On Metabolic Complex Networks for Entropic Robust Autonomy

Chris Nowak
Defence Science & Technology Group
Sydney, Australia
e-mail: chris.nowak.5@gmail.com

Darryn Reid
Defence Science & Technology Group
Adelaide, Australia
e-mail: darryn.reid@dst.defence.gov.au

***Abstract*—Autonomous systems are expected to be robust: they should be resilient to perturbations arising both from the external environment and from within the system itself. In other words, they should maintain a state of dynamic equilibrium, or homeostasis, within known limitations. By framing autonomous systems as metabolic systems, which can be understood as systems with flows and transformations that are capable of functioning well in a complex environment, it is demonstrated how a homeostatic control mechanism could be designed to enable such systems to self-adapt to the changing environment. To illustrate these ideas, they are applied to the problem of designing robust autonomous supply networks; their homeostatic control mechanisms and catalysts are identified.**

Keywords–Robust Autonomy; Resilience to Perturbations; Metabolism; Homeostasis & Allostasis; Transformations; Capability, Function & Behaviour; Self-adaptation; Curvature; Entropy.

I. INTRODUCTION

This paper is concerned with achieving robustness of autonomous systems, meaning the ability to absorb perturbations arising both exogenously—from the environment—and endogenously—from the system itself—within known limits. Using supply networks as an example for analysis, we frame the problem of robustness as "metabolic" complex networks to obtain a set of homeostatic control mechanisms. Illustrating the generality of this approach, this set of homeostatic controls should be capable of scaling irrespective of the complexity of any supply network instance.

Complex networks [1][2] are understood as networks that exhibit *complex behaviours*; complex networks are neither *regular,* nor *random*—and these networks behave in interesting (not regular, not random) ways. The real world, both natural and man-made, *creates* complex networks—and they either behave and function well, or get replaced by new versions which work better. The Internet is a complex network where new computers connect themselves (neither regularly, nor randomly) to the already existing network. It is debatable whether the Internet is a *scale-free network* [3], but the Internet does build itself up using the *preferential attachment* [2] mechanism (used to explain scale-free networks): new nodes prefer to attach themselves to *hubs*, i.e., nodes which are already well connected. Naturally, networks with hubs form. And they have one important advantage of being resilient to random attacks, and one important disadvantage of being vulnerable to targeted attacks. The following question presents itself: how can networks function well?

The question of functioning well implies a range of more specific questions. How is the network supposed to function? Can we measure how well it is functioning now? Can we predict how well it will be functioning in the future? Does the network need to be adaptive, situation-aware (of the environment and self), autonomous? Should the network be self-modifying, and given the library of *plans* that the network has, should it be able to modify the plans, or create new ones? But also: is it that the ability to manage and execute plans is the most important capability of the network? This paper suggests that it is *not.*

This paper suggests that *plans* are *secondary*—and *perturbations* are *primary concerns.* The functioning of a biological organism is primarily about handling perturbations, such as changes of temperature, sugar or pH level—it is the perturbations which dictate which plans are to be selected and executed. It is claimed here that complex systems with non-trivial behaviours should be *metabolic*: they should detect perturbations and employ *homeostatic control mechanisms* in order to maintain a *dynamic equilibrium* state (in which they can function well). The recommended "metabolic perspective" facilitates a conceptual shift from "plans first" to "perturbations first" and indicates that a complex system should: detect and measure perturbations, evaluate their effects on the way the system itself functions and whether it has been pushed out of the equilibrium state, and select and employ appropriate homeostatic (or allostatic) control mechanisms to return to the previous stable state (or move to a new stable state, respectively). This approach suggests that, given a complex system, to *understand* the system we should focus not on plans (or mechanisms of functioning), but on how the system handles perturbations and maintains homeostasis—because not only biological organisms do that, but so do infrastructure systems, cities, ecosystems, and a variety of other complex systems.

Given this stance, the long term goal is to analyse complex networks (which are discrete models of real-world systems) from the metabolic perspective by mapping perturbations to homeostatic and allostatic control mechanisms that can execute plans ensuring that the dynamic equilibrium states are maintained (as in these states the networks function well). Such a research program would be applicable to many complex systems domains—it just adds the metabolic perspective to the complex networks research program. Given that our models would be complex networks, both problems and solutions could be formulated in abstract ways, namely in terms of network topology, node and link attributes, and flows (all of

 ISBN: 978-1-61208-787-0

them dynamic). One would expect that multiple real-world problems would share the same abstract problem and its solution. It is also important to be able to *measure* how well the given network functions; recent complex network research indicates that *curvature* based measures [4]–[5] seem to provide appropriate and powerful tools. The framework requires a conceptual—or ontological—shift; the ontology sketched in Section VII (and Appendix B) provides a step in this direction. The contribution of this paper is mostly conceptual: it proposes an ontology emphasising perturbations and homeostasis, it provides a discussion on measuring the health of networks, and it employs a supply network example to demonstrate the applicability of metabolic notions to non-biological systems. It should be noted that socio-technical systems have been seen as "metabolic systems" before, for instance in the case of urban systems [6][7]—however, metabolism there is being associated with the management of resource flows, and not explicitly with perturbations and homeostatic control.

A discussion on robust autonomy is presented in Section II, with *robustness* understood as *resilience to perturbations.* Section III considers communities, health and functions, indicating that, in a simple case, a community can be healthy and functioning well when it is well-connected. In such a case, it is also clear what the meaning and nature of *network curvature* is: well connected networks (such as cliques) have positive curvature, while other networks (such as trees) have negative curvatures, and, hence, making the network better connected would increase its curvature [8][5]. *Metabolic Complex Networks (MCN)* are described in Section IV, starting with metabolic cycles, and extending to metabolic pathways (networks of connecting transformations) which explain what functions the system can perform. Subsection IV-A focuses on two component systems, analysing them from two perspectives: firstly, by associating the *health* of the network (understood as its ability to function well) with the number of arrows connecting the network's nodes, and secondly, by considering the presence of *sinks* and *sources* as *attributes* of the network (and using the Formal Concept Analysis (FCA) framework [9] to construct a lattice ordering on networks). Then, a supply network scenario is analysed in Section V, and evacuation scenarios are (briefly) discussed in Section VI. The importance of Section VI comes from the fact that an evacuation network should be seen as composed of multiple *interdependent* supply networks (and, in general, networks can consist of many *interdependent* networks). Ontologies for robotics are considered in Section VII; this section explicitly states that Dennett's *intentional stance* [10] should be extended by adding the *metabolic level* above Dennet's physical, design and intentional levels. Further work is described in Section VIII and Section IX concludes this work.

