Topological Clustering as a method of Control for Certain Critical-Point Sensitive Systems

M. J. Dudziak
Institute for Innovative Study
IIS
Washington, DC (USA)
martin@instinnovstudy.org

***Abstract*— New methods can provide more sensitive modeling and more reliable control, through use of dynamically-alterable local neighborhood clusters comprised of of the state-space parameters most disposed to be influential in non-linear systemic changes. Particular attention is directed to systems with extreme non-linearity and uncertainty in measurement and in control communications (e.g., micro-scalar, remote and inaccessible to real-time control). An architecture for modeling based upon topological similarity mapping principles is introduced as an alternative to classical Turing machine models including new "quantum computers."**

Keywords—cybernetics, control system, stability, multi-agent

I. Introduction

In a seminal paper of a decade ago, "Mathematical Models of Catastrophes: Control of Catastrophic Processes," Arnold and co-authors [1] present what may be considered as the starting basis and platform for identification of a class of complex problems characterized by extremes, critical points, singularities and catastrophe functions. These may be named Extreme Complex Systems (XCS). Arguments can be made that such systems are more prevalent than perhaps previously considered, and that such types of systems and their consequent challenges for control theory and engineering are on the increase, with respect to both natural and human origin.

Arnold et al introduced and emphasized connections between singularity theory and bifurcation models in non-linear dynamical systems with catastrophe theory and suggested a basis for "qualitative analysis of complicated systems depending on parameters, such as life supporting systems in real life." The inclusion of "qualitative" concepts in mathematics and cybernetics may be less common in an era dominated by large-scale, high-speed numerical processing (calculation steps). In the domain of extreme behaviors within highly dynamic and erratic systems that require (increasingly) numerical and automated control, the qualitative aspects need to be considered with renewed attention. This leads to an intentional move to discover new "computational mechanics" that will serve qualitative discrimination and decision tasks.

A few direct quotations from [1] are strongly pertinent to the challenges of managing systems that are inherently nonlinear and poorly understood in terms of any finite set of parameters selectable as a working model:

> "The models of real process which are important for the life of our civilization involve many parameters. …
>
> Even the creation of an adequate model is a great problem. Sometimes even the determination of the spaces of internal and external parameters, the existence and smoothness of the corresponding relation or properties of dynamical systems are completely unclear. …
>
> The <u>qualitative analysis based on the simplest models</u> might be useful. As soon as a mathematical description of the system is found the Bifurcation and Singularity Theories furnish quantitative models, but the <u>qualitative deductions seem to be more important and at the same time more trustworthy</u>: they <u>do not depend on the details of the functioning system, whose mechanism and numerical parameters may be insufficiently known</u>.
>
> Napoleon criticized Laplace for his "attempt to introduce the spirit of infinitesimals in government". The mathematical theory of singularities is this part of the contemporary infinitesimal analysis, without which a conscious management of complicated and poorly known nonlinear systems is practically impossible." [1]

The systems we term XCS may be fundamentally different from many other systems that exhibit complicated behavior, even complexity, but without the unpredictability of change and diversity in how key parameters and parameter relations matter in the control of the system as a whole and its components. The extreme behaviors that matter here are those that can tip the whole system "on its head" and "onto the floor," metaphorically speaking. These are not necessarily (or only) concerning scalar values in one or more state-space variables, but involving relationships between system parameters that may radically affect the utility of any prior-established model that is employed in a control system. If control must shift between models and key parameters and the borders of those model utility-spaces are poorly known, this can lead to loss of control, turbulence, instability, catastrophe.

The goal of any method of control must include reliability and the avoidance of losing control. This requires the ability to adapt and shift from one method or model to another. As complexity and the complicatedness of performing the modeling function increases, the performance of any control system that is entirely or substantively dependent upon numerical sequences of calculations, all tied to specific pre-identified parameters originating within the system being modeled, will be subject to degradation. An alternative way of modeling, one that is more responsive to changes in a system that are not bound to specific parameters and their relationships with one another, will be more likely to offer the better adaptivity that is desired. For this we look to discover different ways of mapping, perceiving change, and detecting qualitative, wholistic, and "non-locally" connected phenomena and to do it in ways that can be computationally feasible.

