Holonic Systems as Software Paradigms for Industrial Automation and Environmental Monitoring

Marco Calabrese, Vincenzo Piuri
DTI
University of Milan
Crema, Italy
{marco.calabrese, vincenzo.piuri}@unimi.it

Vincenzo Di Lecce
DIASS
Polytechnic of Bari
Taranto, Italy
v.dilecce@aeflab.net

Abstract—**Holon is a powerful metaphor which captures the recursive structure of biological systems and the organization of their decision processes arranged at various granularity abstraction levels. From a computational intelligence perspective, a holon can be conceived as a goal-oriented community of lower-level holons led by more specific targets. Sub-holons co-operate on sub-problems that represent the source problem at a lower knowledge abstraction level. Such a (recursive) hierarchical organization constitutes the so-called holarchy. Holonic thinking is hence particularly suited for complex and intelligent systems modeling: in particular, its success has been proved in the field of Intelligent Manufacturing. Nevertheless, albeit hierarchical and granular thinking are two fundamental prerequisites in Software Engineering, the use of holonic thinking as software paradigm is still flawing in the literature at the moment. In this regard, the paper introduces the concept of 'holonic granule' as a novel software building-block for modeling complex granular systems. Prospective applications of holonic granule-based software models are then commented with particular emphasis to industrial automation and environmental monitoring settings.**

Keywords - holon; granular computing; holonic granule; knowledge representation

I. INTRODUCTION

The term 'holon' was first introduced by Arthur Koestler in 1967 [1], meaning an entity capable of information processing and action, which is either atomic or contains parts which are lower-level holons. A holon plays the role of a whole and a part at the same time.

Koestler's basic idea moved from the concept of organism as a systemic whole to introduce that of multi-level hierarchy conceived as a self-regulating structure of sub-wholes. The latter have to be considered as functional parts of the system they are hosted in; however, at the same time, they also show autonomous behavior which makes them being a whole system as well.

Holons accounts for a recursive interpretation of the concept of system where part and wholes are not considered as separate entities. This part/whole dichotomy reflects on every level of the hierarchy and can be easily observed in the domain of life (e.g., cells in a living organism, human beings in a society, etc…).

In summary, Koestler considered holon as a basic modeling component suitable for building self-regulating hierarchical organizations. In his mind, the main difference between holon and other model-based entities is the appearance of the so called 'Janus phenomenon' (in the roman mythology, Janus was the god reigning over the realm of doors, passages, beginnings and endings). The holon actually shows a Janus face since it contemplates two distinct but complementary perspectives within a unique entity:

- **Top-down**: the side that looks "down" and acts as an autonomous system following its own goals and rules, also giving directions to lower-level components (sub-holons).
- **Bottom-up**: the other side that looks "up" and serves as a part obeying to a higher-level component (super-holons).

It is noteworthy that holon accounts for a multi-level granular system description (top-down and/or bottom-up) by default, due to its own definition. This aspect has entailed a discrete success of holonic modeling approaches, especially in the field of Intelligent Manufacturing [2].

Indeed, in the literature of holonic systems, little attention has been paid to software models and methodologies. This happens despite the fact that computing with granular information in Software Engineering has been attracting the interest of scholars in the Computational Intelligence sector [3] since almost a decade. In particular, the new-borne field of Granular Computing is currently dealing with the issue (refer, for example, to [4]).

978-1-61284-060-4/11/$26.00 ©2011 IEEE

The goal of this paper is to start filling the gap in research and practice by introducing a design methodology, based on holons and holarchies, which takes into account not only the overall structure of the computation and actions, but also the hierarchical organization of information and knowledge at different abstraction levels by means of various knowledge granularities.

The remainder of the paper is as follows: Section II briefly surveys the use of holons in complex systems modeling; Section III presents the basic concept of granularity and computing with granules as they appear in the current literature; core Section IV introduces the novel idea of holonic granule; example applications of holonic granule as a software modeling paradigm in the two fields of industrial automation and environmental monitoring are presented in Section V; Section VI concludes the paper.

II. Holonic Modelling in the Literature

In 1998, Thompson and Hughes [5] introduced a holonic-based theoretical model to describe (human and computer) activities within a given organization. Their work aimed at finding an improved solution to the design of Computer Integrated Manufacturing (CIM) systems.

The authors started their analysis by a simple observation: the most significant initiatives in defining CIM architecture approaches and enterprise modeling show that these approaches are both difficult to use and comprehend, and significantly lacking in scope. They claimed that the existing approaches do not attend adequately to people and organizational aspects, and their relationship with computer-based systems: traditional hierarchical perspectives are predominant, while business-process views (based on the flow through an enterprise towards the customer) are lacking.

