International Relations and Diplomacy, July-Aug. 2022, Vol. 10, No. 4, 141-157
doi: 10.17265/2328-2134/2022.04.001

Distributed Control Technology for Air and Missile Defence Operations

Peter Simon Sapaty
National Academy of Sciences of Ukraine, Kyiv, Ukraine

This paper relates to accepted presentation at international conference *Air and Missile Defence Technology*, November 16-17, 2022, London UK (day two), reflecting contents of the presentation slides. It describes applications of the patented and internationally tested Spatial Grasp Technology (SGT) and its Spatial Grasp Language (SGL) for Integrated Air and Missile Defense (IAMD). Based on holistic space navigation and processing by recursive mobile code self-spreading in distributed words, SGT differs radically from traditional management of large systems as consisting of parts exchanging messages. The dynamic network of SGL interpreters can be arbitrarily large and cover terrestrial and celestial environments as powerful spatial engines. The paper contains an example of tracking and destruction of multiple cruise missiles by self-evolving spatial intelligence in SGL using networks of radar stations. It also briefs the growing multiple satellite constellation in Low Earth Orbits (LEO) for potential IAMD applications. Starting from Strategic Defense Initiative (SDI) of the past and then briefing the latest project of Space Development Agency, the paper shows SGL solutions for discovery, tracking, and destroying ballistic missiles and hypersonic gliders with the use of collectively behaving constellations of LEO satellites. It also shows how to organize higher levels of supervision of groups of mobile chasers fighting multiple targets (both potentially as missiles or drones), by providing their global awareness even consciousness in SGL which can drastically improve their performance. The latest version of SGT can be implemented on any platforms and put into operation in a short time, similarly to its previous versions in different countries.

Keywords: air and missile defense, radar networks, satellite constellations, Strategic Defense Initiative (SDI), Space Development Agency, hypersonic gliders, Spatial Grasp Technology (SGT), Spatial Grasp Language (SGL), global awareness

1. Introduction

This paper relates to air and missile defense systems (Karako & Rumbaugh, 2017; Van Hooft & Boswinkel, 2021; NATO, 2022; Rozman, 2020), and especially how to organize them for effective protection of population against the growing threats which can come from both air and space. Missile defense is considered as a system, weapon, or technology involved in the detection, tracking, interception, and destruction of attacking missiles. Conceived as a defense against nuclear-armed Intercontinental Ballistic Missiles (ICBMs), its application has broadened to include shorter-ranged non-nuclear tactical and theater missiles, cruise missiles,

Peter Simon Sapaty, PhD, Chief Research Scientist, Institute of Mathematical Machines and Systems, National Academy of Sciences of Ukraine, Kyiv, Ukraine.

manned and unmanned aircraft, and other weapon systems. This adds urgency to the need for better and more Integrated Air and Missile Defense (IAMD). The transparency of the battlefield is also increasing due to improved space-based sensors and sensors based on unmanned systems, and the technologies for more precision, speed, and integration of air and missile weapons systems are becoming of high value. The joint IAMD efforts might be well served by new operational concepts, where networked integration is of particular importance as distributed defense envisions a more flexible air and missile defense.

The paper investigates suitability of the developed Spatial Grasp Technology (SGT) (Sapaty, 1986; 1993; 1999; 2005; 2015; 2016; 2017; 2018; 2019; 2021; 2022a; 2022b; 2023) for management of large distributed air and missile defense systems which may cover any national and international areas, the whole world including. Based on completely different ideology and mechanisms in comparison with conventional models and technologies, it allows us to obtain highly integral and extremely compact solutions in both terrestrial and celestial environments. Its applications already included such areas as intelligent network management, industry, social systems, collective robotics, military command and control, crisis management, national and international security, defense, distributed simulation, physical-virtual symbiosis, space-based systems, and even biology, psychology, and art.

The rest of the paper is organized as follows. Section 2 describes representation of traditional systems and solutions in them as consisting of parts exchanging messages, with growing integrity and manageability problems when they are becoming large and distributed. Section 3 explains main principles of the developed Spatial Grasp Technology (SGT) which provides holistic world coverage with parallel active code, also giving details of the basic Spatial Grasp Language (SGL) and its distributed interpretation. Section 4 provides an example in SGL for finding shortest path tree and shortest paths in distributed networks, which is extremely simple and compact in comparison with traditional solutions. Section 5 describes an example of tracking cruise missiles with complex routes by distributed networks of radar stations under SGT, the latter providing mobile intelligence effectively following the missiles wherever they go. Section 6 mentions rapidly growing satellite constellations which can be used for IAMD purposes. Section 7 briefs the Strategic Defense Initiative (SDI) of the eighties with key component called Brilliant Pebbles designed to destroy ballistic missiles, while expressing the pebbles functionality in SGL. Section 8 briefs the Next-Generation Space Architecture launched by Space Development Agency and provides mobile spatial solutions in SGL for discovery, tracking, and destruction of hypersonic gliders with the use of cooperating constellations of low-orbit satellites. Section 9 shows how to organize in SGL global awareness for a distributed team of chasers fighting many targets (where both can be missiles or drones), drastically improving their performance. Section 10 concludes the paper by acknowledging suitability of the developed model and technology for IAMD applications.

2. Representations of Traditional Systems and Solutions in Them

A system is traditionally considered as a group of interacting or interrelated elements that act according to a set of rules to form a unified whole ("System", n.d.; "Distributed Systems", n.d.; "Distributed Control System", n.d.; Buyya, Vecchiola, & Selvi, 2013; "List of Concurrent and Parallel Programming Languages", n.d.). Its organization may be hierarchical, distributed, or combined, as in Figure 1.

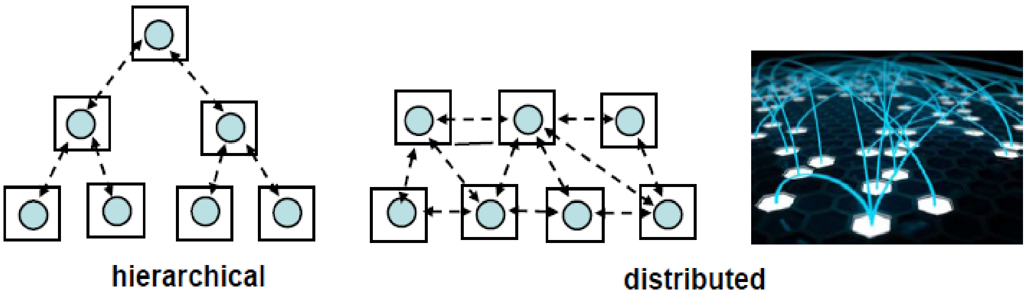

Figure 1. Traditional systems representation.