II. Foundations

Consider a model m constructed for control of system S. Model m may be singular or there may be a set of such models $M = \{m_1, m_2, m_3, \ldots\}$ which can be invoked and applied for control of S. (For example, there can be different operating conditions for an autonomous vehicle, depending upon traffic density, weather and highway conditions, etc.) The optimization (success) of control for S depends upon how accurately and completely m (or set M) accommodates the actual dynamics of S. Certain systems, both natural and man-made, are such where unpredictable and non-deterministic changes impact how any one model m can reliably work for S, or how changes (choices) among such models in set M can be performed in "real-time, real-world" operating scenarios.

Arguably, there are more such extremes emerging in the types of systems that are desirable or needing automated control (or virtual control in the sense of accurate modeling and simulation) in the contemporary era. Examples include weather patterns and climate trends, cooperative robots, large-scale wireless communication networks (including the "internet cloud" and "internet of things" configurations), and virtually everything connected with our large human populations and activities such as transportation traffic, epidemiology of infectious diseases, agriculture and food distribution, and mass-commodity business trends. Table 1 illustrates several such XCS that are important today and for which new methods of model and control are needed.

The expanding complexities derive from emerging new relationships between subsystems that influence positively or negatively the roles and effective powers of different parameters in any formal models that have been or conceivably can be constructed as generic, deterministic methods of control. If the model m (or finite set M of models as selectable choices) works "most of the time" or "in most circumstances," but this interval cannot be well-defined and bounded, specifically regarding under what conditions (involving other system parameters), then as the uncertainties about when to rely upon any model m_i increase, the confidence that the overall system S will not enter into a critical or catastrophic state decreases. At some point one can gamble on the low likelihoods of adverse conditions taking over, but as history has demonstrated repeatedly, this is a dangerous game and the outcomes can be severe [2,3].

TABLE I Major 21st Century Extreme-Complex Systems

General Problem Area	One major dynamic event-type	Critical (singularity) region (catastrophe function behavior)	Consequences of failure to solve the "critical-point"
Climate change and global warming	Surface ice melting, especially in areas of large contiguous surfaces - Greenland and Antarctica.	"Spike" in water/air warming due to Latent Heat Fusion process as result of critically diminished ice AND also amplified methane releases	Society unable to cope and social structures collapse, leading to a global crises disrupting infrastructures including food supplies
Asteroid impact event	Asteroid or fragments of size 100m or larger	Impact in a location that severely affects human life	Major local or regional or global eco-disaster(s)
Increased variance of viral mutations with decreased separation of populations	Pandemic	Mutation achieves both optimal human-to-human transmission AND tolerance for lower respiratory tract	Pandemic with lethality levels exceeding 1918 flu pandemic scale
Global financial system fragility and susceptibility	Major crash in securities markets due to viral social media, cyberattacks or credit-default-swap scenarios	Catastrophic collapse ("Sept. 2008" on global scale and more intense effects)	Financial collapse affecting basic infrastructure industries and inevitable conflicts

There are strong arguments for a type or order of solutions, such that are not available through conventional deterministic models or analysis and calculation, and for which a fundamentally new method of problem-solving is necessary, one that demands a new architecture for computation as well. The latter demand is one that we introduce here along with the argument that there are ways using catastrophe theory, singularity theory, and special attention to mapping high-dimensional parameter sets to low-dimensional sets. The latter can provide models (multiple, as a discrete set of choices) that are themselves approximated, incompletely specified at any given time, and which are dynamically built and refined through real-time ("run-time") measurements and decision processes that are fundamentally stochastic and random or semi-random in nature. A key assertion is that such measurements and even semi-random, stochastic selection algorithms, must be directed at borders and boundary conditions – the regions in any state-space or component thereof where there are significant (sharp, sudden, anomalous, asymmetrical) changes in those key parameters of the model.