According to the authors, a manufacturing enterprise can be represented as a network of semi-autonomous cells, "*alike and fractal in nature*", with the common purpose to satisfy the 'supply=demand' equation. Interesting enough, the cells have a dynamic existence: they exist as long as they have a role to play; their specialization depends on the process involved. In this view, organizational structure "*is provided by the system-subsystem relationship and the classification structure*".

Interaction between subsystems does not imply subsystems loosing their autonomy. There is no superimposed hierarchy of command and control and no hierarchy of decision making. As it happens in biological systems, "*subsystems work autonomously but broadly to the same agenda*" which is a core property for holons.

This kind of 'cooperation in autonomy' capitalizes on the property of *emergence* [6]: some complex system behaviors are evident only at a higher echelon as it happens in biological systems. Granularity levels are then properly accounted for without the need of an external top-down decomposition imposed by a hierarchy of commands/control, but only referring to a system-subsystem part-whole decomposition. The use of this paradigm as a conceptual means for describing complex systems is properly referred to as 'holonic modeling'.

Holonic modeling has been successively endorsed and further formalized in a recent work focusing on Process-Oriented Holonic (PrOH) modeling [7], another methodology that uses holistic thinking and the holon concept to build business process descriptions at different granularity levels. In particular, it has been conceived as a useful support for modelers whishing to analyze business processes in organizations characterized by high complexity, low volume and high variety.

One key element of PrOH modeling is the concept of *granularity*. According to the authors' view, some modelers use the notions of scope (i.e., the range of activities modeled) and/or level (the detail/depth of that modeling) with the intent to frame a model's content [8, 9, 10]. In PrOH terms, this approach is considered as an oversimplification: a more sophisticated notion of granularity is needed. Put in simple terms, it means deciding what goes in and what stays out of the process model [11]. The modeler must decide on the size of each piece of the model (whether that piece is an entity within a model, an entire model or a set of models). It is plain to say that this approach is intrinsically recursive [12] and can be applied at any level of modeling.

PrOH modeling allows for overcoming the traditional 'inside-the-box' task breakdown approach that employs aggregation/reduction patterns in favor of a new 'outside-the-box' methodology based on abstraction/enrichment criteria. In other words, PrOH modeling, as an offspring of holistic thinking, is about providing enrichment/abstraction patterns in a holarchical way. Models explicitly aim to show properties that occur relative to the chosen granularity level. Traditional hierarchical thinking instead aims to define systems in absolute terms.

III. Granularity

Granularity is, by lexicon, the property of resembling or consisting of granules.

In philosophy, granules can be objects, or ideas; in general, they are abstract entities which are self-consistent, at least at the granularity level at which they are considered.

More specifically, a granule can be conceived as any atomic element which is not distinguishable from its peers for manifest features but only for the fact that it represents a singleton (eventually embracing a whole) among other singletons. Under this interpretation, set theory partially grasps the essence of granule with the notion of subsets and elements of a set [13, 14].

Following this direction, a brand new theory called Granular Rough Theory (GRT) has been formalized in very recent times [15]. GRT stems from an ongoing work with the ambitious goal of the redefinition of classical set theory by investigating its granular nature with the sole notion of part-whole relation. In this sense, there could be drawn interesting connections between

GRT and holonic modeling approaches. However, due to the recentness of the proposal, no specific work in the literature can be found on the topic yet.

A. Information Granules

In the field of Computational Intelligence, Lofti Zadeh, the founder of Fuzzy Logic (FL), has been addressing the computational aspects behind the notion of granularity for many years. According to his view, information granules (whether crisp or fuzzy) are [16]:

Def. 1 (information granule): *clump of objects (points) drawn together by indistinguishability, similarity, proximity of functionality*

Form now on in the text, granules and information granules will be used as synonyms.

The Zadeh's first intuition of information granulation as the process of forming information granules can be traced back to late 70's [17]; successively, he formalized that concept in the more general theory of FL [18, 19].

B. Granular Computing

Led by the observation of the human reasoning process, which is inevitably built upon some machinery for handling approximate and imprecise logical inference, Zadeh considers information granulation to be a key aspect of both human concept formation and intelligent information systems.

In the light of FL, information granulation is the basic process of Granular Computing (GrC). Ultimately, it provides a basic framework for Computing with Words (CWW) methodology, i.e., expressing knowledge of observed phenomena in terms of linguistic propositions rather than numerical equations. The real possibilities of GrC and CWW are probably still unclear at the moment since they represent a very recent point of discussion [20]. In our view, as shown further in the text, the two methodologies can greatly benefit from the interaction with the holonic-based perspective both at a theoretical and operational level.

