https://orcid.org/0000-0002-4168-7190

UDC 623.764

P.S. SAPATY[*]

MANAGING MULTIPLE SATELLITE ARCHITECTURES BY SPATIAL GRASP TECHNOLOGY

[*]Institute of Mathematical Machines and Systems Problems National Academy of Sciences of Ukraine, Kyiv, Ukraine

Анотація. *У статті розглядається ряд передових космічних проєктів, орієнтованих на велику кількість супутників, що рухаються навколо нашої планети по низьких навколоземних орбітах, а також досліджуються способи їх взаємоорганізації для вирішення важливих світових проблем, насамперед тих, що стосуються питань глобальної безпеки та оборони. У роботі аналізуються шляхи застосування розробленої Технології просторового захоплення (ТПЗ), яка вже була успішно протестована на багатьох прикладних задачах, для моделювання та управління системами супутників. Особливий інтерес у цьому напрямі представляє останній проєкт Агенції космічного розвитку «Космічна архітектура наступного покоління», в межах якого одночасно планується використання великої кількісті взаємопов'язаних та організованих на різних рівнях супутників. Ця програма є набагато прогресивнішою за широко відому «Стратегічну оборонну ініціативу», представлену у 80-х роках минулого століття. ТПЗ базується на мобільних рекурсивних сценаріях, висловлених на спеціальній високорівневій Мові просторового захоплення (МПЗ), які здатні самостійно покривати розподілені середовища та утворювати в них потужні просторові інфраструктури, що можуть вирішувати будь-які розподілені проблеми. У даній статті викладаються основні деталі останньої версії ТПЗ, наводяться приклади вирішення за її допомогою таких проблем, як розподілене трасування з наступною ліквідацією крилатих ракет та гіперзвукових планерів, що рухаються за складними траєкторіями, формування ефективної системи нагляду не тільки за окремими небезпечними земними об'єктами, але й загалом за будь-якою розподіленою наземною інфраструктурою. Також у роботі показано, як за допомогою встановлення над супутниковими угрупуваннями спеціального віртуального рівня можливо суттєво спростити процеси виявлення та вирішення багатьох проблем як у наземному, так і в космічному середовищі, а також вдосконалити управління складними національними міжнародними операціями та компаніями з космосу.*

Ключові слова: *Стратегічна оборонна ініціатива, супутники «Brilliant Pebbles» («Блискучі камінці»), Космічна архітектура наступного покоління, гіперзвукові планери, система охорони, віртуальний рівень, Технологія просторового захоплення, Мова просторового захоплення, мобільні рекурсивні сценарії.*

Abstract. *The paper reviews some advanced space projects oriented on many satellites moving around the globe in low Earth orbits, and investigates how to organize their collective operation for solving important world problems, especially those related to global security and defense. It analyzes the application of the developed Spatial Grasp model and Technology (SGT), successfully tested on numerous applications, for simulation and management of multiple satellite architectures. Of particular interest is the latest Space Development Agency Next-Generation Space Architecture that uses a great number of cooperating satellites organized on different layers, which appears to be much more advanced than the known Strategic Defense Initiative project of the eighties. SGT is based on mobile recursive scenarios in a special high-level Spatial Grasp Language (SGL) which can self-navigate and self-match distributed environments while leaving throughout them powerful spatial infrastructures capable of solving any distributed problems. Providing basics of the latest SGT version, the paper describes examples of solutions in it of such problems as distributed tracing and elimination of complexly moving cruise missiles and hypersonic gliders, organization of effective custody layer which will be able to observe not only localized dangerous objects on the Earth but also any distributed terrestrial infrastructures as a whole. It also shows how to introduce a higher virtual layer for satellite constellation which may simplify formulation and solution of* *many problems in both terrestrial and celestial environments, including advanced command and control of complex national and international operations and campaigns from space.*

Keywords: *Strategic Defense Initiative, Brilliant Pebbles, Next-Generation Space Architecture, hypersonic gliders, custody layer, virtual layer, Spatial Grasp Technology, Spatial Grasp Language, mobile recursive scenarios.*

DOI: 10.34121/1028-9763-2021-1-3-16

1. Introduction

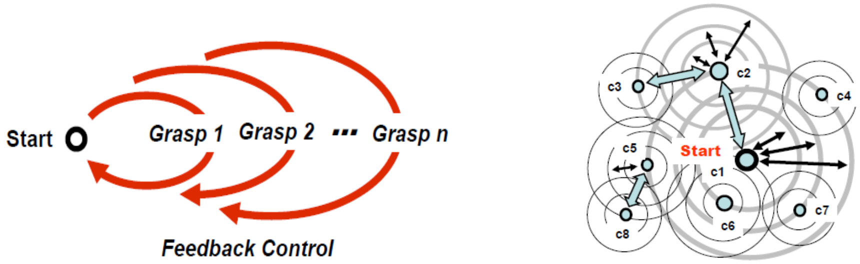
Humanity's interest in the heavens has been universal and enduring. Human space exploration helps us address fundamental questions about our place in the Universe and the history of our solar system. Through addressing the challenges related to human space exploration, we expand technology, create new industries, and foster peaceful connection between nations. Curiosity and exploration are vital to the human spirit and accepting the challenge of going deeper into space will invite the citizens of the world today and the generations of tomorrow to unite on this exciting journey. We may expect space solar power plants, industrial exploration of the Moon, eco-industry, recovery of natural resources, global weather management**,** large-scale artificial structures in space, the use of raw materials of other planets, colonization of Venice and Mars, etc., mentioning here already evolving space industry and robotics [1–4]. Other space projects relate to global defense, like Strategic Defense Initiative (SDI) with multiple minisatellites called brilliant pebbles [5–8] and New Space Architecture of the Space Development Agency (SDA) [9–12]. However, enormous problems appear in space around the Earth too, due to rapidly growing amount of space debris [13]. Briefed in this paper Spatial Grasp Technology (SGT) had previous versions developed before the SDI project and later influenced by it [14], with SDI presented at the conference held in Amsterdam in 1987. A technology version was patented [15] and later used in the defense-oriented Distributed Interactive Simulation (DIS) project [16–18], and tested on many other applications. The modern technology version has been published in many papers, six books including [19–26]. The current paper concentrates on the use of the latest SGT version and its basic high-level Spatial Grasp Language (SGL) [19–24] for simulation and management of different issues related to the SDA New Space Architecture.