Distributed system is a system in which components located on networked computers communicate and coordinate their actions by passing messages. A Distributed Control System (DCS) is a computerized control system with many control loops, in which autonomous controllers are distributed throughout the system, but there is no central operator supervisory control. Main problems with traditional system representations and organizations, especially when they are large and distributed, are: *insufficient integrity and security*, also *difficulties with global goal orientation and overall control.*

3. Spatial Grasp Technology (SGT)

General Description

Within Spatial Grasp Technology (SGT) (Sapaty, 1993; 1999; 2005; 2017; 2018; 2019; 2021; 2022a; 2023), a high-level operational scenario is represented as an *active self-evolving pattern* rather than traditional program. This pattern written in a recursive Spatial Grasp Language (SGL) can start from any point (or points) of the world. It then *propagates, replicates, modifies, covers and matches* the distributed world in a parallel wavelike mode, while echoing the reached control states and data for higher levels decisions and further space navigation, as symbolically shown in Figure 2. SGT provides *holistic super-summative solutions, high integrity and flexibility.*

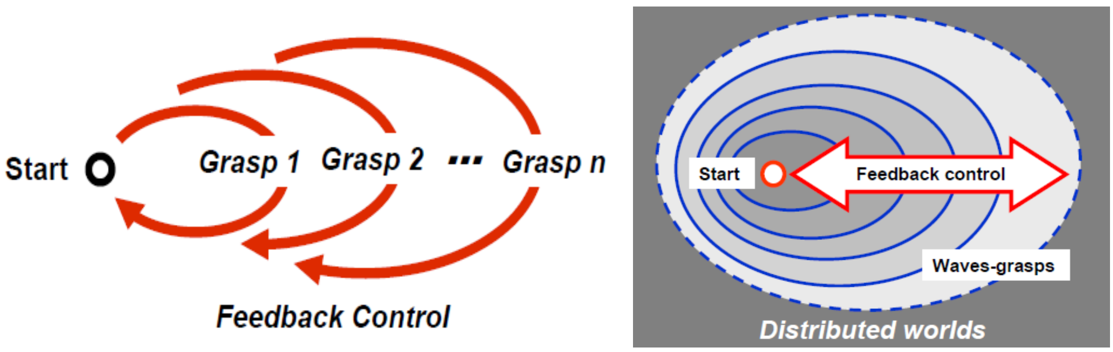

Figure 2. Parallel wavelike world coverage and conquest under SGT.

SGT allows for direct operation with different world representations: *Physical World (PW)*, considered as continuous and infinite, where each point can be identified and accessed by physical coordinates; *Virtual World (VW)* which is discrete and consists of nodes and semantic links between them; and *Executive World (EW)* consisting of active "doers" with communication possibilities between them. Different combinations of these worlds can also be possible within the same formalism, like *VPW, VEW, EPW, VPEW*.

Spatial Grasp Language (SGL)

The Spatial Grasp Language, with details in Sapaty (1986; 2015; 2016; 2022b), allows for expressing *direct space operations*, unlimited parallelism, high code clarity and compactness, as well as unlimited mobility. The SGL universal recursive organization with operational scenarios called *grasp* can be expressed by just a single string, in a formula-like mode, as follows.

grasp→constant|variable|rule ({*grasp*,})

The SGL *rule* expresses certain action, control, description or context accompanied with operands, which can themselves be *any grasp*. More SGL details are as follows.

constant→information|matter|custom|special
variable→global|heritable|frontal|nodal|environmental
rule →type|usage|movement|creation|echoing|
verification|assignment|advancement|branching|
transference|exchange|timing|qualifying

SGL Distributed Interpretation

The SGL interpreter, with details in Sapaty (1993; 1999; 2005; 2017; 2018; 2019; 2021; 2022a; 2023), see also Figure 3, consists of a number of specialized functional processors (shown by rectangles) which are working with specific data structures (depicted as ovals).

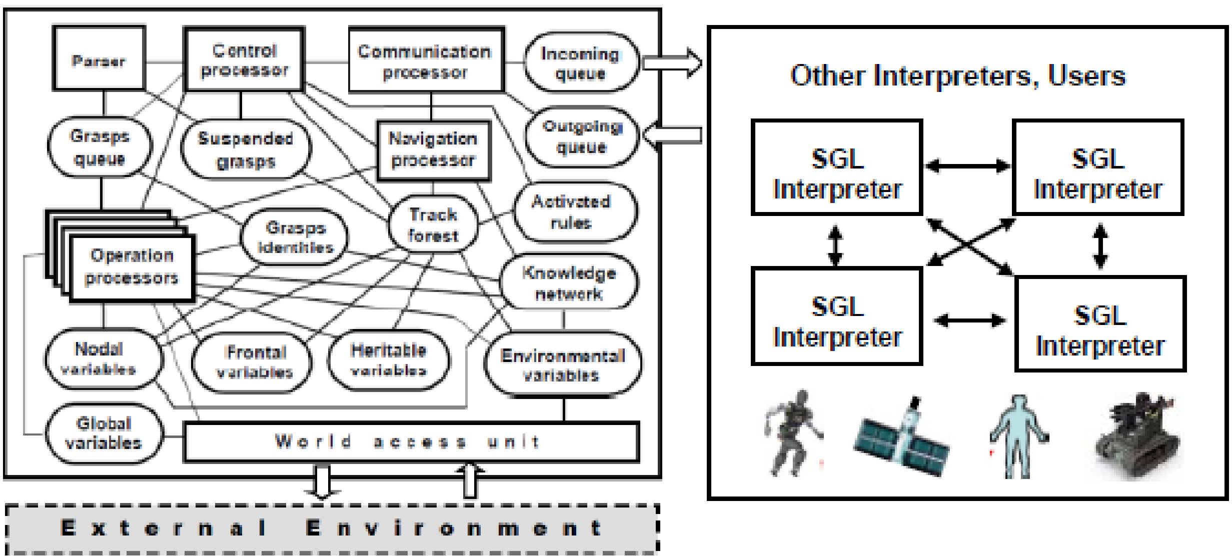

Figure 3. SGL interpreter architecture.

Each interpreter can support and process *multiple SGL scenario code* which happens to be in its responsibility at different moments of time. Integrated with any distributed systems, the interpretation network allows us to form a *spatial world computer with practically unlimited power* for simulation and management of the whole universe. Such collective engines *can simultaneously execute many cooperative or competitive tasks* without any central resources or control, as in Figure 4.

Communicating interpreters of SGL can be *in arbitrary number of copies*, say, up to millions and billions, and their dynamic networks can represent powerful spatial engines capable of solving *any problems in terrestrial and celestial environments*, as symbolically shown in Figure 5.

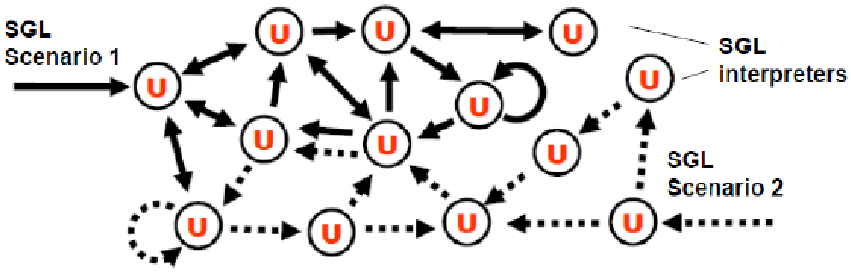

Figure 4. Distributed parallel world processing under SGT.

