman, A Formal Language for Hierarchical Systems Control, *Int. J. Languages of Design*, Vol. 1, No.4, pp. 333-356, 1993.

[9] B. Stilman, A Linguistic Approach to Geometric Reasoning, *Int. J. of Computers & Math. with Appl*, Vol. 26, No. 7, pp. 29-58, 1993,

[10] B. Stilman, Network Languages for Complex Systems, *Int. J. of Computers & Math. with Appl.*, Vol. 26, No. 8, pp. 51-80, 1993.

[11] B. Stilman, Linguistic Geometry for Control Systems Design, *Int. J. of Computers and Their Applications*, 1(2): 89-110, 1994.

[12] B. Stilman, Translations of Network Languages, *Int. J. of Computers & Math. with Appl.*, Vol. 27, No. 2, pp. 65-98, 1994.

[13] B. Stilman, Managing Search Complexity in Linguistic Geometry, *IEEE Trans. on Syst., Man, and Cybernetics*, 27(6): 978-998, 1997.

[14] B. Stilman, Network Languages for Concurrent Multi-agent Systems, *Intl. J. of Computers & Math. with Appl*., 34(1): 103-136, 1997.

[15] B. Stilman, *Linguistic Geometry: From Search to Construction*. Kluwer Acad. Publishers (now Springer-Verlag), 416 pp., 2000.

[16] B. Stilman, Linguistic Geometry and Evolution of Intelligence, *ISAST Trans. on Computers and Intelligent Systems*, 2011 (in press).

[17] B. Stilman, V. Yakhnis, and O. Umanskiy, Winning Strategies for Robotic Wars: Defense Applications of Linguistic Geometry, *Artificial Life and Robotics*, Vol. 4, No. 3, 2000.

[18] B. Stilman, V. Yakhnis, and O. Umanskiy, Knowledge Acquisition and Strategy Generation with LG Wargaming Tools, *Int. J. of Comp. Intelligence and Applications*, Vol 2, No.4, Dec. 2002, pp. 385-409.

[19] B. Stilman, V. Yakhnis, and O. Umanskiy, Chapter 3.3. Strategies in Large Scale Problems, in [5], pp. 251-285, 2007.

[20] B. Stilman, V. Yakhnis, and O. Umanskiy, Linguistic Geometry: The Age of Maturity, *J. of Advanced Computational Intelligence and Intelligent Informatics*, Vol 14, No. 6, pp. 684-699, Sep. 2010.

[21] B. Stilman, V. Yakhnis, and O. Umanskiy, Revisiting History with Linguistic Geometry, *ISAST Trans. on Computers and Intelligent Systems*, Vol. 2, No. 2, pp. 22-38, Oct. 2010.

[22] B. Stilman, V. Yakhnis, and O. Umanskiy, The Primary Language of Ancient Battles, *Int. J. of Mach. Learning and Cyber.*, 2011 (in press).

[23] J. Von Neumann, *The Computer and the Brain*, Yale U. Press, 1958.