The rest of the paper is organized as follows. Section 2 provides basics of SGT and SGL with its networked interpretation. Section 3 briefs two defense projects with multiple satellites in space such as Strategic Defense Initiative of the eighties, and the most recent SDA’s Next-Generation Space Architecture. Chapter 4 shows how self-spreading SGL scenarios can effectively track and destroy complexly moving objects in large distributed spaces, like cruise missiles and hypersonic gliders, using for this multiple cooperating terrestrial and celestial sensors. Section 5 shows how to use mobile SGL code to implement a custody layer of the new space architecture for constant monitoring of terrestrial objects by moving satellites in LEO orbits. It also shows how to organize a higher, or virtual, custody layer which can constantly monitor arbitrarily distributed infrastructures on the Earth in a holistic manner through the use of collectively behaving satellites. Section 6 provides elementary examples of how the introduced virtual layer can simplify solution of problems in both terrestrial and celestial environments, like organization of control and management of distributed campaigns on the Earth from space, analyzing the distance between remote terrestrial objects and discovering launches of hypersonic objects with their subsequent tracing and elimination. The concluding Section 7 reveals plans for continuing research in the described area which include supplying the space infrastructure with DARPA Mosaic Warfare features and already researched under SGT possibilities of global awareness and consciousness of distributed dynamic systems.

2. Spatial Grasp Technology Basics

2.1. General SGT Idea

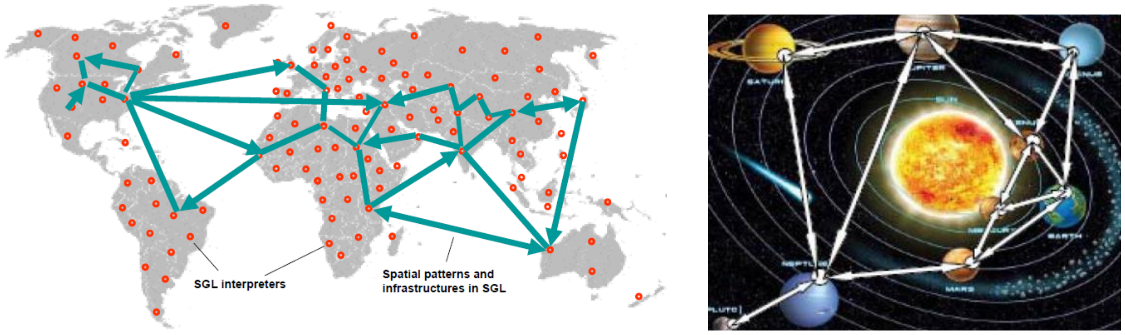
Within Spatial Grasp Technology (SGT), a high-level scenario for any task to be performed in a distributed world is represented as an active self-evolving pattern rather than a traditional program, sequential or parallel one. This pattern, written in a high-level Spatial Grasp Language (SGL) and expressing top semantics of the problem to be solved, can start from any point of the world. Then it spatially propagates, replicates, modifies, covers and matches the distributed world in a parallel wavelike mode, while echoing the reached control states and data found or obtained for making decisions at higher levels and further space navigation, as symbolically shown in Fig. 1 *a*, *b*.



a *b*

Figure 1 – Controlled wavelike coverage of distributed spaces (a) with generation of further waves from any points reached (b)

Many spatial processes in SGL can start any time and in any place, cooperating or competing with each other, depending on applications. The self-spreading and self-matching SGL patterns-scenarios can create active spatial infrastructures covering any region, as shown in Fig. 2 *a*, *b*.



a *b*

Figure 2 – Worldwide coverage with the creation of distributed infrastructures: from terrestrial (a) to celestial (b)

These infrastructures can effectively support or express distributed knowledge bases, advanced command and control, situation awareness, autonomous and collective decisions, as well as any existing or hypothetical computational and/or control models, systems and solutions.

2.2. Spatial Grasp Language

General SGL organization is as follows: syntactic categories are shown in italics, vertical bar separates alternatives, the parts in braces indicate zero or more repetitions with a delimiter at the right if multiple, and constructs in brackets are optional:

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

From this definition, an SGL scenario, called grasp, supposedly applied in some point of the distributed space, can just be a constant directly providing the result to be associated with this point. It can be a variable whose content, assigned to it previously when staying in this or (remotely) in other point of space (as variables may have non-local meaning and coverage), provides the result in the application point too. It can also be a rule (expressing certain action, control, description or context) optionally accompanied by operands separated by comma (if multiple) and embraced in parentheses. These operands can be of any nature and complexity (including arbitrary scenarios themselves) and defined recursively as grasp, i.e. can be constants, variables or any rules with operands (i.e. as grasps again), and so on.

Rules, starting in some point of the world, can organize navigation of the world sequentially, in parallel or any combinations thereof. They can result in staying in the same application point or can cause movement to other points of the world with the obtained results to be left there, as in the final points of the rule. Such results can also be collected, processed and returned to the starting point of the rule, the latter serving as the final one on this rule. The final points of the world reached after the rule invocation can become the starting ones for other rules by themselves. The rules, due to recursive language organization, can form arbitrary operational and control infrastructures expressing any sequential, parallel, hierarchical, centralized, localized, mixed and up to fully decentralized and distributed algorithms. These algorithms, called spatial, can effectively operate in, with, under, in between, over and instead of (as for simulation) large, dynamic and heterogeneous spaces, which can be physical, virtual, management, command and control, or combined. The top level of the SGL syntax is shown below and in Fig. 3 in more details.