Such models can then be manipulated more easily and with both fewer time constraints (e.g., in computation) and more variety (number of possible outcomes, as well as greater visualization and use of geometry). This can help solve increasingly complex and emergent ("first instance") problems. However, doing so will arguably require a new method of computation apart from the calculating machine of finite instructions operating upon binary representations (or variants such as qubit-based "quantum computers"); namely, the classic and contemporary Turing Machine. A different approach to computation is suggested, along with the means to attain (implement) such, and this is based upon topological modeling which involves clustering of local, cellular-like regions in a mapping process from regions of the state-space of a system S into a simpler, fewer-dimensional, fewer-parameter model m. This model can be simulated digitally using today's computers, but it can be faster and more efficient if it can be implemented physically using materials that will change their topology in response to input signals from the changing external system S.

III. Requirements and Goals

A. Requirements

Many of the XCS that demand better control are systems where spatial and temporal constraints prevent the use of and reliance upon what has been for decades the answer to complexity – faster and more powerful (numerical) computers and algorithms (including virtually the whole bandwidth of "artificial intelligence") that require more CPU speed, more memory, more data storage, and simply, more data going into the process. However, looking especially into the relatively near-horizon of the next ten to fifty years, it is clear that many XCS solutions will require something different from a supercomputer or a cryogenically-supported qubit-based "quantum computer" (also a Turing machine, at heart).

Consider a cooperative robot system conducting operations with asteroids in deep space, potentially millions of miles from any human control station, or blocked from direct signal reception and transmission by some other object (asteroid, the Moon, another planet). Such an ensemble must accommodate significant autonomy and this will surely require advanced capabilities for self-modification, reconfiguration of components and software, and even self-repair and new construction [4,5]. Much the same can be said about modeling and control of a swarm of robots and sensors deployed in a large agricultural or livestock management setting, because of the need for compact size, speed and also economic factors. Discrete digital wi-fi networks linking an indeterminate number of mobile processing elements to a central server farm is not suitable for the operating conditions, mechanical requirements, and economics of mainstream farming [6].

Another example can be a system that involves no moving parts at all. This could entail the detection and comparison of trends and dispositions in large ensembles (people, vehicles in traffic, crop regions in an agricultural zone) such that a field-like or surface-like transition is significant information that can be used with more conventional "AI" techniques to predict where some region or sector will emerge into a near-critical state. The data space may be a mix of digital and analog information. The processing may be highly distributed spatially. The challenge includes knowing where to initially direct attention. Biology may have worked out a good method long ago (judging from the variety, distribution and sustainability of many species of organisms) which can be useful for the engineering world in modeling "NP-hard" and extreme-complexity systems. There is strength in how surfaces (physical, 3D, and semantical) are both measured, shaped, and differentiated, that can be useful in the representation and management of information to control the systems which generate such surfaces, provided that there can be a different way to compare the information content in such shapes, other than only the conventional, historical manners of numerical representation (and comparison of such).

B. Operating Goals

Leading up to definition of how a topological process of computing can be engineered are some basic goals:

[1] The definition of a formal logic and an algebraic language that can be utilized, in generalized ways, for identifying, localizing, and encapsulating the critical points, the singularity regions, governing extreme complex systems (XCS)

[2] Identifying manageable, reliable, accurate ways of measuring the outcome of using such language to control the target systems, to either avoid such critical points in their behaviors or to navigate safely and securely through such regions.

[3] Defining a mechanism that can physically embody expressions defined by such language, enabling the execution of functions in a manner that is reproducible with sufficient accuracy and quality for engineering applications.

Arnold et al focus in their paper upon two general classes of catastrophe functions and singularity theory applications. It is suggested here that these two generic classes are extensible to other problems in both natural and synthetic (engineered, "man-made") systems. Moreover, these two classes can help illustrate what must be accommodated in the triad of goals of a logic, a language and a machine.