II. Robust autonomy

The problem of robustness for autonomous systems as addressed herein consists of the following features:

- an *autonomous system* ω has an (overall) *capability* c_ω to perform its (overall) *function* f_ω; we will say that the system is *functioning* when it is performing its function f_ω;
- for the system to be functioning, it must maintain its dynamic equilibrium, i.e., *homeostasis* or *allostasis,* because outside of the equilibrium state the system's functioning is either difficult or impossible.

Thus, we distinguish two fundamental behavioural regimes, homeostasis and allostasis; the ability to manifest both with respect to various kinds of perturbations yields the desired robustness, while the limitations of allostasis provide explicit bounds on robustness:

- homeostasis, when understood in a less restrictive way, allows moving to an *alternative* dynamic equilibrium state rather than returning to the original equilibrium state; this is referred to as *allostasis;*
- *homeostatic and allostatic control mechanisms* allow the system—in the presence of perturbations—to continue performing its original function, or switch to performing an alternative function, respectively;
- homeostatic/allostatic control mechanisms are triggered by *perturbations;* however, it is beneficial for the system to *detect* the perturbations as early as possible, and even to predict the potential for perturbations of various kinds; perturbations need to be *handled,* i.e., *processed* after being detected or predicted.

Regarding systems' *capabilities,* we distinguish the following:

1) capability to perform normal functions;
2) capability to maintain homeostasis and allostasis;
3) capability to handle perturbations;
4) capability to handle plans.

The above constitutes a description of *robust autonomy* as "functioning + handling perturbations." Note that plan monitoring (related to performing function f_ω, with possible slight *homeostatic* variations) and plan modifying (switching from $f_\omega^{'} = f_\omega$ to an *allostatic* $f_\omega^{''} \neq f_\omega$) can be continuously performed while executing functions related to capabilities (1–3); therefore, (4) can be seen as a meta-level homeostatic/allostatic control mechanism. [Plan monitoring is not the focus here.]

It is also of importance to assess how well systems function. One way of achieving this is through *curvature-based methods* [11], cf. Section III. Robustness, curvature and entropy have been linked [4], and so entropic curvature-based measures could be applied to assess systems' *robustness.* Hence, when entropy (and related notions) are applied to autonomous systems, we could talk about *Entropic Robust Autonomy (ERA).*

We suggest introducing entropy and related concepts along the following lines. The assessment of the robustness of a system in terms of its ability within known limits to absorb various kinds of perturbations implies the need for an order parameter or set of order parameters: an order parameter provides a mechanism of abstraction from the myriad of details and thereby yields distinction between different modes in the environment and, correspondingly, different regimes of autonomous system behaviour. Here, we are concerned especially with distinguishing homeostatic and allostatic behavioural regimes in response to various kinds of perturbation. Entropy and related notions provide order parameters that promise to provide robust stable measures or estimates to determine these change points. Moreover, such estimates do not imply detailed predictions about system or environment state evolution; as a means of abstraction, we may rely on

 ISBN: 978-1-61208-787-0

predictions about bounds on overall behaviour. Note that it is argued in [12] that the heart of the problem with autonomy is its need to deal with *uncertainty*.

III. COMMUNITIES, HEALTH AND FUNCTIONS

If a network represents a society—or a social group—then one might want to detect communities within the society, where the communities are strongly connected subgroups. Communities are often being detected in order to perform *sentiment analysis* [8]; however, we can also associate communities with *functionalities*. If a society (a social group) is to perform some *functions,* it might need to delegate sub-functions to specialised communities (sub-groups of the whole social group). Given that communities are strongly connected sub-groups, we can associate “strong connectedness” with “health” and say that what is being detected are “healthy sub-groups.” Then, forming healthy communities is a part of maintaining a healthy society, where “health” can be understood as “capability”—communities have capabilities to perform their functions, in this way building up the total capability of the society. Topological graph theory based methods allow us to detect communities (assess sub-groups’ health), but entropic curvature-based (geometric) methods are also used [5][8] (figures presented in [8] show how the geometric (curvature) transformation of the network can be performed). It should be noted that a community might start to deteriorate by losing connections, which could create *sink-only* and *source-only* nodes—this leads to the analysis of Section IV-A.

Using an entropic measure essentially means mapping the raw system model with all of its full complexities into a different, smaller model, where what comes out are regimes of behaviour rather than masses of possible behaviour. That is, the simplification is through functional abstraction, rather than component abstraction by merely grouping components together. In essence, this is what a method of using the entropic curvatures would give us here: the abstraction amounts to a mapping between the problem space into a smaller one whereby the many details are collapsed down into the regimes of behaviour with respect to the kinds of perturbations the environment or the autonomous system itself can impose on the autonomous system.

IV. METABOLIC COMPLEX NETWORKS

Complex networks are networks that are neither regular, nor random [1]–[3][14][15]. *Metabolic networks* are networks that can handle metabolic flows; Figure 1 shows some biological metabolic pathways [13][16].

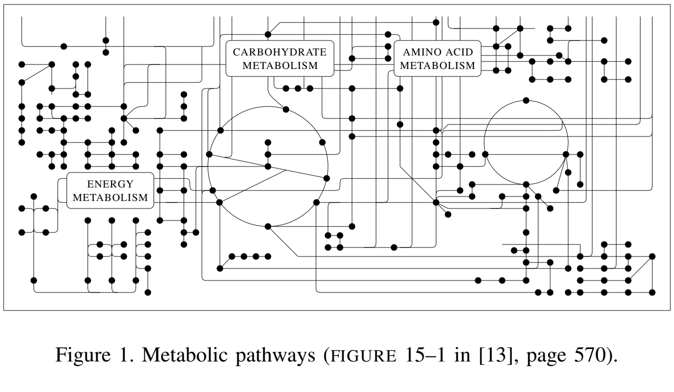


Figure 1. Metabolic pathways (FIGURE 15–1 in [13], page 570).