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

constant → *information* | *matter* | *custom* | *special* | ***grasp***

variable → *global* | *heritable* | *frontal* | *nodal* | *environmental*

rule → *type* | *usage* | *movement* | *creation* | *echoing* |
verification | *assignment* | *advancement* | *branching* |
transference | *exchange* | *timing* | *qualifying* | ***grasp***

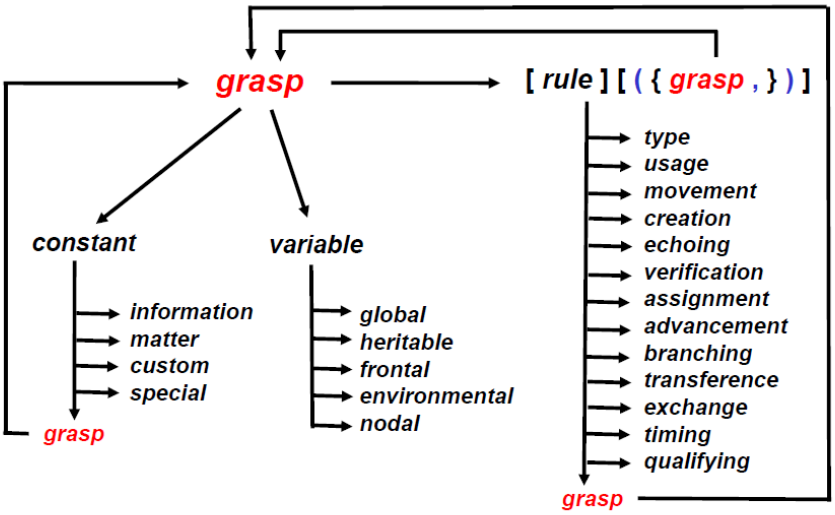


Figure 3 – SGL recursive syntax

2.3. Networked SGL Interpretation

The dynamic network of SGL interpreters covering distributed spaces may have any (including runtime changing) topology and can operate without any central facilities or control, exhibiting at the same time wholeness and high integrity as a system. The overall management of distributed evolution of high-level SGL scenarios is based on a space-covering track infrastructure supporting overall awareness, goal orientation, automatic C2, as well as properly handling various distributed information resources (including their creation, optimization and cleaning/removal).

The distributed execution of SGL scenarios can be effectively implemented in a variety of systems, whether technical or social, with any types of communications between their components (electronic, acoustic, visual, postal, even direct voice or paper writing). SGT can convert any collectives (human, robotic, mixed) into holistic systems operating under global goals and capable of acting in complex and unpredictable environments (this may even be effective for collective underwater operations with slow data transfer, due to highly compact operational scenarios in SGL).

The SGL interpreter main components and its networked organization are shown in Fig. 4.

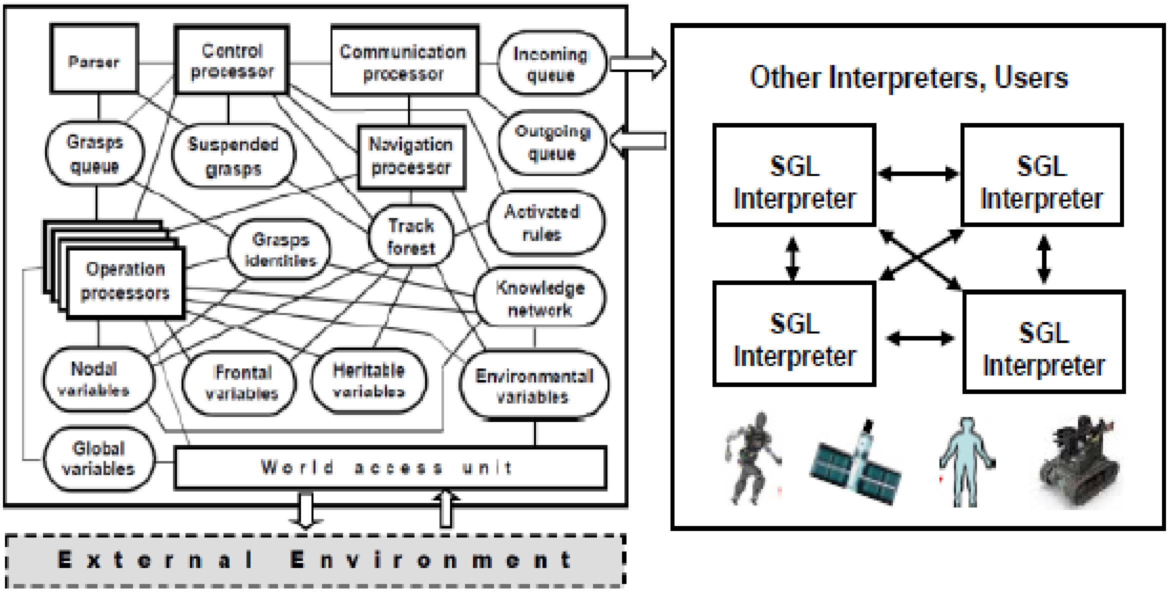


Figure 4 – Main components of SGL interpreter and its networked organization

The interpreter consists of a number of specialized functional processors (shown by the rectangles) working with and sharing specific data structures (shown by the ovals). Each interpreter can support and process a multiple SGL scenario code which appears to be in its responsibility at different moments of time. Implanted into any distributed systems and integrated with them, the interpretation network (having potentially from a million to several billions of communicating interpreter copies) allows us to form a spatial computer with practically unlimited power for simulation and management of the whole world.

3. Projects with Multiple Satellites in Space

3.1. 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 [5–8]. The SDI program was officially launched in 1984, with its key component called Brilliant Pebbles [7, 8], as a proposed space-based weapon for the Global Protection Against Limited Strikes (GPALS) Strategic Defense System, see Fig. 5.