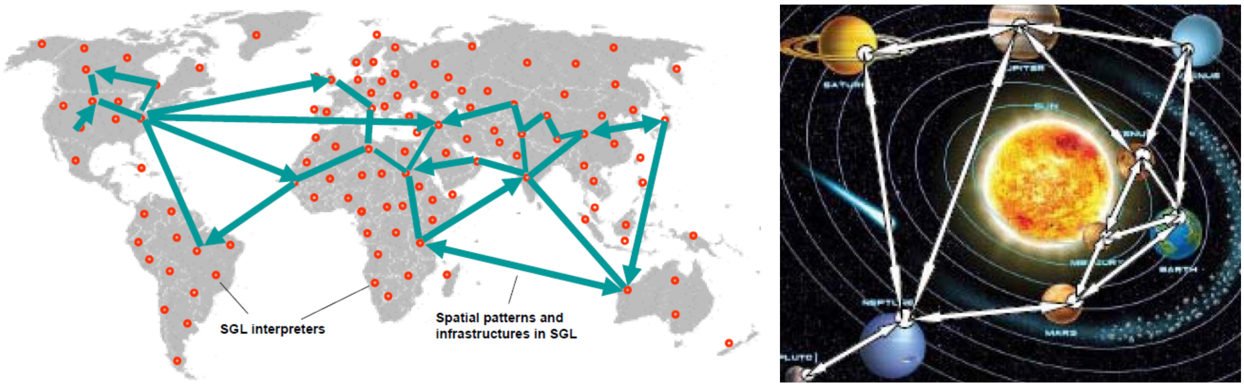

Figure 5. Conquering terrestrial and celestial environments under SGT.

4. Example in SGL: Finding Shortest Path Tree and Shortest Paths

Creation of the Shortest Path Tree (SPT) from a node in a network, covering the whole network, can be expressed in SGL with *enormous simplicity and compactness* in comparison with traditional solutions ("Dijkstra's Algorithm in C++|Shortest Path Algorithm", 2021), which is also discussed in Sapaty (2023). This is due to the fully distributed and parallel method used, where active SGL code *self-spreads through the body of a distributed network*. It automatically brings into its nodes the growing shortest distance from the start, with redefining this distance in nodes if a better solution appears possible after reaching these nodes again. An example of such SGL solution for a network of Figure 6 starting from Node 1 is shown below.

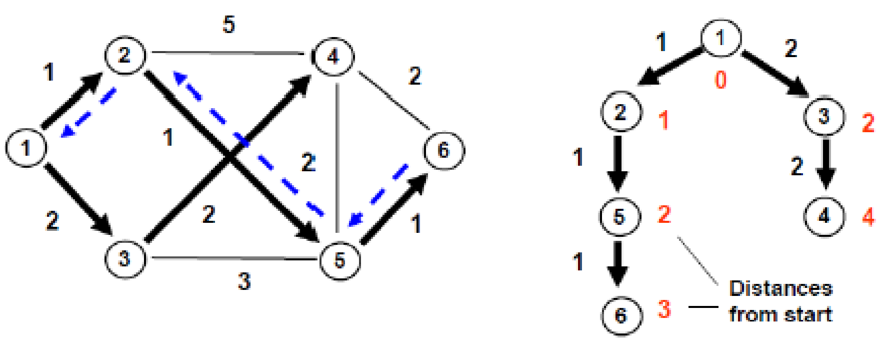

Figure 6. Shortest Path Tree finding in SGL.

```
frontal (Far); hop (1); Distance = 0;
repeat (hop_links (all); Far += LINK;
        or (Distance == nil, Distance > Far);
```

Distance = Far; Before = PREVIOUS)

Having received the SPT solution embedded into the distributed network body, we can easily output the shortest path from the starting node to any other network node, as shown below to Node 6, with the path issued in the opposite order as: 6, 5, 2, 1. With slight changes, the path can also be collected in the straight order, i.e. from Node 1 to Node 6.

frontal (Path);

hop (6); repeat (append (Path, Name); hop (Before));

output (Path)

5. Tracking Cruise Missiles by Networks of Radar Stations Under SGT

Distributed communicating radar stations operating under SGT can catch and follow moving objects (like cruise missiles (Xian, He, Liu, & Lei, 2012; Osborn, 2016; Zinger & Krill, 1997; UTC by Defense Industry Daily, 2017), see also Figure 7, throughout the whole region despite limitations of individual sensors at radar stations.

Figure 7. Cruise missiles and their trajectories.

The radar first seeing a new object becomes the start of a distributed tracing operation, with object's visibility subsequently shifting to other sensors after being lost by the current one. Object's behavior history can be collected by SGT-produced mobile intelligence following its physical movement via the radar network. Depending on the collected history, such object may be ordered to be destroyed. Any number of mobile objects simultaneously propagating through the same area can be served by SGT, as in Figure 8.

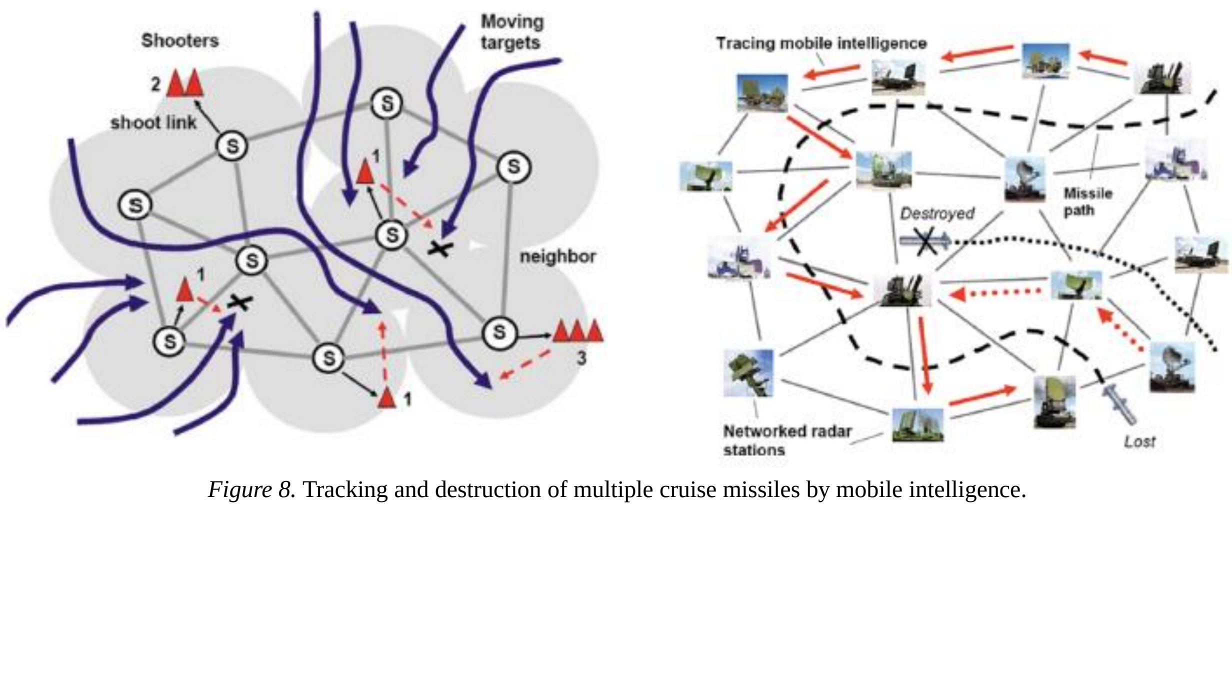

Figure 8. Tracking and destruction of multiple cruise missiles by mobile intelligence.