First, in areas of engineering: "loss of stability and jump–like dynamics in mechanical systems with rigid and elastic elements" and the problem of "bifurcation of steady state positions of loaded elastic beam" depict two generic types of problems and both are clearly not limited to civil and structural engineering of buildings and bridges. The essence here is the dynamics of systems that can be decomposed (analyzed, broken down, or conversely, built from) into parts that are rigid or elastic in varying degrees, where one rigidity or elasticity level (or range) can be optimal or where another level or range can be catastrophic. This brings to mind tensegrity systems and structures of types ranging from geodesic domes to tensegrity towers, but also this may be applied to non-mechanical, non-structural problems. An example could be the collective body of tasks to be performed by a set of coordinated agents (robots) in agricultural cultivation or in preparation of a region of lunar or asteroid surface for mining. This can be seen as a different perspective on the broader problem of large-scale project management, a separate topic for investigation, mentioned here only to point out the diversity and expanse of the function classes for which a new computational model is advantageous, even required.

Secondly, there is an illustration in the broad area of ecosystems: resource-exploitation models, such as incorporate some laws or rules constraining resource growth). These are different from systems built of discrete parts like bridges and buildings. Instead, there are quantities of substances, which may be renewable or non-renewable, and degrees of quality as well. Additionally there are relationships between them as sources (supplies) and outputs (demands). Food production, energy production, materials (textiles, domestic commodities), and consumer electronic products are some examples.

Here there are optimization problems (e.g., maintaining the supply-chain for production and distribution) but the complexities can change the state-space configurations which have been established at some past time, rendering the utility of the current models obsolete or at least unreliable. As with the first class, there is the question of which system parameters are most significant and when to switch between variant models; this implies a decision of observation and measurement, and timing – when to look at the same data space but in a different light. This is the essential act of intelligence, natural and artificial alike. Shifting perceptive frameworks: seeing the broken tree limb as a tool for prying apart some rocks, or as something to burn and provide light and heat, or as a better way to hunt for prey.

Thus we are motivated to accommodate decisions about classes of functions that can be observed or predicted to lead into critical states, singularities. Topologically we are aiming to find regions of extreme curvature and also asymmetries that could be interpreted as precedents for irreversible conditions. As in Fig. 1, we are concerned with the cusps, folds and other features and their formation, more so than with other regions.

Fundamental Types of Catastrophes

Typ katastrofy	# of control parameters	# of state variables	Energetic function	First derivative
Fold	1	1	$x^3/3 - ax$	$x^2 - a$
Cusp	2	1	$x^4/4 - ax - bx^2/2$	$x^3 - a - bx$
Swallowtail	3	1	$x^5/5 - ax - bx^2/2 - cx^3/3$	$x^4 - a - bx - cx^2$
Butterfly	4	1	$x^6/6 - ax - bx^2/2 - cx^3/3 - dx^4/4$	$x^5 - a - bx - cx^2 - dx^3$
Hyperbolic umbilic	3	2	$x^3 + y^3 + ax + by + cxy$	$3x^2 + a + cy$ $3y^2 + b + cx$
Elliptic umbilic	3	2	$x^3 - xy^2 + ax + by + cx^2 + cy^2$	$3x^2 - y^2 + a + 2cx$ $-2xy + b + 2cy$
Parabolic umbilic	4	2	$x^2y + y^4 + ax + by + cx^2 + dy^2$	$2xy + a + 2cx$ $x^2 + 4y^3 + b + 2dy$

Fig. 1 Several variations of catastrophic functions [7]

This differs from searches for local or even global maxima or minima or only static limits. Rather, we are paying attention to both what may be emerging in a local part of a system as well as what we can assert are more global changes in behavior. These can be indicators of something more radical, asymmetrical and of critical value about to happen in a particular region of the state-space or at some particular point in time. How can we do this detection work in a manner that does not depend upon a fixed starting point or limit-set?