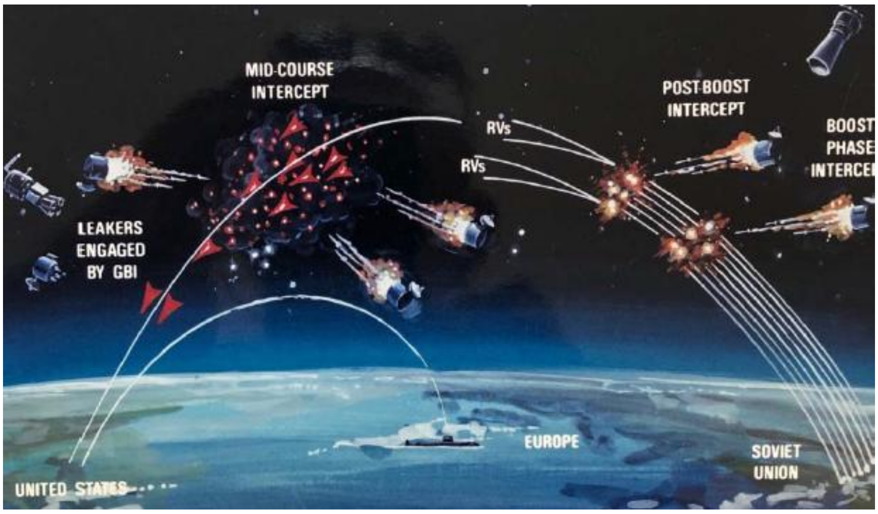


Figure 5 – Strategic Defense Initiative Architecture

It entailed hundreds of individual interceptors in orbit around the Earth at relatively evenly spaced intervals. Each interceptor could be linked by communications to the others and to ground stations. In the event of a ballistic missile attack, each could be given a high degree of autonomy to detect and intercept missiles that enter its battle space. A set of deployed Brilliant Pebbles would be made up of several staggered rings orbiting at about 400 kilometers above the Earth, with several Brilliant Pebbles in each ring. Once enabled by human command, Brilliant Pebbles could select their targets and divert from their orbits into the path of enemy missiles. The interceptors would carry no explosives, but the force of their high-speed collision is expected to destroy targets.

3.2. Next-Generation Space Architecture

The recently launched by Space Development Agency notional architecture [9–12] is made up of several layers (Fig. 6):

- A space transport layer is a global mesh network providing data and communications 24/7.
- A tracking layer provides tracking, targeting and advanced warning of missile threats.
- A custody layer provides «all-weather custody of all identified time-critical targets».
- A deterrence layer provides space situational awareness by detecting and tracking objects in space to help satellites avoid collisions.
- A navigation layer provides alternative positioning, navigation and timing services in case GPS is blocked or unavailable.
- A battle management layer is a command, control and communication network augmented by artificial intelligence that provides self-tasking, self-prioritization, on-board processing and dissemination.
- A support layer: includes ground command, control facilities and user terminals, as well as rapid-response launch services.

It 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, thus appearing to be of much higher interest for the application of SGT model and technology described in the previous section.

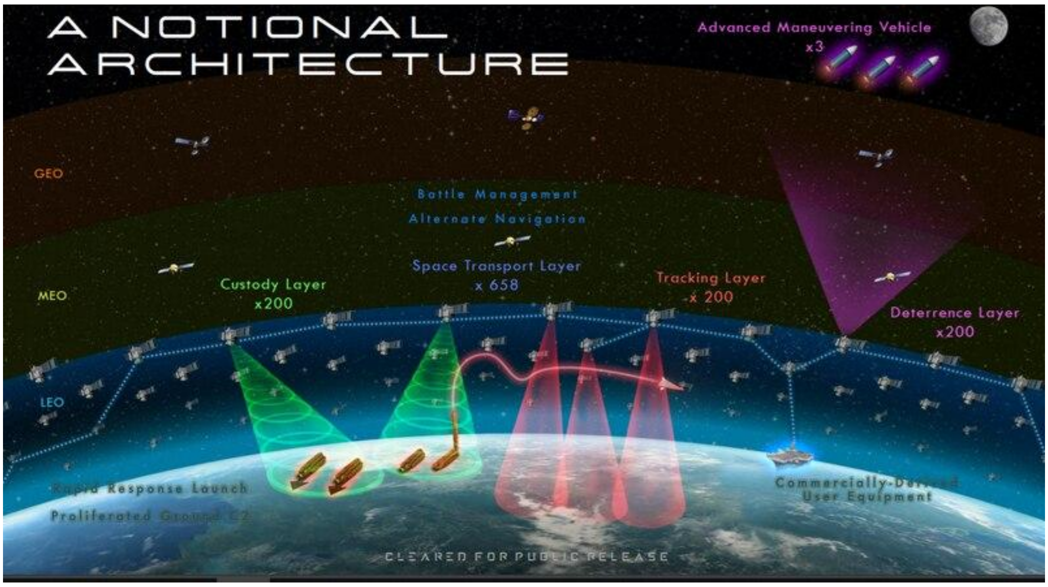


Figure 6 – Next-Generation Space Architecture

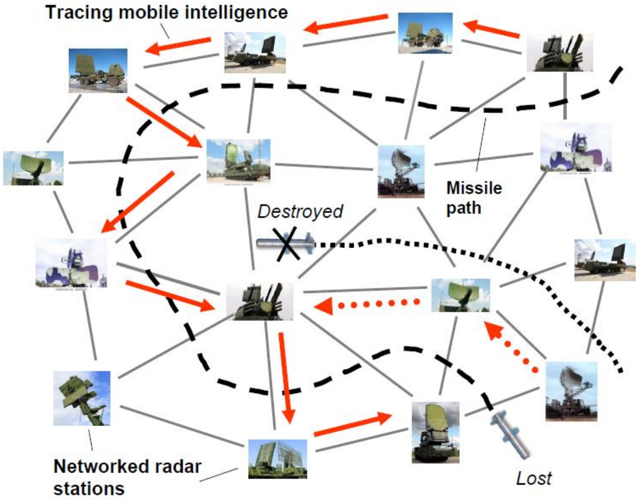


Figure 7 – Distributed objects tracking by a sensor network under SGT

4. Tracing Complexly Moving Objects with Mobile Scenarios in SGL

Distributed sensor networks operating under SGT can catch and follow complexly moving objects, like cruise missiles or hypersonic gliders, throughout the whole region despite the limitations of individual sensors, as shown for terrestrial radar stations in Fig. 7.