In a simple case, the system accepts an *input pattern* and *transforms* it to an *output pattern.* For instance, Figure 2 shows a *metabolic cycle:* a system that accepts the recurring (a, b, c, d) *input patterns* and *transforms* them—using the appropriate recurring patterns (t_1, t_2, t_3, t_4) of *transformations*—into the recurring *output patterns* (e, f, g, h). Metabolic cycles are simple examples of metabolic pathways which are, in general, sequences of connecting transformations (in the case of biological systems, transformations can take the form of *chemical reactions*). Figure 1 shows the *metabolic pathways* of a eukaryotic cell (further information on *pathways* can be found in [13] (FIGURE 15–1), [16] and on the websites [17]–[18]).

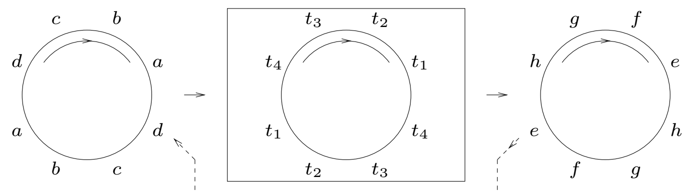


Figure 2. Metabolic cycles.

Assuming that a metabolic system ω is a complex network, the following sets need to be considered:

Θ	nodes	(they accept inputs and produce outputs)
I	inputs	(they are accepted by nodes)
O	outputs	(they are produced by nodes)
Δ	transformations	(transf. a node’s inputs to outputs)
Φ	flows	(a flow is an output connecting to an input)
Σ	streams	(selections of metabolic inputs/outputs)
Υ	pathways	(flow sequences associated with functions)
Γ	catalysts	(facilitate transformations; not consumed)
Ψ	homeostatic control mechanisms	(ensure stability)
Λ	allostatic control	(alternative equilibrium states)
Ξ	interdependent networks	(sources of perturbations)
Π	perturbations	(coming from env. or other systems)

Hence, if we analysed a metabolic network ω, we would analyse it as $\omega(\Theta, I, O, \Delta, \Phi, \Sigma, \Upsilon, \Gamma, \Psi, \Lambda, \Xi, \Pi)$—which is more than $G = (V, E) = (\Theta, \Phi)$—i.e., we would analyse its: nodes, inputs, outputs, transformations, flows, streams, pathways, catalysts, homeostatic and allostatic control mechanisms, related interdependent networks, and perturbations. The ultimate goal of *metabolic analysis* is to understand ω as a *metabolic system* (with the elements just listed) capable of *functioning* and *handling perturbations* by *self-adapting* to the environment. We are interested in knowing the limits of self-adaptation, where we expect to be able to characterise these limits in terms of the appropriate entropy measure.

Recall that a node has inputs and outputs; a single input is accepted by a specific node; a single output is produced by a specific node; a node transforms its inputs into its outputs (and, therefore, a transformation (of a node) is an (inputs, outputs (of the node)) pair); a single flow (from ϑ_a to ϑ_b) is a single output (from ϑ_a) connecting to (or “becoming”) a single input (to ϑ_b). Further, catalysts are expected to be associated with homeostatic and allostatic control mechanisms; homeostatic control mechanisms ensure that the system returns to its stable state; allostatic control mechanisms move the system to stable but possibly new states. Further, a stream can be an arbitrary selection of inputs and outputs, but would normally be considered as a starting point to a pathway investigation; pathways are the goals of the analysis—they identify functionalities and ensure functioning (corresponding to the identified functionalities) while maintaining a state of dynamic equilibrium and, therefore, *functioning while handling perturbations.* Finally, (external) perturbations come from the system's environment, i.e., from systems external to the system in question (but internal perturbations are also possible). As each perturbation must have its *source,* we claim that the sources of the perturbations are other networks, namely *interdependent networks*.

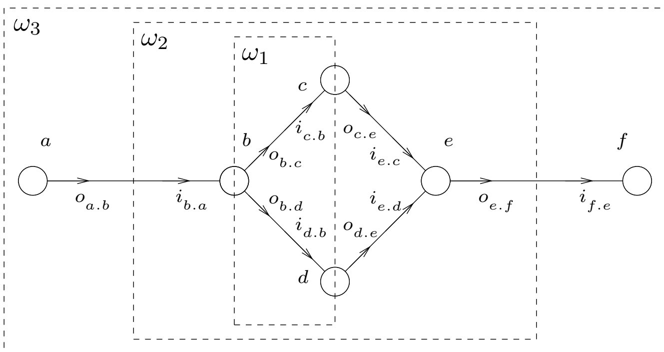


Figure 3. Metabolic systems ω_1, ω_2 and ω_3.

Figure 3 shows systems ω_1, ω_2 and ω_3. We can list these systems' nodes, inputs and outputs: $\omega_3(\Theta_3, I_3, O_3) = \omega_3(\{a, b, c, d, e, f\},\ \{i_{b.a}, i_{c.b}, i_{d.b}, i_{e.c}, i_{e.d}, i_{f.e}\},\ \{o_{a.b}, o_{b.c}, o_{b.d}, o_{c.e}, o_{d.e}, o_{e.f}\})$; $\omega_2(\Theta_2, I_2, O_2) = \omega_2(\{b, c, d, e\}, \{i_b, i_{c.b}, i_{d.b}, i_{e.c}, i_{e.d}\}, \{o_{b.c}, o_{b.d}, o_{c.e}, o_{d.e}, o_e\})$; $\omega_1(\Theta_1, I_1, O_1) = \omega_1(\{b, c, d\}, \{i_{c.b}, i_{d.b}\}, \{o_{b.c}, o_{b.d}\})$. System ω_3 is isolated: it has neither inputs, nor outputs connecting it to the environment (or systems immersed in the environment); all inputs and outputs of ω_3 are internal. We also note that node a is a *source-only* node, and that node f is a *sink-only* node. But ω_3 has *flows* and *transformations:* a flow from node a to node b, denoted $\varphi_{a.b}$, is the output $o_{a.b}$ connecting to the input $i_{b.a}$, i.e., $\varphi_{a.b} = o_{a.b}.i_{b.a}$; and a transformation of node b, denoted δ_b, is the pair, with the first element of the pair being the inputs I_b of b and the second element being the output O_b of b; given that $I_b = \{i_{b.a}\}$ and $O_b = \{o_{b.c}, o_{b,d}\}$ we have that $\delta_b = I_b.O_b = \{i_{b.a}\}.\{o_{b.c}, o_{b,d}\}$. Regarding system ω_2 (which has, comparing to ω_3, lost some nodes and their inputs and outputs), it has one input (to ω_2) from the environment, namely i_b (rather than $i_{b.a}$, as we have dropped node a), and one ouput (from ω_2) to the environment, namely o_e (rather than $o_{e,f}$, as we have dropped node f). This allows us to say that the *stream* flowing through ω_2 is the input i_b (to ω_2) transformed (by ω_2) to the output o_e (from ω_2). Regarding system ω_1, it has lost (comparing to ω_2) only one node e, but we set ω_1 boundaries in such a way that we only consider flows from b to c and from b to d—consequently, we consider neither inputs to b, nor outputs from c or d, and thus also do not consider the nodes' transformations. We will analyse ω_1 network—as a *supply network*—in Section V.