Highly parallel and fully distributed SGL scenario for tracing and destroying multiple cruise missiles with very complex and unpredictable routes can be hundred times simpler and more compact than by other models and languages, as follows.

```
hop (all_nodes);
frontal (Object, History, Threshold = …);
whirl (
  Object = search (aerial, new); visibility (Object) ≥ Threshold;
  free_repeat (
   loop (
     visibility (Object) ≥ Threshold); update (History, Object);
     if (dangerous (History), blind_destroy (Object)));
   max_destination (hop (all_neighbors); visibility (Object))))
```

6. Growing Satellite Constellations

The rapidly growing satellite constellations (Skibba, 2020; The United Nations Office for Outer Space Affairs, n.d.; Dredge, von Arx, & Timmins, 2017; Curzi, Modenini, & Tortora, 2020, see also Figure 9), especially in Lower Earth Orbits (LEO), can be effectively used for advanced AMD operations oriented on discovery, tracing, and elimination of high speed supersonic and hypersonic objects with very complex, tricky, and unpredictable routes.

Figure 9. Growing satellite constellations.

Near-Earth space is also becoming increasingly privatized, with the number of LEO satellites predicted to grow dramatically from about 2,000 to over 100,000 due to the launch of planned satellite constellations. They are often even called mega-constellations by their expected size. The use of multiple satellites for AMD purposes is considered in the following sections, starting from the Strategic Defense Initiative of the eighties and then describing their use by the latest Space Development Agency project.

7. Strategic Defense Initiative

The Strategic Defense Initiative (SDI) was a long-term technology research program developed to examine the feasibility of developing defenses against a ballistic missile attack (The White House, 1984; Hildreth, 1987; Gattuso, 1990; “Brilliant Pebbles”, n.d.), see also Figure 10.

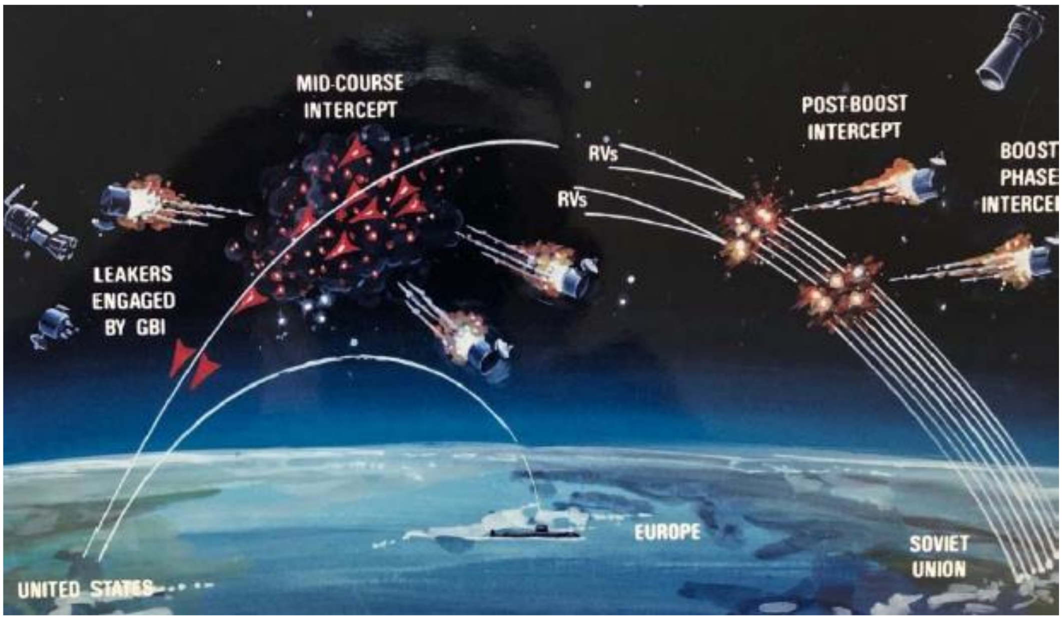

Figure 10. Strategic Defense Initiative architecture.

The SDI program was officially launched in 1984, with its key component called Brilliant Pebbles, as a proposed space-based weapon for the Global Protection Against Limited Strikes (GPALS). Brilliant Pebbles were designed to destroy ballistic missiles when they were most vulnerable, i.e. at first stages of their flight while still keeping many warheads to be released only later in the flight.

Discovery & Destruction of Targets by Pebbles in SGL

Extended operation of pebbles for “boost”, “post-boost”, and possibly even “mid-course” stages of flight can be organized in SGL as below, see also Figure 11. Pebbles will be making regular updates of the missile coordinates and trying to reduce distance to them, each time at least on *Limit 1* value. If pebbles appear closer to the missile on *Limit 2*, they will be trying to attack and finally destroy the missile in a collision impact.

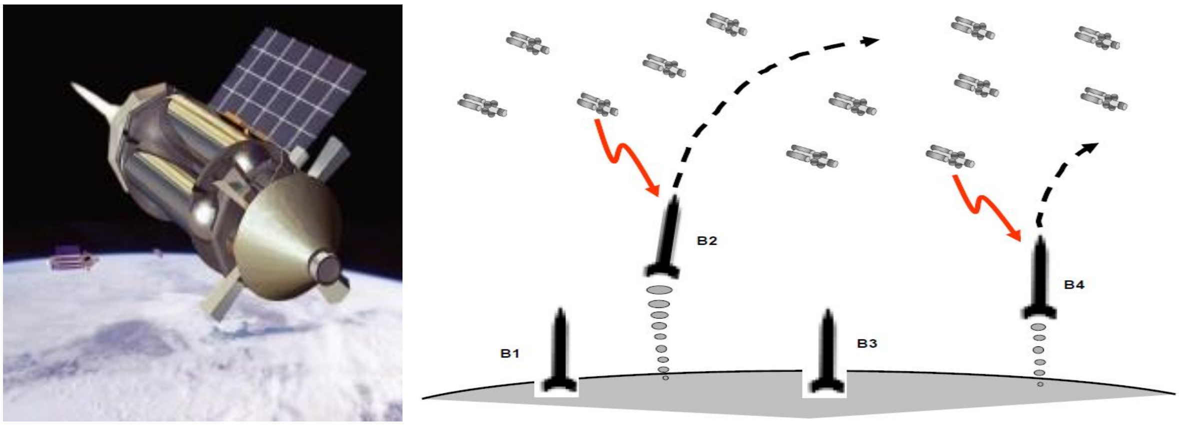

Figure 11. Fighting multiple missiles by brilliant pebbles.

```
hop (all_pebbles); Limit1 = …; Limit2 = …;
repeat (
   Target = search (missile_launch);
```

```
if (nonempty (Target),
   repeat (update (Target); reduce (WHERE, Target, Limit1);
            if (close (WHERE, Target, Limit2), engage (Target))));
sleep (delay))
```

8. Next-Generation Space Architecture

This Notional Architecture has been recently launched by Space Development Agency (Air & Space Forces, 2019; Magnuson, 2019; Messier, 2019; SDA, n.d.a; n.d.b), see Figure 12.