The moment when the radar sees a new object for the first time (say, within a given visibility threshold) is a start of a distributed tracing operation, after which the object is monitored by this radar and then it shifts to the neighboring sensors-radars after its visibility is reduced. The history of the object movement and behavior can be collected and updated at the each passed radar by a SGT-produced mobile spatial intelligence individually assigned to this object and accompanying its physical movement via the radar network. Depending on the collected history, such object may be decided to be destroyed; it may also be finally lost after passing through the whole radar-controlled area.

The following self-spreading SGL scenario will accompany the moving object wherever it goes, despite its possible complex and tricky route. When sensors discover new targets, each new target is assigned with an individual tracking intelligence propagating in a distributed networked space in parallel with other similar intelligences. Each such mobile intelligence is itself splitting into parallel branches entering all neighboring sensors to find the one with the best target visibility, into which the tracking scenario subsequently moves, and so on.

```
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(danger(History), done_destroy(Object)));
    max_destination(
      hop(all_neighbors); visibility(Object));
    if(visibility(Object) < Threshold,
       done_output(Object, 'lost', History))))
```

The shown SGL organization can also be effectively used for tracing and destroying hypersonic gliders by cooperating space satellites. Hypersonic weapons break all the rules of the traditional missile defense, as they are much harder to be detected than traditional ballistic missiles. The advanced sensors mounted on satellites can detect the threat and then regularly pass its details to a neighboring LEO sensor with a better vision of the object. If to supply at least some satellites with a capability of impacting the moving gliders, they could be destroyed much more efficiently than from ground stations.

5. Managing Custody Layer under SGT

The SDA Custody Layer is oriented on all-weather custody of time-critical and mostly stationary targets on the Earth by LEO satellites in orbits, which need to pass the observation duty to other satellites when moving away from the controlled objects, as shown in Fig. 8. (For simplicity, only a single orbit with multiple satellites following each other in space is shown, whereas real satellite constellations may include many intersecting orbits.)

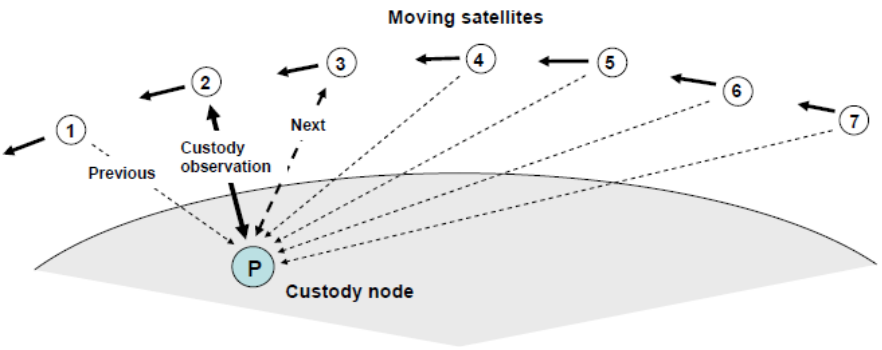


Figure 8 – Observation of stationary custody node by moving satellites

5.1. Continuous Custody Observation by a Mobile Scenario

Observation and monitoring of a stationary object on the Earth by moving LEO satellites may be, in some sense, considered as similar to the previous example where mobile objects were physically moving through the network of communicating satellites, which themselves were moving in space. The custody node may itself be considered as an object moving through the satellite network, and this can be perfectly expressed in SGL. The movement from satellites to satellites will take place when a neighboring satellite appears to be closer to the same location on the Earth, into which the mobile intelligence with an updated history of the watched custody node will now move, and so on. This can be described by the following SGL scenario, which is also depicted in Fig. 9.

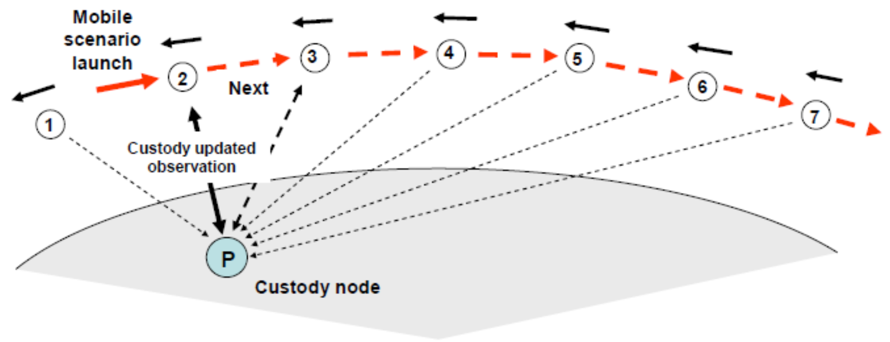


Figure 9 – Managing custody observation by a mobile scenario in SGL

```
frontal(Custody = X_Y, History); Neighbors = distance);
min_destination(hop_sats(all); distance(WHERE, Custody));
repeat(
  if(distance(WHERE, Custody) <= threshold,
     (output(update(History, observe(Custody))); sleep(delay)),
     min_destination(hop_sats(Neighbors); distance(WHERE, Custody))))
```

5.2. Introducing a Virtual Layer for Advanced Custody Support

The custody layer may need constant observation not only of an object or objects at a certain location, but also those distributed over a certain (even vast) area also connected with each other, like a spatial infrastructure. In addition, this may need involvement of a number of satellites simultaneously overseeing different parts of this infrastructure to work together as a system. Such organization can be effectively provided under SGT too, by introducing a higher virtual custody layer constantly overseeing the corresponding physical objects and their infrastructures, regardless of constant movement of satellites around the globe. An example of such a layer consisting of two virtual nodes P and Q, whose physical nodes are similarly named, is presented in Fig. 10. The following SGL scenario creates and constantly supports such virtual nodes located in substituting each other satellites.