IV. Topological Modeling

A new approach is based upon a conceptual framework that gives a topological description of systems as composites of active state-space regions. These regions are definite by parameters and their relationships as defined within some models. The abstract surfaces and volumes can be manipulated as dynamic-bounded cellular automata networks (clusters). Each cluster is a region within the state-space of the system being modeled. A topological cluster connects some finite number of parameters within some ranges over their potential values, where by definition of the model there is some relationship between these parameters. An example could be measurements of stress within elements of an aircraft wing or fuselage, within finite sections of the wing geometry, or measurements of energy consumption within a power network, or pressures within a fluid distribution network. The cluster geometry is not constrained to physical space-time coordinates.

The cluster defines some arbitrary but roughly expected bounds to the expected behaviors of the variables during the time-evolution of the system. They are chosen initially at random and also maintained on a stochastic basis. Seminal foundations for this approach can be found in SPSA (simultaneous perturbation and stochastic approximation) and randomized algorithms for control [8,9,10]. These have been extended recently to experiments for control of turbulence in aerodynamics [11].

Any system that we want to control actively or passively (e.g., purely to simulate) can be represented as an m-dimensional surface (c-surface, control-surface) in an embedding n-dimensional space (e-space). These spaces can be divided, randomly or following a structured pattern, into cellular regions that have defined borders with one another. The borders of the regions are significant as indicators of approaching criticality points and catastrophe regions. There may be inexact, imprecise elements to this mapping transformation but it will be approximate and sufficient.

So we have the following:

embedding-space (e-space, E) : n dimensions which pertain to the parameters of significance within the space.

control-surface (c-surface, C) : m dimensions which pertain to the parameters of significance within the surface.

In all cases $m < n$, $n \geq m+1$

Example: Consider an airplane wing as one example of a c-surface. It has a continuous surface (upper and lower) that is wrapped around its frame. (We omit for this exercise the region of the wing that is connected to the airplane fuselage.) This surface is mappable to a simple 2d surface. This in turn can be wrapped around a sphere. Measurements such as air pressure may be made for points on this wing surface at different intervals. These values may be represented as positive or negative values in a third dimension and the resulting geometry of the wing-as-sphere will be a rough, curved, fuzzy spherical surface that has peaks and pits, hills and valleys on its surface. Although there may be additional parameters (and thus dimensions) introduced into the c-surface representation, we will presently omit others beyond air pressure, within this example case.

The e-space is the physical environment, the 3d world in which the airplane operates. It may have a higher dimensionality based upon known parameters of significance that include air pressure, wind velocity, air temperature, humidity, and other measurable data.

Control of the wing will involve changes in the air pressures for all the measurable points on its surface. In terms of the general, abstract, topological model, we need to change the shape of the ball (sphere) and the fractal-like dimensionality of its surface, to some optimal values that will be desirable for how we want to manipulate the wing in flight.

We may know in advance an optimal configuration for the c-surface in its ball-like representation, or we may need to experiment with a variety of topological configurations. We do so by working with a model that is simpler and easier to manipulate in two forms of computation: numerical calculations using a simpler geometry where the model approximates the c-surface, and also what can be called topological computation (conformational analog computation) that is non-numeric and involving a physical analog to the c-surface that can be manipulated in its geometry; it works on the basis of similarities of geometry between the computing model and the c-surface, and the execution of conformational changes within that model that reflect the mapping of similar changes in the c-surface. (It is also referred to as the "pantograph" model of computation, by explicit analogy to the drafting instrument known as the pantograph.)

The Generalized Computing Machine (GCM) is a heterogeneous architecture that operates by performing both types of computation tasks – numerical and topological. These involve digital and non-digital processes. These may also be described as discrete and continuous. The digital computations are what are performed in a conventional "Turing Machine" (TM) type microprocessor. The non-digital processes are performed through a microfluidic MEMS device that employs molecular conformation actions involving the translation of digital information into electromechanical actions within a set (array, field) of molecular components [12,13,14]. These molecular structures can change conformation and those changes can be measured and translated into digital signals that will serve as command and control values for processes governing the behavior of the c-surface system that is the object for control. This type of device, is the subject of current design research and is modeled upon microfluidics employed for DNA and DNA amplification as part of the standard proecure of the well-known polymerase chain reaction (PCR) used in nucleic acid sequence matching [15,16]. For simplicity the entire process can be described as "Topological Computation."