A. Two component systems

This section considers a system with two components A and B; in the environment, there are also other systems C, D, E and F which can connect to A and B, cf. Figure 4. If all the arrows shown in Figure 4 are present, then the binary code for this network will consist of six 1s representing that all six arrows j_1–j_6 are present; the code for this network can be found in row (1), column (a) of Table I. [It should be noted that neither *one component systems,* nor *three component systems* are considered here. The case of a one component system is trivial, as there would only be two possible arrows and four (2^2) possible networks: a system that receives an input from the environment and produces an output to the environment, a system that is a sink only system, a system that is a source only system, and an isolated system (without any arrows). The case of a three component systems would require considering twelve arrows (six arrows between the three nodes, and two arrows between each of the three nodes and the environment) and 2^{12} possible networks (although multiple networks would be have the same topology). Although such three (or more) component networks could be analysed in an analogous way to two component systems discussed in this section, a multi-component system would probably be initially partitioned into two subsystems (and therefore treated as a two component system), with simple interdependence between the components. For a specific multi-component network, the connectivity between the components could be simple, and given the network, in many cases there would only be a limited number of alternative topologies to which the network could transform. Multiple case studies should allow building a library of functioning complex networks, with different real world networks sharing the same abstract model; for instance, a particular food distribution network could be functioning in an exactly the same way as a particular information distribution network.]

Different connection topologies are determined by subsets of the set (of arrows) $J = \{j_1, \ldots, j_6\}$. Table I shows *binary codes* (indicating presence and absence of arrows in the given network) for systems with different subsets of J (and Figure 5 shows the corresponding systems). It is straightforward to order the resulting systems by the subset relation on the sets of directed edges present in the systems' topologies.

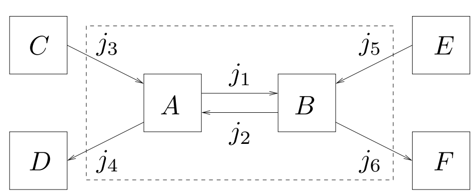


Figure 4. $S_{a_1} = S_{111111}$ with code $c_{a_1} = 111111$ indicating $\{j_1, \ldots, j_6\}$.

Table I consists of cells $(a, 1), \ldots, (h, 6)$ corresponding to the systems of Figure 5 and, therefore, these systems could be named $a_1, \ldots, h_6$. Hence, in Figure 5, the top system could be labelled a_1, the bottom system could be labelled h_6, and so on. Only some of these labels are used in Figure 5, but all labels can easily be derived from the correspondence between the nodes of Figure 5 and the cells of Table I (we will also use