GrC is a relatively new discipline, grown up in almost the same years of first holonic modeling approaches. The term first appeared in 1997 [21] to provide a unique label for a number of models, ideas, applications sprouted from different domains such as machine learning, data mining, high-performance computing and so on.

According to Pedrycz [22], "*GrC deals with representing information in the form of some aggregates (that embrace a number of individual entities) and their ensuing processing*". GrC as opposed to numeric computing (which is data-oriented), is knowledge-oriented and accounts for a new way of dealing with information processing in a unified way. Since knowledge is basically made of information granules, information granulation operates on the granule scale thus defining a pyramid of information processing where low levels deal with ground data and higher level with symbolic abstraction.

IV. Introducing the Concept of Holonic Granule

According to Koestler's original ideas, holon is an intelligent entity playing the role of a part and a whole at the same time. This is a bit weird at the operational level since holon definition requires the same holonic entity having both the properties of a singleton and a community. The problem is: how to do that?

Our proposed solution stems from the observation that holons are agents [23] natively showing an *architectural recursiveness* [24]. As shown before, a holon in fact can be recursively decomposed at a lower granularity level into a community of other holons thus producing a holarchy.

If we also considered a holarchy to behave intelligently as if it were a holon, then the apparent dichotomy between parts and whole would vanish in favor on a new unique computational entity being a holon and a holarchy at the same time. The two roles would then become interdependent and one would not have sense without the other. In particular, viewed from the extern, both holons and holarchies should appear as recursively-decomposable intelligent agents.

As a matter of course, two underpinning assumptions explicitly based on the notion of granularity can be drawn.

- **Claim 1**: *holon is an agent of a holarchy at a given granularity level.*
- **Claim 2***: viewed at a certain granularity level, holarchy is an agent.*

The first claim is compliant with the traditional holonic literature, i.e., holon as an autonomous whole (agent) being also a part of the holarchy at a certain granularity level; the second claim accounts for a stronger notion of intelligence as a society of agents assuming any level of the holarchy to be as a whole resembling to its intelligent parts. Actually, this interpretation of collective intelligence can be considered close to that of Minsky, as published in the seminal work on AI *Society of Mind* [25].

To better grasp the meaning of our assumptions, it is useful to figure out the holon as playing the role of an entity and the holarchy as playing the role of an information granule. From the two previous claims, it follows that a granule behaves like an entity and groups of entities behave like a granule *for any given granularity level*.

Under this perspective, a holarchy, "sliced" at each granularity level, actually shows a Janus face on each level.

- One face looks from the *inside* of the level according to an enrichment criterion (viewing the whole as made of parts, i.e., entities).

- The other face looks from the *outside* of the level according to an abstraction criterion (viewing parts in terms of a whole, i.e., granule).

Such twofold nature is in full accordance with the traditional holonic-based paradigm [26] and can be expressed in object-oriented UML [27] notation as in Figure 1.

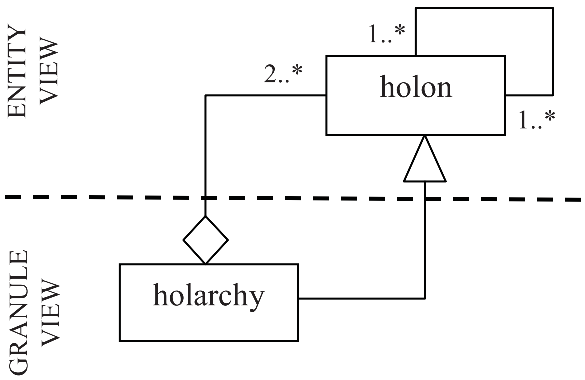

Figure 1. Object-oriented representation of a holonic system at a given granularity level.

According to Claim 2, given a certain granularity level, holarchy (considered as a granule) is also a holon (entity) in the proper sense. The idea is then to enhance the concept of information granule with that of holarchy in order to embed any possible granularity level in it. This way, an entire holonic system spanning through different levels can be defined with a unique computational entity. Hence, a first informal identikit of the holonic granule can be drawn as reported in the following definition.

Def. 2 (holonic granule - informal): *A holonic granule is an information granule showing the properties of a holon as defined by Claim 1 and Claim 2.*

Note that this definition is an extension of that of Zadeh presented previously: it adds the aspect of holon/holarchy which accounts for an inherent architectural recursiveness. It is noteworthy that this aspect, in GrC, is generally not taken explicitly into consideration.