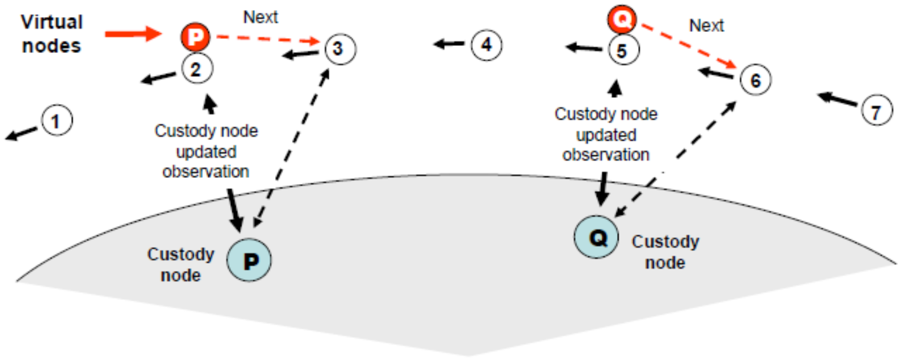

Virtual nodes
P
Next
1
2
3
4
Q
Next
5
6
7
Custody node updated observation
Custody node updated observation
Custody node
P
Q
Custody node

Figure 10 – Introduction and management of the virtual custody layer

```
frontal(Name, Custody), Neighbors = distance); nodal(History);
parallel(
 (Name = P; Custody = XP_YP),
 (Name = Q; Custody = XQ_YQ));
min_destination(hop_sats(all); distance(WHERE, Custody));
create_node(Name);
repeat(
  if(distance(WHERE, Custody) <= Threshold,
     (update(Custody); update(History, observe(Custody))),
      DOER = min_destination(hop_sats(Neighbors); distance(WHERE, Custody))))
```

Using special environmental variables designated as DOER, the virtual P and Q nodes are regularly relocated to other neighboring satellites when the current ones, hosting these nodes, move away from the ground locations they have to observe. This constantly preserves spatial correspondence between physical nodes and their virtual copies regardless of the movement of satellites. (Of course, if there are enough of them effectively distributed throughout the globe, or at least present around the area of interest, and they can directly communicate with each other.)

Any semantic relations can be established between such virtual nodes which may reflect different links or connections between the related physical custody nodes (for example, roads, communication channels, or command and control subordination), as shown in Fig. 11, with a semantic link set up between these nodes by:

```
hop_node(P); linkup(Semantic, node(Q)) or
hop_node(Q); linkup(Semantic, node(P))
```

The virtual nodes connected with each other by a semantic relation will be always preserving this relation between them on the internal SGL interpretation level, which is automatically adjusting to the propagation of nodes between moving satellites, the latter being their temporary «doers».

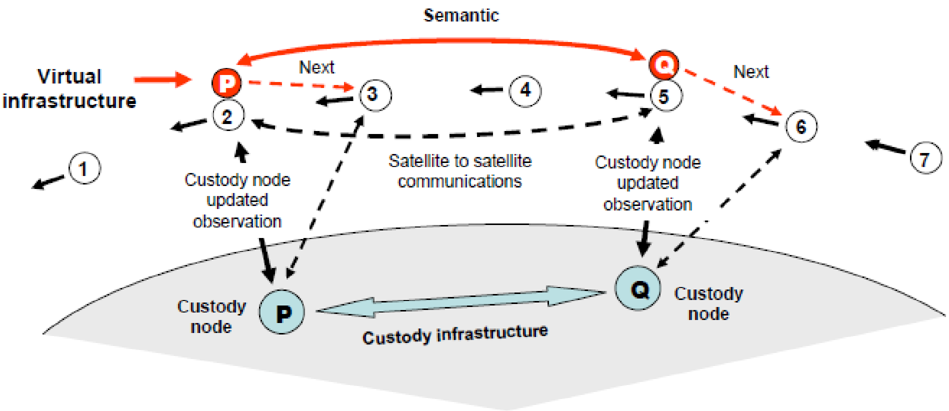

Semantic
Virtual infrastructure
P
Next
3
4
Q
5
Next
6
7
2
1
Custody node updated observation
Satellite to satellite communications
Custody node updated observation
Custody node
P
Custody infrastructure
Q
Custody node

Figure 11 – Introduction and management of the virtual custody layer

Such virtual layer with any number of nodes and any number and types of semantic links between them can be effectively supported under SGT in constellations of moving and communicating satellites, with several examples of some elementary operations using this layer shown in the next section.

Figure 12 – An example of a multiple orbit satellite constellation

If semantic links between virtual nodes reflect physical links between custody nodes, and the physical distance between these custody nodes allows communication between satellites which happened to be over them, then traversing of such semantic links can be implemented by direct contacts between the satellites. This may avoid routing from satellites to satellites via other satellites, which may be rather complex in dynamic network topologies from LEO satellites, like in [27, 28], which may generally include very different orbits as shown in Fig. 12. SGT, however, with parallel and freely propagating spatial scenarios can directly work with dynamic and unpredictable network topologies; it can also simulate and implement any existing routing techniques, including those with dynamically established routing tables, as in [19, 22–24].