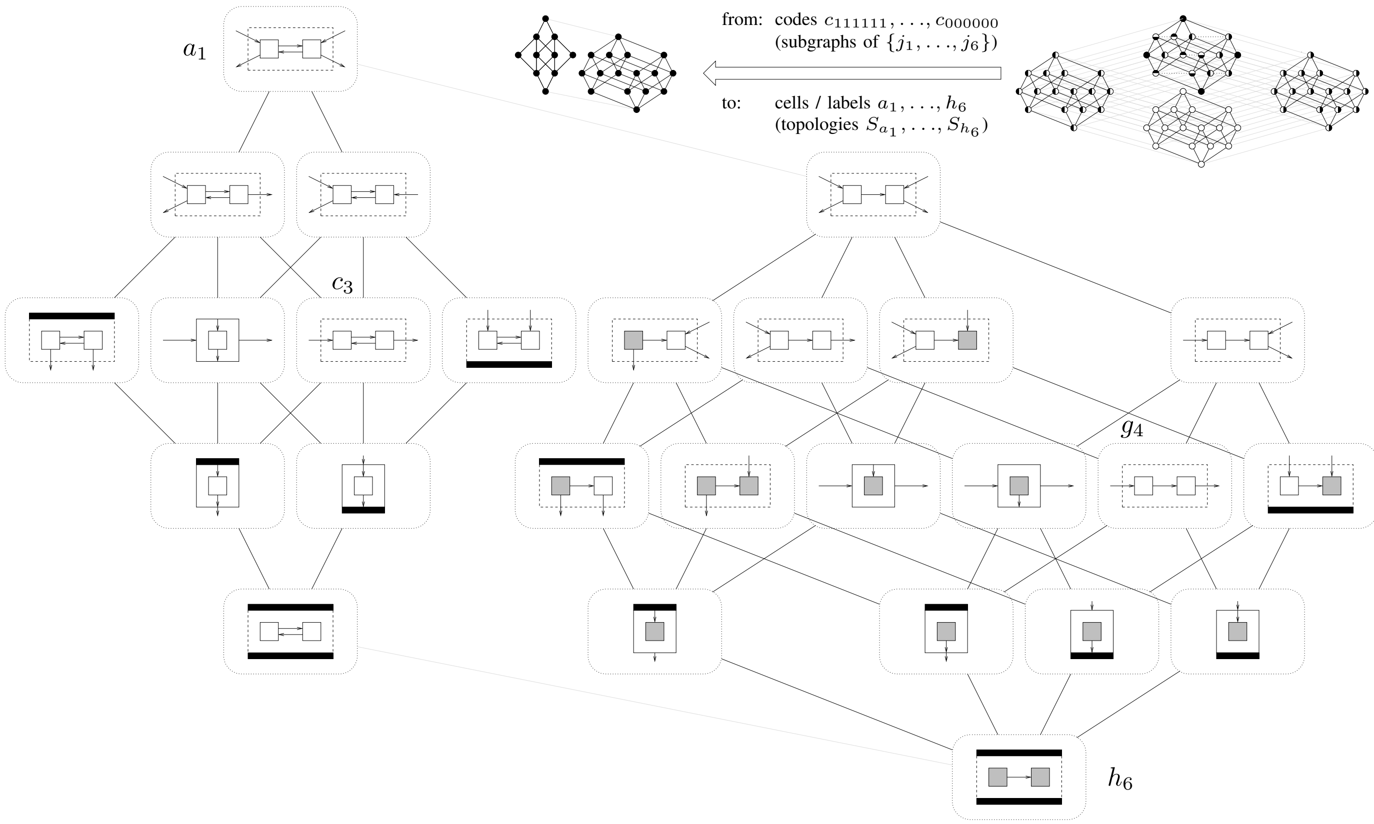


Figure 5. Ordering on $\boldsymbol{S}_J = \{S_{a_1}, \ldots, S_{h_6}\}$ with codes of Table I (the systems' labels—only a_1, c_3, g_4, h_6 are shown here—can be derived from Table I).

these labels for the systems presented in Figure 6). The ordered set $(\mathcal{P}(J), \subseteq))$ induces, in an obvious way, the order on $\boldsymbol{S}_J = \{S_{a_1}, \ldots, S_{h_6}\}$, as shown in Figure 5. It is the ordering of Figure 5 which places, for instance, the network a_1 (the best connected network which has the code 111111 indicating that all arrows are present) at the top of the ordered set, and the network h_6 (the least connected network which has two codes 100000 and 010000 indicating that the network topology is the topology of exactly one arrow between the components A and B of Figure 4; the topology with an arrow from A to B is equivalent to the topology with an arrow from B to A) at the bottom of the ordered set. This is why the code 111111 has been placed at the topmost row (row 1) of Table I and the codes 100000 and 010000 have been placed at the bottom-most row (row 6—or more precisely, rows 6a and 6b): it is the ordering of Figure 5 that has been used when placing the network codes in Table I.

$\boldsymbol{S}_J$ is the set of all posssible *connection topologies* (for a two component system). An alternative way of ordering $\boldsymbol{S}_J$ can be produced if we treat the systems in $\boldsymbol{S}_J$ as "objects" and consider some "properties" these systems have, rejecting the idea (which induced the ordering of Figure 5) that adding arrows makes the network "better" (cf. the paragraph below which discusses $g_4\uparrow$). Using the framework of *Formal Concept Analysis (FCA)* [9], we can form an *FCA context* K in which the systems in $\boldsymbol{S}_J$ are *FCA objects*, and some properties are used as *FCA properties*. Note that the FCA objects are (listing all FCA object): $\boldsymbol{S}_J = \{S_{a_1}, S_{a_2}, S_{b_2}, S_{h_2}, S_{a_3}, S_{b_3}, S_{c_3}, S_{d_3}, S_{e_3}, S_{f_3}, S_{g_3}, S_{h_3}, S_{a_4},$

TABLE I. CODES $111111, \ldots, 010000$ AND CELLS (LABELS) $a_1, \ldots, h_6$.

	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
(1)	11 11 11							
(2a)	11 11 01	11 11 10						10 11 11
(2b)	11 01 11	11 10 11						01 11 11
(3a)	11 01 01	11 11 00	11 10 01	11 10 10	10 01 11	10 11 01	10 11 10	10 10 11
(3b)		11 00 11	11 01 10		01 11 01	01 01 11	01 10 11	01 11 10
(4a)	11 01 00	11 10 00	10 01 01	10 01 10	10 11 00	10 00 11	10 10 01	10 10 10
(4b)	11 00 01	11 00 10	01 01 01	01 10 01	01 00 11	01 11 00	01 01 10	01 10 10
(5a)	11 00 00				10 01 00	10 00 01	10 00 10	10 10 00
(5b)					01 00 01	01 01 00	01 10 00	01 00 10
(6a)								10 00 00
(6b)								01 00 00

$S_{b_4}, S_{c_4}, S_{d_4}, S_{e_4}, S_{f_4}, S_{g_4}, S_{h_4}, S_{a_5}, S_{e_5}, S_{f_5}, S_{g_5}, S_{h_5}, S_{h_6}\}$, i.e., $\mathrm{card}(\boldsymbol{S}_J) = 26$. We need to determine what properties of systems in $\boldsymbol{S}_J$ should be considered.

We introduce the following definitions (note that we assume that there is at least one connection between the (two) components of the system).

Definition 1: (sink/source systems)
Let M be a two component system,

- if a component of M has only inputs, then this component is a *sink-only* system;
- if a component of M has only outputs, then this component is a *source-only* system;
- if M has a component that is a sink-only system, then