A. Providing a Formal Definition of Holonic Granule

In this paragraph, a more rigorous assessment of the concept of holonic granule is presented.

Def. 3 (holonic granule): *Let E be a set of n elements, let G_E be a set of subsets of E such that the union of their elements gives E and let R_G be a set of binary relationships defined over G_E such that the graph < G_E , R_G > is connected:*

$$E = \{e_1, e_2, \ldots, e_n\}$$

$$G_E = \left\{ g_i = \{e_{i_1}, e_{i_2} \ldots, e_{i_k}\} \subseteq E \mid \bigcup_i g_i = E, k \leq |E| \right\}$$

$$R_G = \{r_j \in G_E \times G_E \mid \langle G_E, R_G \rangle \text{ is a connected graph}\}$$

a holonic granule (HG) is the graph $HG = \langle G_E, R_G \rangle$.

Viewed from the outside, g_i elements represent groups of similar and indistinguishable entities, consequently they are granules in the Zadeh's sense. Mathematically, every g_i is a node of the HG graph and corresponds to an edge of the hyper- graph defined over the elements of set *E*. Nevertheless, considering HG representation in terms of hypergraph can be misleading from an operational point of view. When dealing with multi-level system description, as shown further in the text, it is more practical to consider HG as a recursive object.

In order for a HG to be represented at a given level, we need to define the granules of the next neighbor sub-level. If this sub-level is further enriched, it suffices considering its elements as new HGs at a lower granularity level, thus exploiting HG recursive behavior. HG are then abstract categories that account by default for any given granularity level. In other words, HG concept guarantees recursive decomposition/granulation.

Notice that HG, by definition, can be expressed as a particular type of a class diagram (see Figure 2) where composition relationships are recursively defined over HG class in addition to the relationships defined at each given granularity level.

One could ask why composition rather than aggregation relationship has been considered. The reason is that composition is conceptually strongest than aggregation. Given a certain HG, if we leaved out a sub granule from it we would obtain a different whole, hence a different HG. In other words, composition accounts explicitly for the semantics of the phrase 'a whole is more than the sum of its parts', which is a well-known motto in the holistic thinking community.

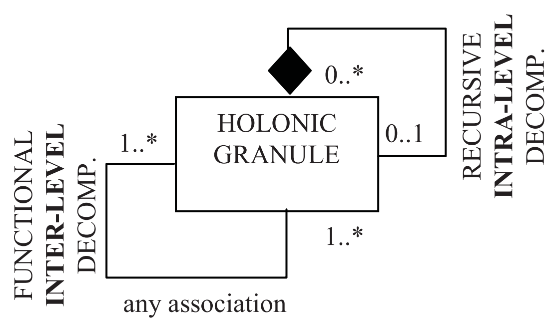

Figure 2. Representation of HG as a UML (abstract) class diagram.

B. Links Between Holonic and Information Granule

It is important to observe that while a recursively-decomposable HG is always an entity (semantically, everything can be defined as to be an entity!), an entity can be either a recursively-decomposable HG or not. If an entity is a recursively-decomposable HG, we are simply defining a new decomposition where what we previously called as entities now become granules thus implying the need to find out some other more specific entity as basic building block. Otherwise, if entity is not a recursively-decomposable HG, this means that, according to our problem representations, system representation 'has reached the ground' and then entities are atomic concept. In this case, we refer to these non-decomposable entities as 'primitive' or 'ground' HGs.

Def. 4 (primitive HG): *a primitive (or ground) HG is any granule at the lowest granularity level.*

From Def. 3 and 4, it appears clear how the concept of primitive HG is the same of information granule in Def. 1. Hence, HG is a proper extension of information granule in the sense of recursion.

C. HG-Based System Description

The use of HG as a modeling instrument is well-suited for handling complex system description at multiple granularity levels; furthermore, it provides slightly a better insight into granularity level understanding in GrC terms.

As it happens in process-oriented analysis where a complex process is decomposed recursively into simpler sub-processes, HG can be recursively self-described by means of an enrichment process until some basic granularity level is reached, i.e., all children HGs are primitive ones.

If HG^L represents a system granule observed at a given granularity level L, its enrichment at a finer granularity level leads to something like this:

$$HG^L = HG_1^{L-1} \cup HG_2^{L-1} ... \cup HG_m^{L-1}$$

It is interesting to note that enrichment is a top-down description pattern aimed at zooming inside system structure. The result of this zoom is the discovery of new entities and relationships, hence new HGs.