The Topological Computation Process (TCP)

This may be considered as the analog to a digital program that embodies a discrete algorithm. However the TCP has both digital and non-digital components (as described earlier, above).

The embedding-space (e-space, E): n dimensions which pertain to the parameters of significance within the space.
Control-surface (c-surface, C): m dimensions which pertain to the parameters of significance within the surface.

For a given problem (application) there is one e-space and there may be multiple c-surfaces operating within that e-space and also interacting with one another.

The first stage of TCP defines the control environment model:

[1.1] a set of models of the e-space E:

- representation-model (r-model; all that can be represented, but not necessarily changed or transformed within E)
- transformation-model (t-model; all that can be manipulated and changed within E)

[1.2] a set of models of the c-surface C – and C may consist of one or more c-surfaces C_i := {C1, C2, … Cx}

- representation-model (r-model; all that can be represented, but not necessarily changed or transformed within C_i)
- transformation-model (t-model; all that can be manipulated and changed within C_i)

The t-models are subsets of their respective r-models.

For both sets of models, for both E and C, there is a definition of the parameter sets within each type of model.

P[e] is the parameter set for the e-space and $P[c]_i$ is the parameter set for each c-surface C_i within E.

P[e] constitutes all the parameters of E that matter in the problem, and also, which may be targets for modification also. (Bear in mind that within any system "world environment" of an e-space and one or more c-surfaces, it may be that there can be significant changes to the e-space introduced by one or more c-surfaces. (Example: planes taking off and landing at an airport can easily affect the wind space (which is part of the e-space) creating turbulence that will in turn modify the e-space and affect other c-surfaces (other aircraft taking off and landing).)

The P[c] are the parameters that matter in the problem for the c-surface C_i and which are potentially modifiable, controllable. Normally each $P[c]_i$ will be the same parameter types, the same dimensionality (e.g., all the the objects represented by the c-surfaces are the same things – UAVs, UGVs, airplanes, ships, social groups, financial networks, etc.)

The parameter set of a t-model is a subset of the parameter-set of an r-model, since there may be parameters within the r-model that are not computable or not modifiable.

So we have four types of parameter sets:

$P[e]^{\mathbf{r}}$ = all knowable parameters of the e-space E
$P[e]^{\mathbf{t}}$ = all transformable parameters of the e-space E
$P[c]^{\mathbf{r}}$ = all knowable parameters of the c-surface C (and $P[c]^{\mathbf{r}}_i$ is for one c-surface C_i)
$P[c]^{\mathbf{t}}$ = all transformable parameters of the c-surface C (and $P[c]^{\mathbf{t}}_i$ is for one c-surface C_i)

[2] The second stage of the TCP – mapping complex surfaces to simple spheres

Mapping the c-surface(s) to spherical surfaces is a numeric process. The C_i is mapped from its objective geometry (e.g., a component of an aircraft, automobile, or other device) onto a surface that can be approximated to the surface of a sphere. Next we look at the smooth surface and introduce the values for some parameter of interest within the set $P[c]^{\mathbf{r}}_i$. This results in a change to the surface of the sphere – a third dimension, the peaks and valleys. If there are additional parameters within $P[c]^{\mathbf{r}}_i$ then these can be introduced, and now the dimensionality of the surface is increased, but it is still always less than that of the e-space in which this sphere is embedded. This yields one or more c-surface spheres (or simply, "c-spheres") within an e-space.

These are simplifications of the c-surfaces. All of the relevant parameters in $P[c]^{\mathbf{r}}_i$ have been transformed from the geometry of the original c-surface to points in a surface wrapped around a sphere, and these points have additional dimensions of zero or some value relative to the flat surface of the c-sphere. Now any computations done with respect to the interactions between a given C_i and its E or between multiple C_i can be performed in the model of simpler-geometry, simple-topology c-spheres.