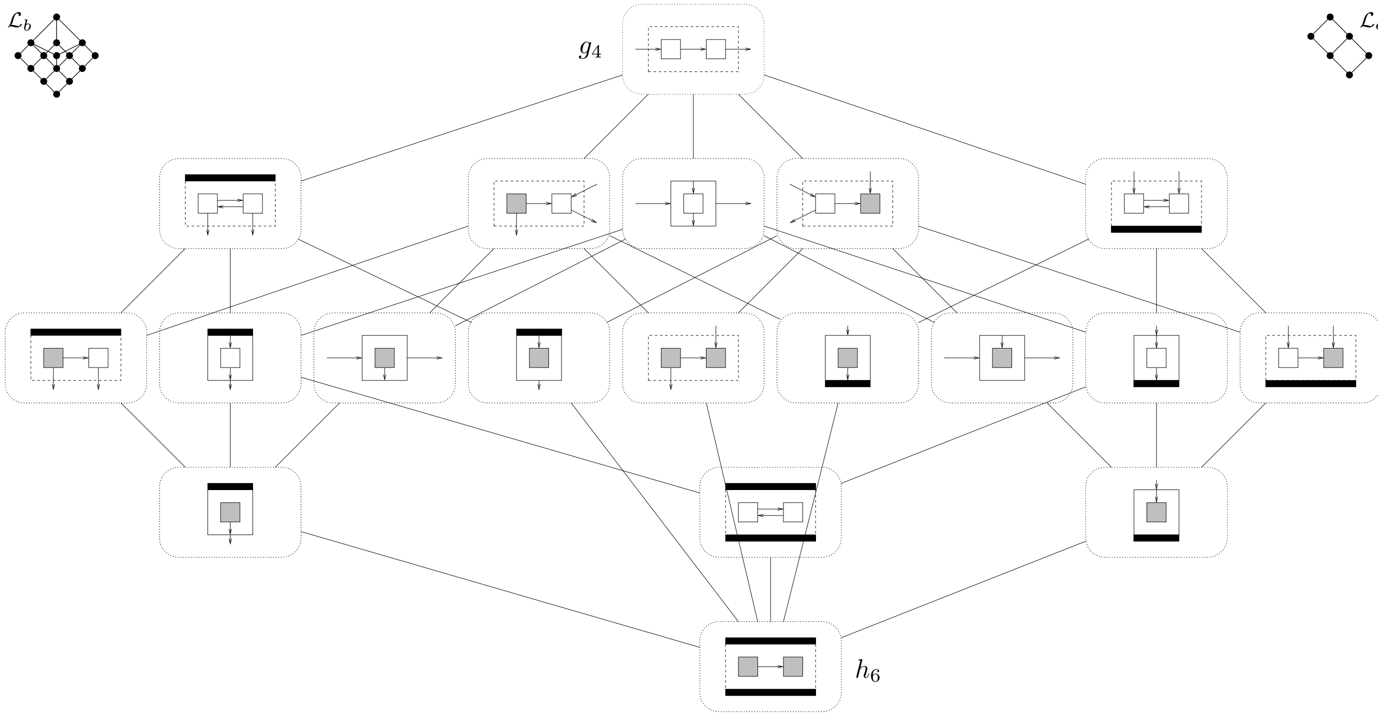


Figure 6. FCA concept lattice $\mathcal{L} = (\boldsymbol{S}_{F}, \leqslant)$ for context K of Table II (labels are collected in Table III; further comments are at the end of Section IV-A).

M is a *partial sink* system;

- if M has a component that is a source-only system, then M is a *partial source* system;
- if M—w.r.t. the environment—has only inputs, then M is a *total sink* system;
- if M—w.r.t. the environment—has only outputs, then M is a *total source* system;
- if a component of M is disconnected from the environment, then M is a *partially closed* system;
- if M—w.r.t. the environment—has neither inputs nor outputs, then M is a *totally closed* system.

TABLE II. AN FCA CONTEXT K.

	$n_\circ$	$n_\bullet$	$r_\circ$	$r_\bullet$	$c_\circ$
g_4					
b_3					×
a_3				×	
d_3		×			
a_4				×	×
b_4		×			×
a_5		×		×	×
e_3			×		
g_3	×				
c_4			×	×	
d_4	×		×		
e_4	×				×
f_4			×		×
h_4	×	×			
e_5	×			×	
f_5			×	×	×
g_5		×	×		
h_5	×	×			×
h_6	×	×	×	×	×

Using the symbols $n_\circ$, $n_\bullet$, $r_\circ$, $r_\bullet$, $c_\circ$, $c_\bullet$, for the predicates *partial sink*, *total sink*, *partial source*, *total source*, *partially closed*, *totally closed*, respectively, we have that $c_\bullet(S) \leftrightarrow n_\bullet(S) \wedge r_\bullet(S)$ (i.e., $c_\bullet$ can be expressed using $n_\bullet$ and $r_\bullet$). Hence, we use $n_\circ, n_\bullet, r_\circ, r_\bullet$ and $c_\circ$ as *properties* of systems and, therefore, employ the set $\{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$ as the set of *FCA properties*. The FCA context K of Table II associates the elements of $\boldsymbol{S}_{J}$ with properties in $\{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$.

Given the FCA context K of Table II, the corresponding *FCA concept lattice* $\mathcal{L}$ can be derived, and provides an ordering on *FCA concepts*—and, therefore, also an ordering on FCA objects, i.e., on systems of $\boldsymbol{S}_{J}$. The lattice $\mathcal{L}$ is shown in Figure 6 (given an FCA context, its FCA lattice can be constructed using the portal `latviz.loria.fr`). As mentioned before, we could use the labels $a_1, \ldots, h_6$ (corresponding to cells $(a, 1), \ldots, (h, 6)$ of Table I), to label the systems presented in Figure 6; these labels being collected in Table III. [Note that Table III provides labels for the nodes of the lattice of Figure 6 in a similar way as Table I provides labels for the nodes of the ordering of Figure 5.]

TABLE III. LABELS FOR THE FCA CONCEPT LATTICE OF FIGURE 6.

				g_4					
	a_3		e_3	b_3		g_3		d_3	
c_4	a_4	f_4	e_5	d_4		g_5	e_4	b_4	h_4
	f_5				a_5			h_5	
					h_6				

Notice (cf. Figure 5) that $g_4\uparrow = \{x \mid x \geq g_4\} = \{g_4, f_3, h_3, h_2, c_3, a_2, b_2, a_1\}$ and all the corresponding systems have none of the properties in $\{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$ (i.e., they are all *good* w.r.t. $\{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$) and, therefore, g_4

is included in Table II as the only representative of $g_4\!\uparrow$—this is why the set of systems reduced from $\mathrm{card}(\boldsymbol{S}_J) = 26$ to $19 = \mathrm{card}(\boldsymbol{S}_F)$, where $\boldsymbol{S}_F = \boldsymbol{S}_J \setminus \{S_j \mid j \in g_4\!\uparrow \setminus \{g_4\}\}$.