We assume that, at any level, decomposition (hence enrichment) is lossless, meaning the inverse composition process (hence abstraction) should give back exactly the original HG.

During HG-based decomposition process, there may be clumpiness among granules at level L-1. These accounts for non-crisp distinctions among HGs at that level. For example, the idea of HG, the core of this work, is mainly presented in this section, but some references to it can be found in other parts of the text as well. A finer granulation, for example at paragraph level, should be undertaken to appraise more crisp boundaries for that concept.

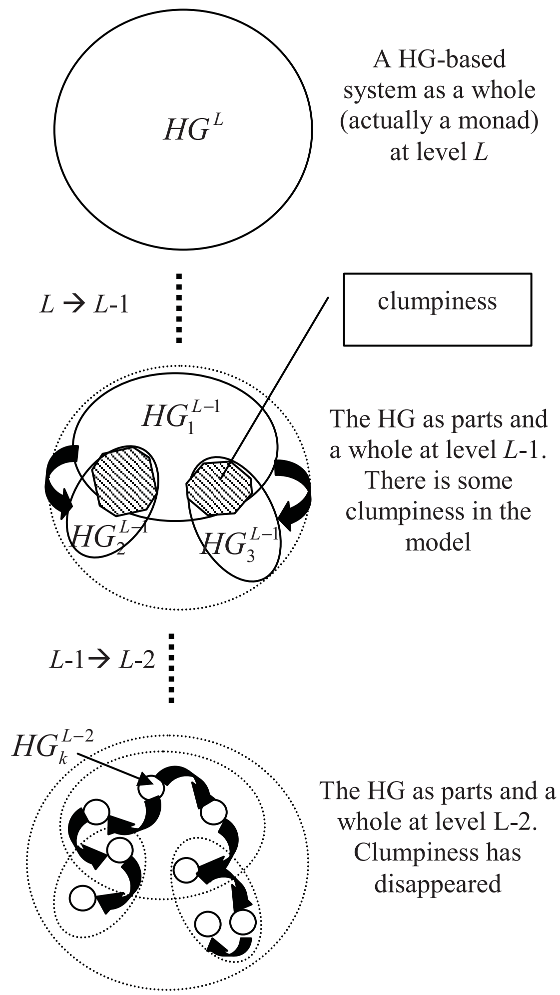

Figure 3. HG-based system decomposition example.

Notice that HG-based enrichment can be iterated until a desired abstraction level is reached, i.e., primitive HGs appear. This actually produces a holarchy structured at different granularity levels. Hence, we define a HG-based system this way:

Def. 4 (HG-based system): *A system is said to be HG-based iff it can be described as a holarchy of HGs, i.e., at multiples granularity levels.*

Figure 3 helps figure out the idea of granularity levels in HG definition by means of an example where three levels are shown. It is therefore clear that HG-based description as a modeling technique can be used to characterize any sort of complex system or process.

D. HG-Based System Description: Inside-the-Box and Outside-the-Box Aspects

According to previous considerations, a HG-based system can be represented architecturally as a holarchy of HGs arranged in a given multi-level structure which depends on the nature of the system under scope. In this regard, two main relationships among HGs can be identified: *inter-granule relationships* and *intra-granule relationships*.

When the HG-based system is described at a given (entity) level, the focus is on inter-granule relationships. In this case, the meaning of inter-granule relationships depends on the semantics of the HG. Alternatively, when the focus in on the mappings between one level and its subsumed components, then intra-granule recursive relationships are being considered. This because a process of conceptual refinement in HG description is being carried on.

The distinction between intra- and inter-granule relationships defines two complementary granular system description approaches: *inside-the-box* and *outside-the-box*.

The former relates to connections among subsystems at the same level, i.e., it describes a whole in terms of its parts; the latter relates to internal sub-system decomposition (mappings from a level to the next one). The two approaches are both needed when a complete HG-based system is studied for modeling a system at multiple granularity levels.

V. Holonic Granules as a Modelling Paradigm

The recursive structure of HG let an entire system be described across multiple granularity levels in a very modular way. HG, the basic building block, is in fact the same conceptual and computational entity for any considered granularity level.

When HG is not further decomposable it means that it accounts for a ground property, which is not in the interest of the modeler to be further analyzed. In general, primitive HGs can be conceived as the equivalent of application programming interfaces (APIs) in software engineering: they are the basic functions on which to built-up the entire program flow *in a given language*.

With this reference in mind, it is possible to look at HG also from a computational perspective. HGs perform computations on data at each granularity level and then roll back the holarchy at the caller HG level to output the processed result. In this regard, with respect to other modeling techniques (e.g., flow diagrams), the use of HGs provides in addition the ability to look through abstract/enrichment patterns with a unique modeling box, thanks to its recursive nature.