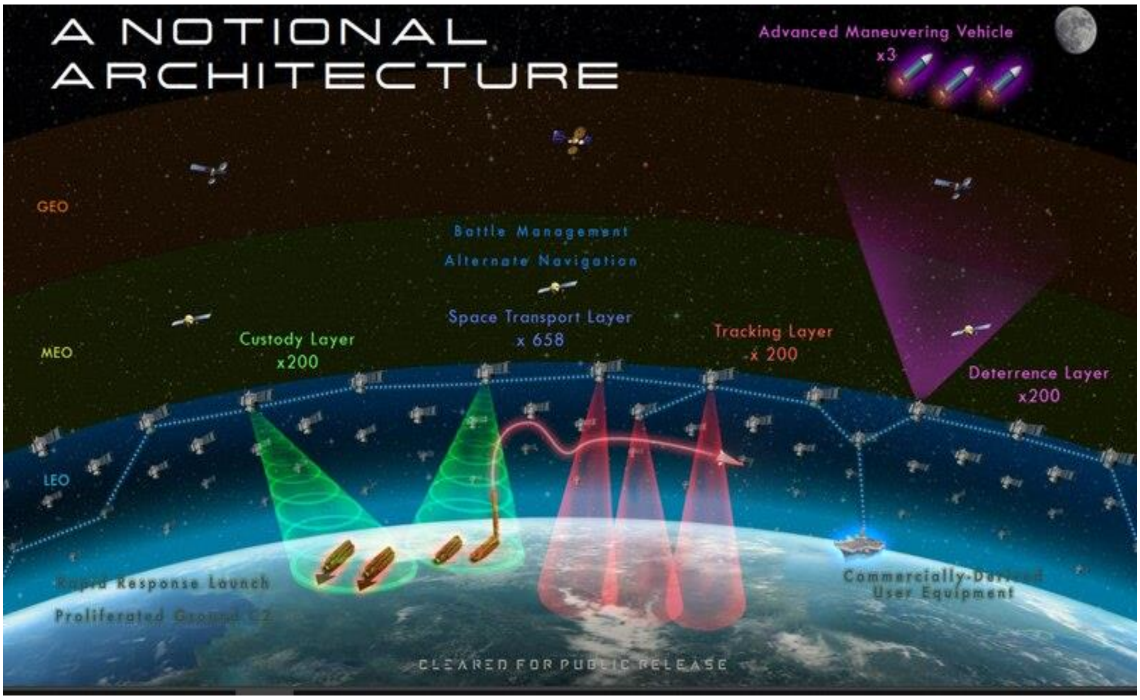

Figure 12. Space Development Agency Notional Architecture.

This Space Architecture plans to fight growing space-based threats, to move quickly on hypersonic defense and track hypersonic threats from space, also arm satellites with lasers to shoot down missiles, and so on. Unlike the SDI project, this architecture is oriented on intensive cooperation and collective behavior of many satellites. We will be considering implementation of some of its planned functionality by control scenarios written in SGL.

Single-Threaded Tracking of Hypersonic Gliders

The moment when the satellite sensor sees a new object for the first time (within a given visibility threshold) is *the start* of a distributed tracing operation, and the object is continually monitored by this satellite until its visibility remains acceptable. Otherwise, the monitoring *shifts to the neighbor* which has *the best vision* of the object after analyzing its visibility in all neighbors, as in Figure 13.

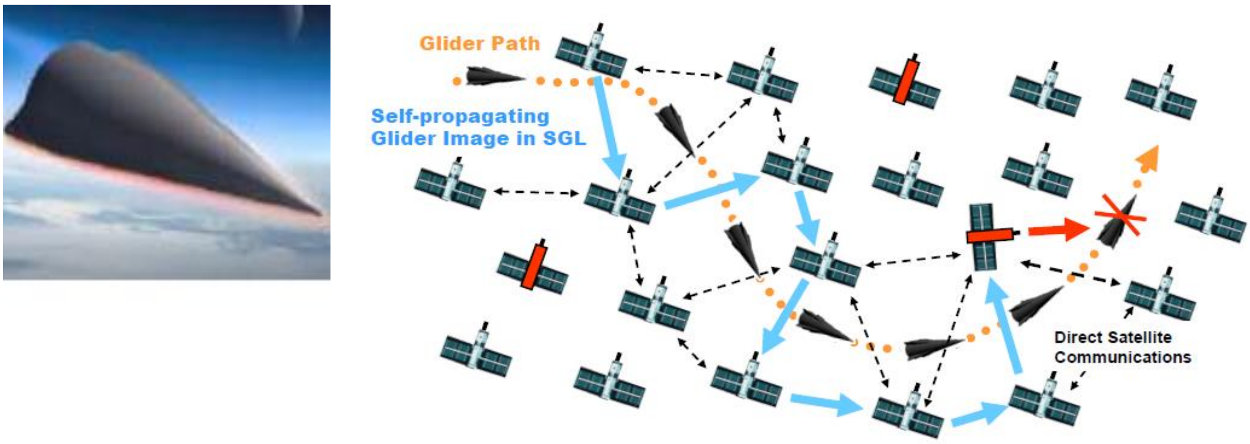

Figure 13. Single-threaded tracking and destruction of hypersonic gliders by SGT.

The history of the object movement and behavior can be collected and updated at each passed satellite by SGT mobile intelligence individually assigned to this object and accompanying its physical movement via satellite network, with *eventual destruction* of the object. The related SGL scenario will be as follows.

```
frontal (Depth = ..., Type = …, Object, History, Threshold1 = …, Threshold2 = …,
All = {stay, (hopfirst (Depth, Type); run (All))};
hopfirst (any, Type); run (All);
whirl (
  Object = search (glider, new); visibility (Object) ≥ Threshold1;
  repeat (
   loop (visibility (Object) ≥ Threshold1; update (History, Object);
        if (and (TYPE = destroyer, danger (History), distance (Object, WHERE) < Threshold2),
          pursue_destroy (Object)));
   max_destination (hop (Depth, Type); visibility (Object))))
```

Multithreaded Parallel Tracking

In highly dynamic satellite network topologies we can also allow *multithread tracing*, where next tracing stages may take place *from all neighbors with sufficient vision* of the object, as in Figure 14.

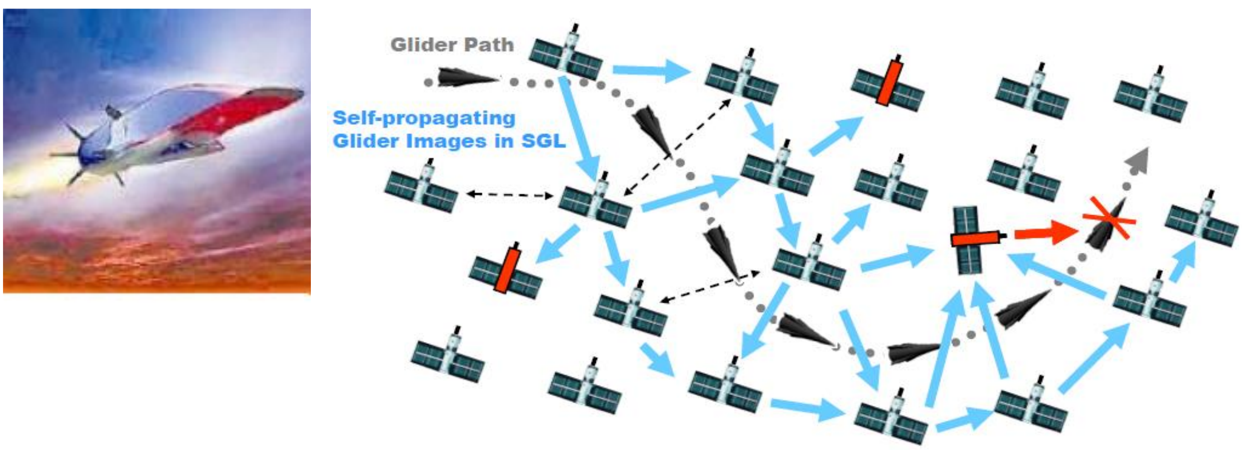

Figure 14. Multithreaded tracking and destruction of hypersonic gliders by SGT.