Figure 6 presents an FCA *concept lattice* $\mathcal{L}$ (for context K of Table II) with nodes being metabolic systems, and links indicating the health ordering, where the health of system ω is related to the set of its FCA properties $A_\omega \subseteq \{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$ (see Table II and Figure 6). Alternative sets of FCA properties and contexts could be used—consider the following contexts:
(a) $A_a = \{n_\circ, n_\bullet, r_\circ, r_\bullet, c_\circ\}$ and context $K_a = K$ of Table II;
(b) $A_b = A_a$ but K_b conforms to $n_\bullet \to n_\circ$ and $r_\bullet \to r_\circ$;
(c) $A_c = \{n_\circ \vee r_\circ, n_\bullet \vee r_\bullet, c_\circ\}$ with K_c a modification of K_b.
Note that Figure 6 shows the concept lattice $\mathcal{L}_a = \mathcal{L}$ for $K_a = K$, while the small inset lattices show FCA concept lattices $\mathcal{L}_b$ and $\mathcal{L}_c$ for the contexts K_b and K_c, respectively. The FCA objects $\boldsymbol{S}_F = \{S_{g_4}, \ldots, S_{h_6}\}$ form a lattice $\mathcal{L} = (\boldsymbol{S}_F, \leqslant)$, with "$\leqslant$" being the ordering relation. We expect the following: if $S_1 \leqslant S_2$ then $\mu(S_1) \leq \mu(S_2)$, where "μ" is a *numeric* (entropic, curvature based) measure on $\boldsymbol{S}_F$. [For additional information on FCA analysis of $\boldsymbol{S}_F$, cf. Appendix A.]

V. Supply Networks as Metabolic Networks

We consider a small supply network $\{b, c, d\}$ (which could be seen as a part of a larger network $\{a, b, c, d, f, g\}$); this supply network is shown in Figure 7, its self-adapting behaviour in Figure 8, and the list of nodes' *attitudes* in Table IV.

It will be demonstrated that the supply network of Figure 7 will, while adapting to the changing environment, exhibit a complex behaviour—note that it is the complex behaviour of the network that should be associated with the word "complex" in the term "complex network." For some research on using complex networks for supply chains, see [19][20].

We have specific values for the sets $\Theta, I, O, \Delta, \Phi, \Sigma, \Upsilon, \Gamma, \Psi, \Lambda, \Xi, \Pi$. We have $\Theta = \{\vartheta_b, \vartheta_c, \vartheta_d\}$ or simply $\Theta = \{b, c, d\}$. $I = \{i_{c.b}, i_{d.b}\}$. $O = \{o_{b.c}, o_{b.d}\}$. $\Delta = \varnothing_\Delta = \varnothing$ (because for node b we only consider its outputs (but not inputs), and for c and d we only consider their inputs (but not outputs)). $\Phi = \{\varphi_{b.c}, \varphi_{b.d}\}$. $\Lambda = \varnothing_\Lambda = \varnothing$ (it seems we will not have allostatic control mechanisms—but we will have some homeostatic ones). Σ and Υ would not be considered, given that all we have is flows from b to c and from b to d—however, if the system ω_2 of Figure 3 was considered, we could take $\Sigma = \{i_b.o_e\}$ and search for *pathways* Υ associated with the *streams* of Σ. Then, $\Xi = \varnothing_\Xi = \varnothing$ (because no interdependent networks are considered; but we would have interdependent networks, if we modeled *evacuation scenarios,* cf. Section VI). Π could be taken to be the flows $\Phi = \{\varphi_{b.c}, \varphi_{b.d}\}$—imposed by b on c and d (but earlier, possibly imposed on b by a).

Γ (catalysts) and Ψ (homeostatic control mechanisms)—omitted above—are the interesting ones, and we now analyse ω's *homeostatic control mechanisms* and *catalysts*. Figure 8 is almost self-explanatory: it demonstrates that both c and d will need to *split* (forming $\{c_1, c_2\}$ and $\{d_1, d_2\}$, respectively); then, c_2 and d_2 will need to *merge* in order to maximise efficiency while maintaining effectiveness.

Table IV shows (at lines 05–11) how nodes c and d calculate their *effectiveness* and *efficiency.* At line 13, c splits an ineffective node (namely, node c), and d splits an inefficient node (namely, node d). After the splits, we get (apart from

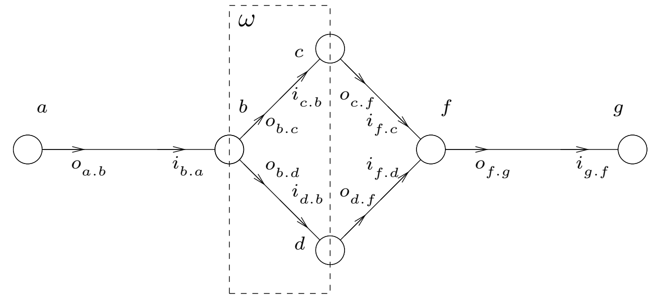


Figure 7. A metabolic supply network ω with nodes $\{b, c, d\}$.

effective and efficient nodes c_1 and d_1) an ineffective node c_2 (with 0 capability) and an inefficient node d_2 (with 0 flow). At line 25 (after a short negotiation) nodes c_2 and d_2 decide to *merge,* forming node e. It happens that e is both effective and efficient (if it was not, it would act in a way analogous to either c or d, and we would be left with an unmatched node with either 0 capability, or 0 flow, respectively); therefore, e accepts the flow of 2—the resulting system is, in this case, effective and efficient. The discussion demonstrates that the supply network can handle perturbations by using *homeostatic control mechanisms* of *split* and *merge.* Namely, we get the following. $\Psi = \{\psi_{\text{split}}, \psi_{\text{merge}}\}$ (more precisely, we get two types of *splits:* $\psi^{\text{ineffect}}_{\text{split}}$ (splitting an *ineffective* node), and $\psi^{\text{ineffic}}_{\text{split}}$ (splitting an *inefficient* node); regarding the merge operation $\psi_{\text{merge}} = \psi^{\text{c0f0}}_{\text{merge}}$ *ineffective* nodes with 0 capability are merged with *inefficient* nodes with 0 flow). Regarding the catalyst, we have $\Gamma = \{\gamma_{\text{ee}}\}$, and γ_{ee} should be understood as an *effectiveness and efficiency filter* that triggers the *split* and *merge* homeostatic control mechanisms. What the nodes *believe, intend,* and what (*split* and *merge*) operations they perform is shown in Table IV; however, some details—such as those related to *controlling, commanding, delegating, influencing* and *reporting*—are not included. Briefly: node c can *influence* node d by making d aware of c's beliefs or intentions; it can *command* node d by explicitly issuing a command (cf. line 23 in Table IV); or, finally, it can *control* other nodes (cf. lines 02–04 in Table IV). Work is underway on large scale supply networks self-adapting to perturbations, as described here.