6. Examples of Spatial Operations via the Virtual Layer

• Distributed Command and Control from Virtual to Physical Layer

Supposing P is a manager, and Q is a subordinate, the scenario directly starts in the virtual node P, lifts an accumulated there history of behavior of physical custody P, and then goes through the semantic link to the virtual node Q. After comparison with the accumulated there behavior history of physical custody Q, it decides, directs and executes appropriate actions in custody Q.

```
hop_node(P);
frontal(Hist = History; Order = actions); nodal(Custody);
hop_link(Semantic);
if(danger(Hist, History),
   (access(Custody); execute(Order)))
```

• Verifying Changing Distance between Remote Custody Locations

Directly starting from virtual node Q, it lifts the related current custody physical coordinates, then goes with them to the virtual node P via the semantic links between them, and lifts there physic related custody coordinates as well. Then it determines physical distance between the two physical custody nodes, and if it finds out that the nodes are too close to (or on the opposite, far away from) each other, issues a «danger» message with the distance found, which will be delivered to some global management level. The SGL scenario in the previous section was constantly supporting the virtual network level and regularly updating physical locations of custody nodes in the custody variables (in case they could change their positions in physical space). The following SGL scenario will repeatedly work with a certain delay between different iterations.

```
hop_node(Q);
frontal(Cust, Threshold = distance, Distance);
whirl(
  Cust = Custody; hop_link(Semantic);
  Distance = distance(Cust, Custody);
  if(Distance <= Threshold,
     output(“Danger”, Distance));
  sleep(delay))
```

• Discovering and Tracing Hypersonic Gliders

The following scenario, starting in virtual node P and continually operating there, try to define through the observed physical custody the moment of launch of a hypersonic glider there. After fixing such launch, the glider-tracing mobile intelligence is activated in SGL, which will accompany this glider wherever it goes via the satellite network. This will be accomplished via the sensors mounted on satellites which can communicate directly, with elimination of this object if needed and possible, as presented in Fig. 13, similar to the tracing scenario described in Section 4. The SGL scenario will work repeatedly and endlessly, and if a new glider launch is detected in the observed custody, another tracing intelligence will be associated with this object and it will follow it via the satellite networks, and so on.

```
frontal(Neighbors = distance);
hop_node(P);
whirl(
  History = observe(Custody);
  if(belong(glider_launch), History),
     free_repeat(
       loop(visibility(glider) >= Threshold);
            if(danger(glider), done_destroy(glider)));
       max_destination(
        hop_sats(neighbors); visibility(glider)))))
```

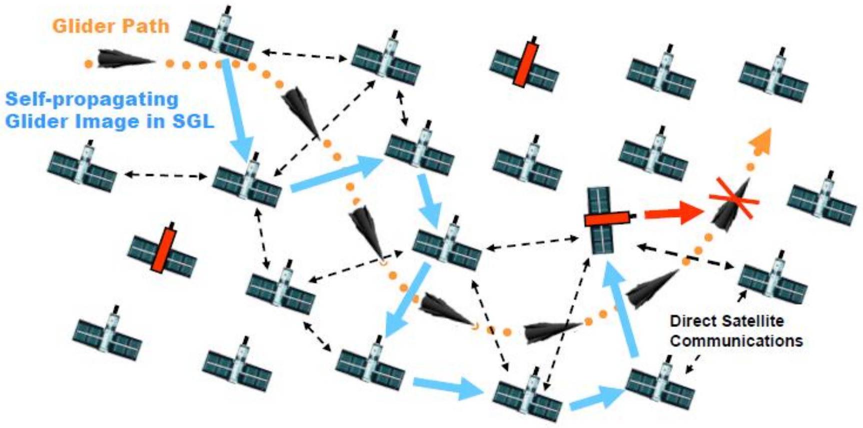


Figure 13 – Tracing and destruction of hypersonic gliders by cooperating satellites under SGT

7. Conclusions

The SGT’s self-evolving holistic parallel and the distributed philosophy, model and technology were found to be adequately matching the current and future plans of the space conquest, especially such recent project as the SDA New Space Architecture based on numerous cooperating satellites in predominantly LEO orbits. The main problem with cheap and relatively simple LEO satellites is that they are rapidly changing their positions over Earth locations, and to provide continuous observation, they regularly need to transfer their duties and accumulated information to other satellites. That can be effectively solved by the super-virus-like SGT and its basic SGL language, which navigate in parallel multi-satellite constellations and solve both continuous custody observation and tracing hypersonic gliders by collectively operating satellites. SGT can also automate other functions and layers of this architecture, allowing, for example, full control and management from space of any terrestrial and celestial operations, missions and campaigns. Taking into consideration the experience of previous technology versions used in different countries, it can be quickly implemented, at least its operational basics, even within standard university environments, with the author always ready to help with this.

This is a starting paper on multiple satellite management with the developed technology, with more publications and conference presentations planned, a new book included, and a new international patent is being prepared too. In the subsequent papers it is also planned to consider how to provide in SGT high AI level for the satellite constellations, including providing gestalt-based holistic solutions in distributed systems, already investigated in detail in [23], and also supplying multi-satellite collectives with high spatial awareness and even a sort of spatial consciousness, recently considered for distributed systems too [19]. The use of novel ideas of DARPA’s Mosaic Warfare initiative [21, 22] for organizing the overall New Space Architecture in a more flexible self-restructuring and self-repairing way, especially in case of its partial damage and destruction, say, by adversaries, which can also be properly handled by self-propagating, self-modifying and self-recovering SGL code, is considered as well.

REFERENCES

1. United Nations Register of Objects Launched into Outer Space. The United Nations Office for Outer Space Affairs. URL: http://www.unoosa.org/oosa/en/spaceobjectregister/index.html.

2. Martin G. NewSpace: The «Emerging» Commercial Space Industry. URL: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20140011156.pdf.