And even if the object is not currently visible from any neighbor after being lost by the current satellite, the further flexibility can be introduced where the tracing activity is transferred to *some or all neighbors with any visibility*, in hope to eventually reach satellites seeing this object, as follows in SGL.

frontal (Depth = ..., Type = …, Object, History, Threshold1 = …, Threshold2 = …,
All = {stay, (hopfirst (Depth, Type); run (All))};
hopfirst (any, Type); run (All);
whirl (
  Object = search (glider, new); visibility (Object) ≥ Threshold1;
  repeat (
   loop (visibility (Object) ≥ Threshold1); update (History, Object);
     if (and (Type = destroyer, danger (History), distance (Object, WHERE) < Threshold2),
       pursue_destroy (Object)));
   or_sequence ((hop (Depth, Type); visibility (Object) ≥ Threshold1),
         hopforth (Depth, Type))))

Managing Custody Observation by a Mobile Scenario in SGL

Watching earth-based custody nodes (in which missiles can originate) from space will need rapid changing their observing satellites, as each quickly moves in space, see Figure 15.

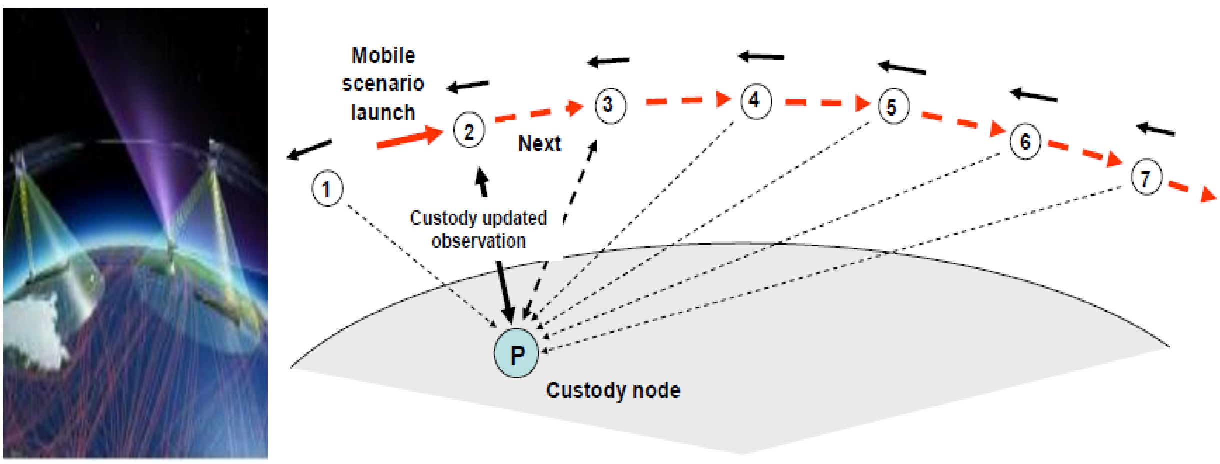

Figure 15. Management of custody nodes by satellite constellation under SGT.

The predominantly stationary custody nodes may be symbolically considered as moving through the dynamic satellite network themselves, with an example of SGL code following.

frontal (Custody = *X_Y*, History, Depth = ..., Type = …, Threshold = …);
hopfirst (any, Type);
repeat (Distance (WHERE, Custody) > Threshold; hopfirst (Depth, Type));
repeat (
 if (distance (WHERE, Custody) ≤ Threshold,
   (update (History, observe (Custody)); output (History)),
   min_destination (hop (Depth, Type); distance (WHERE, Custody))))

Integrating Custody and Tracking Functionality

After fixing a launch at the custody object, the glider-tracing mobile intelligence will be activated in SGL, which will accompany this glider wherever it goes via the satellite network, as in Figure 16.

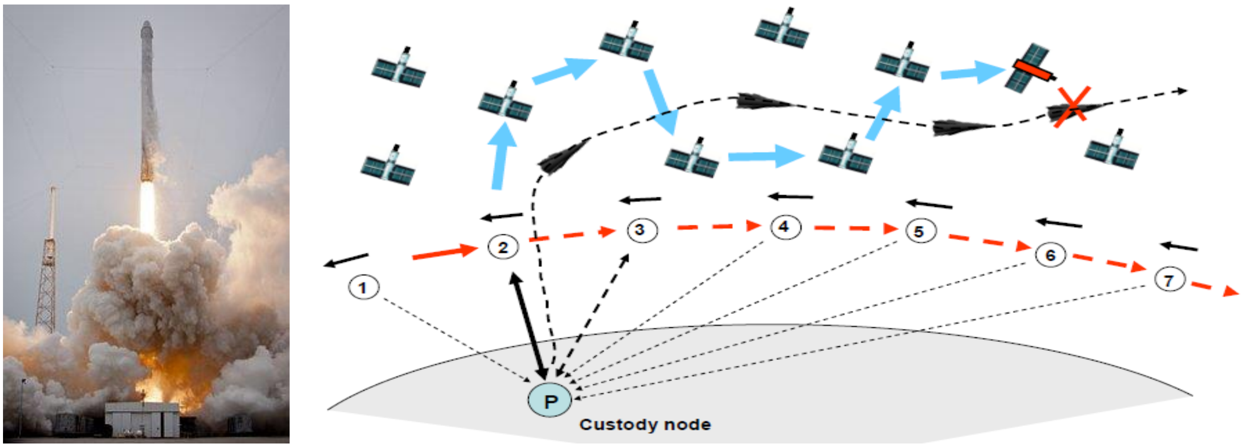

Figure 16. Integrated management of custody node with tracking discovered hypersonic gliders.

This integrated SGL scenario will be working repeatedly and endlessly, and if a new glider launch is detected in the observed custody, another tracking intelligence is associated with this object and will follow it via the satellite networks. At the same time, the continuing custody observation is updated too.

```
frontal (Custody = X_Y, History, Threshold1 = …, Threshold2 = …, Threshold3 = …);
min_destination (hop (all); distance (WHERE, Custody));
repeat (
  if (distance (WHERE, Custody) ≤ Threshold1,
      (update (History, observe (Custody);
      if (belong (glider_launch, History),
          free_repeat (
              loop (if (visibility (glider) ≥ Threshold2,
                            if (and (Type = destroyer, distance (Object, WHERE) < Threshold3),
                                pursue_destroy (Glider))));
              max_destination (hop (neighbors); visibility (glider))))),
      min_destination (hop (neighbors); distance (WHERE, Custody))))
```

9. Providing Global Awareness and Consciousness for Distributed Systems

We will provide here an example in SGL of a swarm of "chasers" that are constantly moving, discovering, and eliminating distributed targets seen (where both chasers and targets can potentially represent missiles, drones, or any other units). Will also show how to enrich the chasers swarm with a sort of global awareness (possibly, even consciousness (Chella & Manzotti, 2011; Galland & Grønning, 2019; Massimini, 2016; Nelson & Bancel, 2016; Moran, 2015; Sapaty, 2020; 2021b, Figure 17) over the whole operational area, which would allow individual chasers and the whole swarm to drastically improve performance.