[3] The third stage of the TCP – modeling the behavior and interactions of spheres with the embedding space and with other spheres in that space

The next step is to compute how the parameters of the e-space and any other c-spheres that must be taken into account are affecting the features of a given c-sphere. We want to establish functional relations that move from some $P[e]^{\mathbf{r}}$ to some $P[c]^{\mathbf{r}}_i$.

This is the major task of the GCM. Modeling at the simple level of c-spheres within the e-space, and then translating the results into answers that can be interpreted and transformed into actions which effect new types of control of the objects represented by the c-spheres.

There are both digital-numerical computations and analog-topological computations that can be employed here.

In any case, a comprehensive system model is needed for the interactions of both the e-space and the c-surface (or c-surfaces). In this process we must develop within the model his requires identifying those parameters that can be altered, and in the process separating them from those that cannot be modified. This separation process will have been used in the process of creating the simplified environment of c-spheres and e-space – thus, we are at this point concerned with the following processes:

How $P[e]^{\mathbf{r}}$ affect the behaviors of the c-spheres
How $P[c]^{\mathbf{r}}_i$ affect behaviors of other c-spheres and e-space
How $P[c]^{\mathbf{t}}_i$ can be altered for a given c-sphere
How $P[e]^{\mathbf{t}}$ can be altered

Numerical computations: This is where cellular automata can be applied for modeling local neighborhood interactions. It is also where inverse methods can potentially be used, similar to their applications in signal processing and imaging. The goal is to determine how some parameters within $P[e]^{\mathbf{r}}$ are affecting (interacting with) some features in the c-surface. However, this step may be optional. Perhaps we do not need to examine this aspect of causal relations, if we are only interested to find ways to alter the properties of the c-surface.

Topological computations: With a representation of one or more c-spheres, this representation includes the c-sphere geometry and its dynamics – how it can change shape under specific metrics – conditions that can indicate singularity points emerging in the state-space even though unpredicted by any of the established models (set M) that already may exist. This is an important point – setting up a method that can "notice" when anomalies are occurring that do not fit any pre-existing model m_i.

Now we need to translate that into the component of the GCM that can "resonate" geometrically, topologically, with that c-sphere, changing its conformation in a way that can be understood as a direct mapping. We can call this the TPU of the GCM – the Topological Processing Unit. This is the similarity-simplication or "pantograph" function that must be performed.

Once we have "set" or "initiated" the state of the TPU, we are able to apply inputs to it in the form of physical energy signals that will cause the TPU to modify its physical shape and perhaps other properties that occur as a result of conformal shape-changing.

These signals are of two types or classes:

-- what we know from the actual objects represented by the c-spheres, or what we can compute digitally to be some "next projected states"

OR

-- what we want as some future-states of those objects, and we want to use the TPU to non-algorithmically compute how the objects should behave. Thus we will want to know how the simplified c-spheres should behave, and from those states, we can translate information to the actual objects as commands, modifications to whatever subsystems in those objects can do things like change trajectory, velocity, or other parameters which are essentially of the class $P[c]^{\mathbf{t}}$.

What makes the GCM "generalized" is its ability to compute functions by using geometry, by topological shape-changing that mimics the non-discrete, non-digital "analog" behaviors of a theoretically unlimited variety and number of objects which each change some element of shape (position) – some element or elements involving position relative to some embedding space in which the object exists and functions.

Thus, the TPU modifies its geometry in a manner that is determined to be consistent with the ways that the c-spheres can be modified, and this models the more complex behaves of the objects that are reflected in the c-sphere parameter sets and their dynamics.

The new shapes of the TPU are what has been computed on the basis of then translated into modifications of the digital c-spheres. We translate from the TPU into making changes in those parameters for both the c-sphere(s) and the e-space, and these changes are in the parameters known as $P[c]^{t}_{i}$ and $P[e]^{t}$ respectively (since these are the modifiable parameters, about which we can effect alterations). The process of going from the TPU to the c-spheres and ultimately the actual objects of interest is a reversing of the original information path leading from those objects and their behaviors to the TPU inputs that trigger its conformational changes.