The property of handling different granularity levels is of great interest in distributed system modeling. For example, in most pervasive monitoring applications, the use of locally distributed smart devices is not sufficient to guarantee an effective and efficient management alone. The amount of information that comes from scattered sources, actually representing different observation points of the same macro-phenomenon, needs in fact to pass through a number of phases (i.e., pre-processing, validation, elaboration, fusion etc…) determining data to be elaborated at progressively higher abstraction levels.

In the following, two prospective applications of the proposed modeling paradigm dwelling in the fields of industrial automation and environmental monitoring are reported. Their HG-based models are depicted in Figure 4.

A. Test Case 1: Electric Network Management System

Electric power distribution network represents one manifest example of a complex system arranged as a geographical network of electric producers and consumers. Basically, the aim of the network management system is to guarantee, at any given time, that the electric load requested by all customers (being them domestic or industrial) is supplied by, at least, an equal amount of electric power produced by electric plants connected to the network.

A lot of equipments are necessary for the process to be satisfied. They are encountered by the electric flow from the power plant to the customer's meter socket, namely: extra high (i.e., 800kV), high (i.e., 220kV) or medium (i.e., 33kV) voltage transmission lines (generally aerial), transmission towers, transformer stations and substations, low voltage distribution lines, relais, inverters and other electrical stuff.

The architecture of wide electric networks (covering the needs of entire states and countries) is always hierarchic since at least two levels can be identified:

- <u>Transmission level</u>: it accounts for the bulk transmission of electrical energy from generating power plants to substations located near to population centers. These levels can be further subdivided into two sub-levels: one for extra-high voltage transmission and the other for high voltage transmission. In both cases, a three phase alternating current is used to cover very long distances (hundreds or miles) minimizing the energy loss in transmission. Generally, interconnections at this level are redundant hence having the form of a grid so that electric energy has more possible routes to flow if some failures occur.

- <u>Distribution level</u>**:** it covers more limited areas (urban or rural). It is made of local wiring between high voltage substations and customers and can be schematized as a bus with different kind of loads attached. It is noteworthy that small energy production sources (such as city power plants, solar or wind farms) are generally connected at this level.

Architecturally, a HG-based representation of the electric network can be hence drawn with the following simple assumptions:

- A **HG** is any information processing object placed at the junction node between two transformer stations.

- A **primitive HG** is any basic element of the network such as electric energy power producers/consumers (between a transformer station and the ground) and transmission/distribution lines.

- Transformer station represents the **interface** between two or more HGs.
- An overarching layer representing the whole electric network is introduced as **super-HG** subsuming the entire holarchy. By spanning the whole holarchy from top to bottom, the super-HG is recursively decomposed until primitive HGs are found.

The electric power measured at any HG is equal to the algebraic sum of all the loads and the produced energy power of the subsumed HGs. In normal situations the sum is zero according to the Kirkchhoff's law; if the energy power demand cannot be satisfied by an equal amount of produced energy offer a drop in current affects the underlying holarchy and a blackout occurs. An intelligent energy distribution management system can be then imagined as operating at the HG level to prevent local failures, optimize network load balance, minimize production costs, etc.

B. Test Case 2: Air Quality Monitoring Systems

Typical examples of indoor Air Quality Monitoring (AQI) applications are big civilian buildings and industrial environments. These are composed of areas characterized by strong micro-climatic heterogeneities (i.e., different rooms, area around different machineries, etc.). For monitoring purposes, a dense sensor network can be accustomed to analyze each micro-climatic area and raise appropriate alarms triggered by potentially hazardous situations.

Suppose to deal with the problem of designing an AQI system by means of a HG-based approach. One of the advantages of such a choice is that it is possible to add as many levels as one wants without affecting the modularity of the basic holon. The lowest level (Level 0, made of primitive HGs) is composed of all the sensors, alarms and human-computer interfaces of the network. Here data is spot, namely, it is referred to specific monitored locations. Consequently, a number of overarching levels can be added for information processing purpose. In this example, three other levels are introduced (refer to [28] for a more in-depth description).

Level 1. The main tasks of the HGs at this level are:

- Data pre-processing: raw data coming from in-field sensors are sampled and validated.
- Information publishing: the user is visually given a local description of each monitored parameters.