3. Bockel J.-M. The Future of the Space Industry. *General Report*. 2018. Nov. 17. URL: https://www.nato-pa.int/download-file?filename=sites/default/files/2018-12/2018%20-

%20THE%20FUTURE%20OF%20SPACE%20INDUSTRY%20-%20BOCKEL%20REPORT%20-%20173%20ESC%2018%20E%20fin.pdf.

4. Min Read Space robotics market to reach $3.5bn by 2025: GMI report. 2019. May 12. URL: https://satelliteprome.com/news/space-robotics-market-to-reach-3-5bn-by-2025-reveals-gmi-report/.

5. Strategic Defense Initiative. *The White House*. 1984. URL: https://fas.org/irp/offdocs/nsdd/nsdd-119.pdf.

6. The Strategic Defense Initiative: Program Facts. 1987. URL: https://www.everycrsreport.com/files/19870722_IB85170_64d13e614c37eecbed39c00741ddfb269f814fef.pdf.

7. Gattuso J. Brilliant Pebbles: The Revolutionary Idea for Strategic Defense. 1990. URL: https://www.heritage.org/defense/report/brilliant-pebbles-the-revolutionary-idea-strategic-defense.

8. Brilliant Pebbles. URL: https://en.wikipedia.org/wiki/Brilliant_Pebbles.

9. Space Development Agency Next-Generation Space Architecture. 2019. https://www.airforcemag.com/PDF/DocumentFile/Documents/2019/SDA_Next_Generation_Space_Architecture_RFI%20(1).pdf.

10. Magnuson S. Web Exlusive: Details of the Pentagon's New Space Architecture Revealed. 2019. URL: https://www.nationaldefensemagazine.org/articles/2019/9/19/details-of-the-pentagon-new-space-architecture-revealed.

11. Messier D. Space Development Agency Seeks Next-Gen Architecture in First RFI. 2019. URL: http://www.parabolicarc.com/2019/07/07/space-development-agency-issues-rfi/.

12. Insinna V. Space agency has an ambitious plan to launch 'hundreds' of small satellites. Can it get off the ground? *DefenceNews*. *Space*. 2019. April 10. URL: https://www.defensenews.com/space/2019/04/10/sda-has-an-ambitious-plan-to-launch-hundreds-of-small-satellites-can-it-get-off-the-ground/.

13. Research on space debris, safety of space objects with nuclear power sources on board and problems relating to their collision with space debris. *Committee on the Peaceful Uses of Outer Space*. Vienna, 2019. URL: http://www.unoosa.org/res/oosadoc/data/documents/2019/aac_105c_12019crp/aac_105c_12019crp_7_0_html/AC105_C1_2019_CRP07E.pdf.

14. Sapaty P.S. WAVE-1: A new ideology of parallel processing on graphs and networks. *Proc. Intl. Conf. Frontiers in Computing*. Amsterdam, 1987. 10 p.

15. Sapaty P.S. A distributed processing system: European Patent N 0389655. Publ. 10.11.93. 40 p.

16. Sapaty P.S., Corbin M.J., Borst P.M. Mobile WAVE programming as a basis for distributed simulation and control of dynamic open systems. *Report at the 4th UK SIWG National Meeting, SGI Reality Centre*. Theale, Reading, 1994. October 11. 12 p.

17. Sapaty P.S., Corbin M.J., Seidensticker S. Mobile intelligence in distributed Simulations. *Proc. 14th Workshop on Standards for the Interoperability of Distributed Simulations* (IST UCF, Orlando, FL, March 1995). Orlando, 1995. P. 1045–1058.

18. Sapaty P.S., Borst P.M., Corbin M.J., Darling J. Towards the intelligent infrastructures for distributed federations. *Proc. 13th Workshop on Standards for the Interoperability of Distributed Simulations* (Orlando, FL, Sept. 1995). Orlando, 1995. P. 351–366.

19. Sapaty P.S. Symbiosis of Real and Simulated Worlds under Spatial Grasp Technology. Springer, 2021. 305 p.

20. Sapaty P.S. Advanced terrestrial and celestial missions under spatial grasp technology. *Aeronautics and Aerospace Open Access Journal*. 2020. Vol. 4 Issue 3. URL: https://medcraveonline.com/AAOAJ/AAOAJ-04-00110.pdf.

21. Sapaty P.S. Mosaic Warfare: From Philosohpy to Model to Solution, Mathematical Machines and Systems. 2019. N 3. URL: http://www.immsp.kiev.ua/publications/articles/2019 /2019_3/03_Sapaty_19.pdf.

22. Sapaty P.S. Complexity in International Security: A Holistic Spatial Approach. Emerald Publishing, 2019. 160 p**.**

23. Sapaty P.S. Holistic Analysis and Management of Distributed Social Systems. Springer, 2018. 234 p.

24. Sapaty P.S. Managing Distributed Dynamic Systems with Spatial Grasp Technology. Springer, 2017. 284 p.

25. Sapaty P.S. Ruling Distributed Dynamic Worlds. New York: John Wiley & Sons, 2005. 255 p.

26. Sapaty P.S. Mobile Processing in Distributed and Open Environments. New York: John Wiley & Sons, 1999. 410 p.

27. Xiaogang Q.I., Jiulong M.A., Dan W.U., Lifang L.I.U., Shaolin H.U. A survey of routing techniques for satellite networks. *Journal of Communications and Information Networks*. 2016. Vol. 1, N 4. P. 66–85. URL: http://www.infocomm-journal.com/jcin/EN/10.11959/j.issn.2096-1081.2016.058.

28. Madni M.A.A., Iranmanesh S., Raad R. Review DTN and Non-DTN Routing Protocols for Inter-CubeSat Communications: A comprehensive survey. *Electronics*. 2020. N 9. P. 482. URL: https://www.mdpi.com/2079-9292/9/3/482.

Стаття надійшла до редакції 27.01.2021