Figure 17. Fantastic imagination of global awareness and consciousness.

This global awareness may be naturally and deeply embedded into the communicating chasers as part of their regular functionality. It can also be organized in SGL as an additional, superior, level which can constantly migrate and oversee the swarm body and surrounding area, regularly supplying this global vision to individual units.

Supplying the Swarm of Chasers With Deeply Embedded Global Awareness

By enriching the swarm with a sort of global awareness over the operational area, we may essentially improve its performance locally and as a whole. This global awareness quality may be naturally embedded into the communicating chasers, where targets seen by individual chasers are regularly exchanged with their neighbors, enriching their awareness, and these neighbors, of all they know and see, exchange with their neighbors too, and so on, see Figure 18.

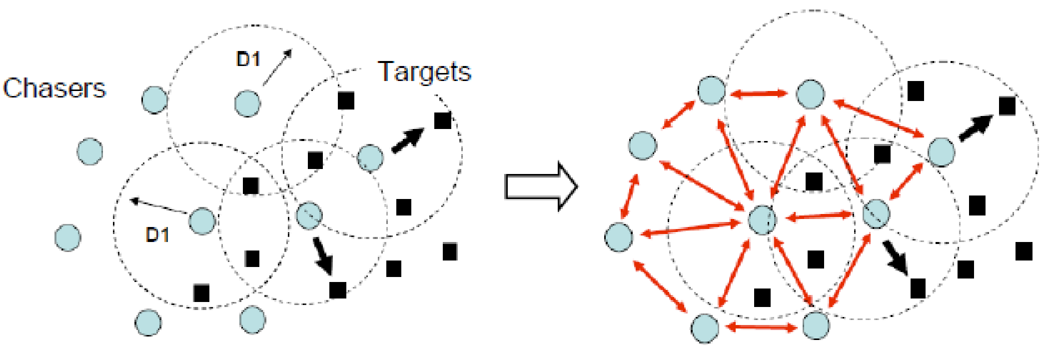

Figure 18. Deeply embedded global awareness.

This makes all swarm members gradually become aware of all targets in the region, despite not all of them visible individually, and always organize their movement in proper direction, first to be closer to the targets, and then to destroy them. An example of such SGL solution may be as follows.

hop_chasers (all);
nodal (D1 = *distance*, Targets); frontal (Exchange);
repeat (
  extend (Targets = search (D1);
  select_move_destroy_remove (Targets);
  stay (Exchange = Targets; hop (neighbors);

```
        merge (Targets, Exchange));
  sleep (Delay))
```

With Superior, Migrating, Global Awareness and Consciousness

We may also provide a higher-level *awareness operating autonomously and independently* over the basic swarm organization. Initially applied in any swam node, it will be capable to contact all nodes in parallel, collect all they see, and then distribute to all nodes this global vision, in parallel too. The focus of such superior consciousness may be constantly migrating between the chasers for security reasons, as in Figure 19 and SGL solution that follows.

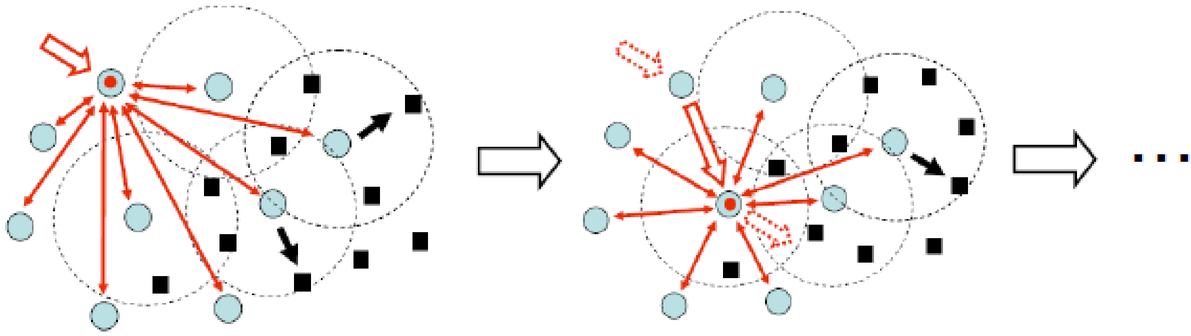

Figure 19. Separated migrating superior awareness and consciousness.

```
nodal (D1 = distance, Targets); frontal (Global);
parallel (
 (hop_chasers (all);
  repeat (
  extend (Targets) = search (D1);
  select_move_destroy_remove (Targets))),
 (hop_chasers (any);
  repeat (
   Global = merge (hop_hhasers (all); Targets);
   stay (hop_nodes (all); merge (Targets, Global));
   move (any_neighbor))))
```

Also, for higher security and survivability, where each chaser unit can be potentially damaged or destroyed any time on a battlefield, we can introduce and operate *many such migrating consciousness centers* in parallel. This will always guarantee capability of this superior awareness and consciousness with any number of remaining chasers.

10. Conclusions

The following results of the conducted and described investigations can be named:

- It is possible to organize *effective, intelligent, and universal solutions* for fighting different types of hostile objects, including cruise missiles, hypersonic gliders, and military drones, under the *parallel and distributed model and technology* developed.
- The SGT is effectively *matching* the current and future plans of the space conquest, especially based on numerous *cooperating satellites* in predominantly LEO orbits.

• The main problem with LEO satellites is that they are *rapidly changing their positions* over Earth locations, and to provide continuous observation they regularly need to transfer duties and accumulated information to other satellites.

• That can be effectively solved by *super-virus* SGT and its basic SGL language, which *navigate in parallel multi-satellite constellations* and solve continuous custody observation and tracing hypersonic gliders by cooperating satellites.

• We also demonstrated how to organize in SGL a sort of *global vision and awareness* for mobile multi-component fighting systems in distributed environments, with each unit originally seeing only part of the operational area.

• Due to fully interpreted SGL language, it is possible to organize *active dialogue* with distributed missile defense systems at any stages, also *combine distributed interactive simulation with live control*, as was successfully tested in our previous defense-oriented projects and publications.

• The technology has *simple and effective implementation* on any platforms, which was prototyped and tested in different countries for the previous technology versions.

References

Air & Space Forces. (2019). Space development agency next-generation space architecture request for information SDA-SN-19-0001. Retrieved from https://www.airforcemag.com/PDF/DocumentFile/Documents/2019/SDA_Next_Generation_Space_Architecture_RFI%20(1).pdf

Brilliant Pebbles. (n.d.). Retrieved from https://en.wikipedia.org/wiki/Brilliant_Pebbles

Buyya, R., Vecchiola, C., & Selvi, T. (2013). Principles of parallel and distributed computing. In *Mastering cloud computing* (pp. 29-70). Oxford: Elsevier Inc. Retrieved from https://www.sciencedirect.com/topics/computer-science/distributed-programming

Chella, A., & Manzotti, R. (December 2011). Artificial consciousness. *ResearchGate.* Retrieved from https://www.researchgate.net/publication/225838750_Artificial_Consciousness

Curzi, G., Modenini, D., & Tortora, P. (2020). Large constellations of small satellites: A survey of near future challenges and missions. *Aerospace, 7*(9), 133. Retrieved from https://www.mdpi.com/2226-4310/7/9/133/htm