Level 2. The second level of the architecture works at a higher granularity level. The monitored building is considered as divided in various regions. A region is composed of several neighboring locations. An HG is assigned to each region. This HG works in cooperation with several neighboring HGs standing in the previous level; it also handles information referred to a whole region of the monitored environment. At this level, each HG has the following tasks:

- Area modeling: the aim is to build a model about the daily evolution of the monitored parameters in the region under scope.
- Spatial validation: this is a further level of validation implemented in this system. Data sampled from various nodes are compared. If the monitored situation is compatible with the model then data are labeled with a high reliability coefficient, otherwise the reliability coefficient is reduced.
- Alarm management: when critical conditions are detected, the system is able to raise various levels of alarm, according to the criticality of the event. Using area modeling and spatial validation, it is possible to infer if a given situation is due to a sporadic local event or to a phenomenon that is interesting a wider area.

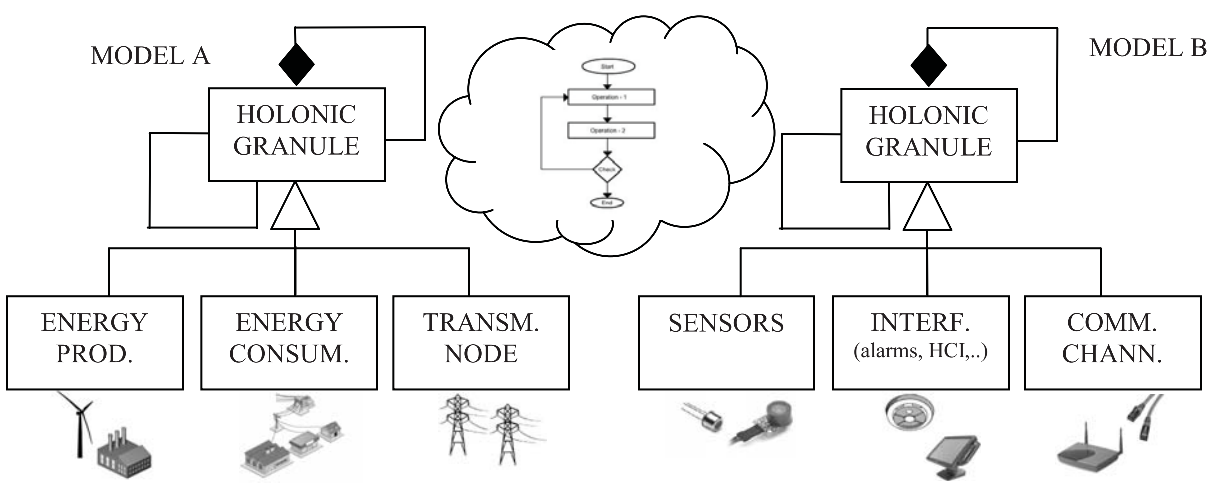

Figure 4. HG-based models (A and B) for the provided examples: electric network management system (left) and AQI system (right).

- Information publishing: the interface shows average information about the whole region. When a local critical condition is detected, the interface passes the control to the HG at the lower level in order to show local detailed data about the event under analysis.

Level 3. The third level of the architecture implements the same functions of Level 2 but works at the maximum possible granularity. Each task is referred to the entire monitored structure. The alarm management function for example is used to handle those critical events involving more than one regions of the monitored structure. Likewise, the user interface shows the same kind of information but referred to the whole monitored environment.

VI. CONCLUSION

Holonic models are natively suited for describing granular structures; however, current GrC approaches seem to dismiss such connection.

In this work, a novel software modeling approach, dealing with granular systems by means of the new concept of holonic granule (HG), has been introduced. Based on the object-oriented paradigm, the proposed model is rather simple and intuitive. The holonic model and the holarchy allow for describing the structure of the information processing and actions: within this structure we introduced an holonic perspective also for information and knowledge by proposing the holonic granules, as a granular hierarchical aggregation of information and knowledge at different abstractions levels.

Examples of HG-based models have been also presented for prospective applications in the field of industrial automation and environmental monitoring.

The proposed approach can be placed and expanded in the wider field of GrC and CWW. By definition, HGs provide an abstract framework for dealing with (granular) entities that can be aggregated into higher-level entities or decomposed into lower-level entities maintaining always the same computational structure. In this sense, HGs can be an 'alphabet' in which to inflect GrC and CWW ideas. Hopefully, the proposed technique represents a first step in this direction.

REFERENCES

[1] A. Koestler, The Ghost in the Machine, (1st Edition) Hutchinson, London, 1967.

[2] P. Kopacek, "Intelligent Manufacturing: Present State and Future Trends", Journal of Intelligent & Robotic Systems, Springer, 1999, vol. 26(3-4), pp. 217-229.