From the new states of E and C_i, we can now either control the future behavior of the C_i or we have a better understanding of what there is which we cannot or should not control or pay attention to. This information can now be incorporated into meta-level heuristics and rules that then influence (limit) the stochastic searching and approximation methods used on testing local clusters. This the system can be self-modifying, self-learning. The assumption throughout is that we have achieved these new states (changing the actual objects or having the knowledge of how they can, should or will change) by means that are computationally less time-consuming, and less resource-demanding, than to approach the problem using conventional TM computers.

References

[1] V. Arnold, A. Davydov, V. Vassiliev, V. Zakalyukin, "Mathematical Models of Catastrophes. Control of Catastrophic Processes," *Institute for Applied Systems Analysis, EOLSS Publishers Co.*, 2005

[2] N. Talib, "The Black Swan: the Impact of the Highly Improbable," *Random House*, 2007

[3] M. Holtman, "Paradoxes of professionalism and error in complex systems," J. Biomedical Informatics, 44, 3, June 2011, 395-401

[4] D. Agnolon et al, "Study Overview of the Near Earth Asteroid Sample Return," European space Agency, SCI-PA/2007/004/DA, http://sci.esa.int/trs/40608-near-earth-asteroid-sample-return/

[5] T. Howard, C. Green, A. Kelly, D. Ferguson, "State space sampling of feasible motions for high-performance mobile robot navigation in complex environments," J. Field Robotics, 2008, 25, 6-7, 325-345

[6] K. Sabarina, N. Priya, "Lowering Data Dimensionality in Big Data for the Benefit of Precision Agriculture," Procedia Computer Science, 48, 2015, 548-554

[7] R. Thom, "Structural Stability and Morphogenesis: An Outline of a General Theory of Models." Addison-Wesley, 1989

[8] M. Dudziak, O. Granichin, "Extreme complex systems, uncertain and uncooperative robotic networks, and control strategies based upon stochastic algorithms," *Proc. of the 28th International scientific-and-technological conference extreme robotics (ER-2017)*, November 2-3, 2017, St. Petersburg, Russia, 218-230

[9] O. Granichin, N. Amelina, "Simultaneous perturbation stochastic approximation for tracking under unknown but bounded disturbances," *IEEE Trans. on Automatic Control*, vol. 60, no. 6, 2015, 1653–1658

[10] K. Amelin, O. Granichin, "Randomized control strategies under Arbitrary external noise," *IEEE Transactions on Automatic Control*, Vol. 61. No. 5, 2016, 1328-1333

[11] O. Granichin, T. Khantuleva, O. Granichina, "Local Voting Protocol for the Adaptation of Airplane's 'Feathers' in a Turbulence Flow," *American Control Conference*, May 24–26, 2017, Seattle, USA.

[12] O. Granichin, S. Molodtsov, "Creation of Hybrid Ultrafast Computers and System Programming," *Sankt Peterburg*, 2006, ISBN 5-288-03919-4 (Russian)

[13] M. Dudziak, Control Criticality within High-Complexity and Low-Threshold Systems, and the Role of Topological Quantum Computing in such Solutions, submitted to *SICS Int'l symposium on Control Systems* (Tokyo, 9-11.March, 2018) (pre-print available)

[14] Granichin, O. Vasiliev, V., Computational Model Based Upon Evolutionary Primitives: Turing Machine Generalization, *Int'l Journal of Nanotechnology and Molecular Computation*, 2(1), 30-43, Jan.-March 2010

[15] S. Lee, S. Kim, J. Kang, C. Ahn, "A polymer lab-on-a-chip for reverse transcription (RT)-PCR based point-of-care clinical diagnostics," Lab Chip 8(12), 2008, 2121–2127.

[16] S. Pernagallo, G. Ventimiglia, C. Cavalluzzo, E. Alessi, H. Ilyine, M. Bradley, H. Diaz-Mochon, "Novel Biochip Platform for Nucleic Acid Analysis," Sensors (Basel), 12(6), 2012, 8100–8111