Dijkstra's Algorithm in C++|Shortest Path Algorithm. (May 20, 2021). Retrieved from https://favtutor.com/blogs/dijkstras-algorithm-cpp

Distributed control system. (n.d.). Retrieved from https://en.wikipedia.org/wiki/Distributed_control_system

Distributed systems. (n.d.). Retrieved from https://computersciencewiki.org/index.php/Distributed_systems

Dredge, K., von Arx, M., & Timmins, I. (September/October 2017). LEO constellations and tracking challenges. Retrieved from www.satellite-evolution.com

Galland, D., & Grønning, M. (April 2019). Spatial consciousness. *ResearchGate*. Retrieved from https://www.researchgate.net/publication/330753755_Spatial_consciousness

Gattuso, J. (1990). *Brilliant Pebbles: The revolutionary idea for strategic defense.* Retrieved from https://www.heritage.org/defense/report/brilliant-pebbles-the-revolutionary-idea-strategic-defense

Hildreth, S. A. (1987). The strategic defense initiative: Program facts. Retrieved from https://www.everycrsreport.com/files/19870722_IB85170_64d13e614c37eecbed39c00741ddfb269f814fef.pdf

Karako, T., & Rumbaugh, W. (December 29, 2017). *Distributed defense: New operational concepts for integrated air and missile defense.* Washington DC: Rowman & Littlefield. Retrieved from https://books.google.com.ua/books/about/Distributed_Defense.html?id=cRVEDwAAQBAJ&redir_esc=y

List of concurrent and parallel programming languages. (n.d.). Retrieved from https://en.wikipedia.org/wiki/List_of_concurrent_and_parallel_programming_languages

Magnuson, S. (2019). Web exclusive: Details of the Pentagon's new space architecture revealed. Retrieved from https://www.nationaldefensemagazine.org/articles/2019/9/19/details-of-the-pentagon-new-space-architecture-revealed

Massimini, M. (April 2016). The distribution of consciousness: A difficult Cartesian chart. *Research-Gate*. Retrieved from https://www.researchgate.net/publication/307808562_The_Distribution_of_ Consciousness_A_Difficult_Cartesian_Chart

Messier, D. (2019). Space development agency seeks next-gen architecture in first RFI. Retrieved from http://www.parabolicarc.com/2019/07/07/space-development-agency-issues-rfi/

Moran, M. (March 11, 2015). Network theory sheds new light on origins of consciousness. *Medical Xpress*. Retrieved from https://medicalxpress.com/news/2015-03-network-theory-consciousness.html

NATO. (January 27, 2022). NATO integrated air and missile defence. Retrieved from https://www.nato.int/cps/en/natohq/topics_8206.htm

Nelson, R., & Bancel, P. A. (October 2016). Exploring global consciousness. *ResearchGate*. Retrieved from https://www.researchgate.net/publication/268001767_Exploring_Global_Consciousness

Osborn, K. (September 14, 2016). F-35 intercepts cruise missile to defend ship during important test. *The National Interest*. Retrieved from https://nationalinterest.org/blog/the-buzz/f-35-intercepts-cruise-missile-defend-ship-during-important-17707

Rozman, J. (2020). Integrated air and missile defense in multi-domain operations, SPOTLIGHT 20-2. Retrieved from https://www.ausa.org/sites/default/files/publications/SL-20-2-Integrated-Air-and-Missile-Defense-in-Multi-Domain-Operations.pdf

Sapaty, P. (1986). A wave language for parallel processing of semantic networks. *Computers and Artificial Intelligence, 5*(4), 289-314.

Sapaty, P. (1993). A distributed processing system. *European Patent No. 0389655, Publ. 10.11.93*. European Patent Office.

Sapaty, P. (1999). *Mobile processing in distributed and open environments.* New York: John Wiley & Sons.

Sapaty, P. (2005). *Ruling distributed dynamic worlds.* New York: John Wiley & Sons.

Sapaty, P. (2015). A brief introduction to the Spatial Grasp Language (SGL). *Journal of Computer Science & Systems Biology, 9*(2), 76-92.

Sapaty, P. (2016). Spatial Grasp Language (SGL). *Advances in Image and Video Processing, 4*(1).

Sapaty, P. (2017). *Managing distributed dynamic systems with spatial grasp technology*. New York: Springer.

Sapaty, P. (2018). *Holistic analysis and management of distributed social systems*. New York: Springer.

Sapaty, P. (2019). *Complexity in international security: A holistic spatial approach*. Bingley: Emerald Publishing.

Sapaty, P. (2020). Symbiosis of real and simulated worlds under global awareness and consciousness. In *The Science of Consciousness (TSC) Conferences*. Tucson, Arizona. Retrieved from https://eagle.sbs.arizona.edu/sc/report_poster_detail.php?abs=3696

Sapaty, P. (2021a). *Symbiosis of real and simulated worlds under spatial grasp technology*. New York: Springer.

Sapaty, P. (March 2021b). Simulating distributed and global consciousness under SGT. In *Symbiosis of real and simulated worlds under spatial grasp technology* (pp. 157-170). Cham: Springer. Retrieved from https://link.springer.com/chapter/10.1007/978-3-030-68341-2_6

Sapaty, P. (2022a). *Spatial grasp as a model for space-based control and management systems*. London: CRC Press.

Sapaty, P. (2022b). A language to comprehend and manage distributed worlds. *Mathematical Machines and Systems, No. 3.*

Sapaty, P. (2023). *The spatial grasp model: Applications and investigations of distributed dynamic worlds.* Bingley: Emerald Points.

SDA. (n.d.a). Tracking. Retrieved from https://www.sda.mil/tracking/

SDA. (n.d.b). Custody. Retrieved from https://www.sda.mil/custody/

Skibba, R. (February 26, 2020). How satellite mega-constellations will change the way we use space. *MIT Technology Review*. Retrieved from https://www.technologyreview.com/2020/02/26/905733/satellite-mega-constellations-change-the-way-we-use-space-moon-mars/

System. (n.d.). Retrieved from https://en.wikipedia.org/wiki/System

The United Nations Office for Outer Space Affairs. (n.d.). United Nations register of objects launched into outer space. Retrieved from http://www.unoosa.org/oosa/en/spaceobjectregister/index.html

The White House. (1984). Strategic defense initiative. Retrieved from https://fas.org/irp/offdocs/nsdd/nsdd-119.pdf

UTC by Defense Industry Daily. (February 13, 2017). JLENS: Co-ordinating cruise missile defense—And more. Retrieved from https://www.defenseindustrydaily.com/jlens-coordinating-cruise-missile-defense-and-more-02921/

Van Hooft, P., & Boswinkel, L. (November 2021). Surviving the deadly skies, integrated air and missile defence 2021-2035. The Hague Centre for Strategic Studies. Retrieved from https://hcss.nl/wp-content/uploads/2021/12/Integrated-Air-and-Missile-Defense-HCSS-Dec-2021.pdf

Xian, Y., He, B., Liu, G., & Lei, G. (2012). Cruise missile route planning based on quantum immune clone algorithm. *J. Inf. Comput. Sci.*, *9*(8), 2097-2105.

Zinger, W. H., & Krill, J. A. (1997). Mountain top: Beyond-the-horizon cruise missile defense. *Johns Hopkins APL Technical Digest*, *18*(4), 501-520. Retrieved from https://www.jhuapl.edu/Content/techdigest/pdf/V18-N04/18-04-Zinger.pdf