[3] W. Pedrycz, "Computational intelligence as an emerging paradigm of software engineering", Proc. of the 14th international conference on Software engineering and knowledge engineering, 2002, pp. 7-14.

[4] J. Han, J. Dong, Perspectives of Granular Computing in Software Engineering, Proc. of the IEEE International Conference on Granular Computing, 2007, pp. 66-71.

[5] D. Thompson, D.R. Hughes, "Holonic Modelling, Manufacturing Engineer", vol. 77(3), 1998, pp. 116-119

[6] M. Ulieru, R. Doursat, "Emergent Engineering: A Radical Paradigm Shift", ACM Transactions on Autonomous and Adaptive Systems (submitted April 2010).

[7] B.T. Clegg, D. Shaw, "Using process-oriented holonic (PrOH) modelling to increase understanding of information systems", Information Systems Journal, 2008, vol. 18, pp. 447-477.

[8] S. Robinson, Simulation: The Practice of Model Development and Use, John Wiley and Sons, Chichester, UK, 2003.

[9] A. Greasley, Simulation Modelling for Business, Ashgate Press, London, UK, 2004.

[10] J.S. Valacich, J.F. George, J.A. Hoffer, Essentials of Systems Analysis and Design, 3rd edn. Prentice Hall, Upper Saddle River, NJ, USA, 2006.

[11] G.S. Gardiner, M.J. Gregory, "An audit based approach to the analysis, redesign and continuing assessment of a new product introduction system", Integrated Manufacturing Systems, 1996, vol. 7, pp. 52–59.

[12] M.C. Jackson, P. Keys (1984), "Towards a system of system methodologies", Journal of Operations Research, vol. 35, pp. 473–486.

[13] J.R. Hobbs, "Granularity", Proc. of the 9th International Joint Conference on Artificial Intelligence, 1985, pp. 432-435

[14] Z. Pawlak, "Rough sets", International Journal of Computer and Information Sciences, 1982, vol. 11, pp. 341-356.

[15] B. Chen, M. Sun, M. Zhou, "Granular Rough Theory: A representation semantics oriented theory of roughness", Applied Soft Computing, 2009, vol. 9(2), pp. 786-805.

[16] L. A. Zadeh, "Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems", Springer-Verlag Soft Computing , 1998, vol. 2, pp. 23-25.

[17] L.A. Zadeh, "Fuzzy sets and information granularity", in: Advances in Fuzzy Set Theory and Applications, Gupta, N., Ragade, R. and Yager, R. (Eds.), Amsterdam: North-Holland, 1979, pp.: 3-18.

[18] L.A. Zadeh, "Fuzzy logic = computing with words", IEEE Transactions on Fuzzy Systems, 1996, vol. 4(2), pp. 103-111.

[19] Zadeh LA, "Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic", Fuzzy Sets and Systems, 1997, vol. 90, pp. 111-127.

[20] J. Mendel, L. Zadeh, E. Trillas, R. Yager, J. Lawry, H. Hagras, S. Guadarrama, " What Computing with Words Means to Me", IEEE Computational Intelligence Magazine, 2010, vol. 5(1), pp. 20-26.

[21] T. Y. Lin, "From Rough Sets and Neighborhood Systems to Information Granulation and Computing in Words", Proc. of European Congress on Intelligent Techniques and Soft Computing, 1997, pp. 1602-1607.

[22] W. Pedrycz, "Granular computing: an introduction", Proc. of the Joint 9th IFSA World Congress and 20th NAFIPS International Conference, 2001, pp. 1349-1354.

[23] S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 2nd ed., Prentice Hall, 2003.

[24] A. Giret, V. Botti, "Holons and agents", Journal of Intelligent Manufacturing, 2004, vol. 15, pp. 645-659.

[25] M. Minsky, The Society of Mind, Simon and Schuster, New York, 1998.

[26] M. Ulieru, R. W. Brennan, S.S. Walker, "The holonic enterprise: a model for Internet-enabled global manufacturing supply chain and workflow management", Integrated Manufacturing Systems, 2002, vol. 13(8), pp. 538-550.

[27] Object Management Group, Documents associated with Architecture-Driven Modernization (ADM)/ Knowledge Discovery Meta-Model (KDM), v1.0, http://www.omg.org/spec/KDM/1.0/, 2007.

[28] M. Calabrese, A. Amato, V. Di Lecce, V. Piuri, "Hierarchical-granularity holonic modelling", Journal of Ambient Intelligence and Humanized Computing , Springer, 2010, vol. 1(3), pp. 199.