Regarding *pathways* Υ, for ω of Figure 7, we have $\Upsilon = \varnothing$ (or rather $\Upsilon = \{\omega\}$). However, for large supply networks, it could easily be the case that pathways—corresponding to specific *functionalities*—could be identified (for instance, in food supply networks, we could identify pathways of *refrigerated* supplies). Regarding *interdependent networks* Ξ and *perturbations* Π we have $\Xi = \varnothing$ and $\Pi = \varnothing$. However, when evacuation scenarios are considered, they bring interdependent networks, cf. Section VI.

VI. Evacuation scenarios

A simple evacuation scenario involving *people*, *buses* and *fuel* is presented in Figure 9.

The scenario of Figure 9 involves three *interdependent networks:* ω_P is a network consisting of three nodes, $\Theta_P = \{\vartheta^b_P, \vartheta^c_P, \vartheta^d_P\}$ with *flows of people;* ω_B is a network consisting of three nodes, $\Theta_B = \{\vartheta^b_B, \vartheta^c_B, \vartheta^d_B\}$ with

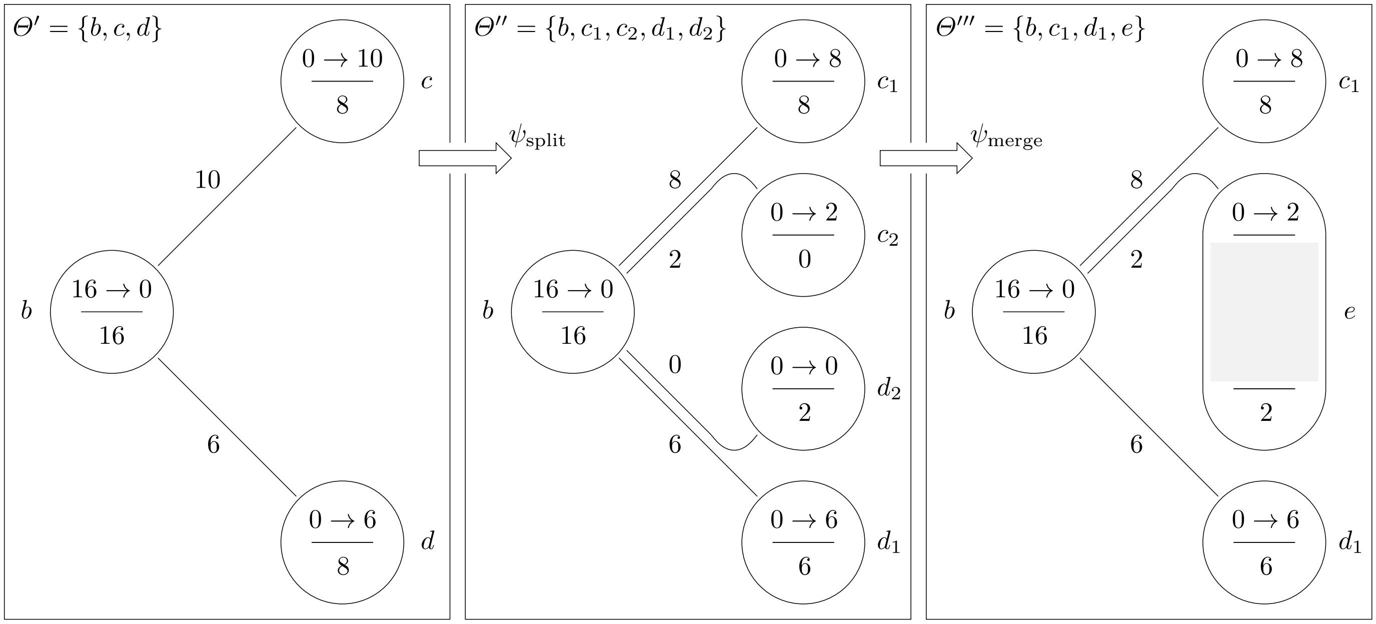


Figure 8. Self-adapting metabolic complex network $\omega(\Theta, I, O)$ with $\Theta' = \{b, c, d\}$, $\Theta'' = \{b, c_1, c_2, d_1, d_2\}$ and $\Theta''' = \{b, c_1, d_1, e\}$.

flows of buses; ω_F is a network consisting of three nodes, $\Theta_F = \{\vartheta_F^b, \vartheta_F^c, \vartheta_F^d\}$ with *flows of fuel.* The network ω_P can *request services* from ω_B—in order to evacute people, ω_P needs services that ω_B can provide; however, what is important is that the request for services obtained by ω_B can be seen as *perturbations* that ω_B has to handle. Similarly, the network ω_B can *request services* from ω_F—in order to run buses, ω_B needs services (namely: fuel) that ω_F can provide; again, it is important to note that the request for services obtained by ω_F can be seen as *perturbations* that ω_F has to handle.

This scenario will be further analysed in our future work—but the point we can make now is that we can see *perturbations* as interactions between *interdependent networks.*

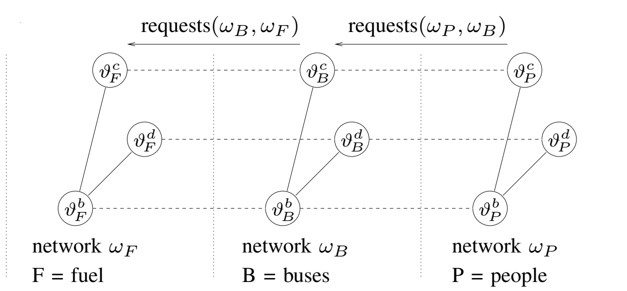


Figure 9. Evacuation scenario.

VII. Ontologies for robotics

There is a push for building autonomous systems, and autonomous robots. Given that autonomous robots' capabilities are becoming more complex, there is an urgent need to clarify such concepts as *capability, function, behaviour* and *structure*. Such analysis, however, could be performed at different levels, with capabilities (and the related functions and behaviours) being associated with some identified *conceptual levels.* Dennett [10] suggested the *intentional stance*, explaining that when conceptualising and analysing the world we might do so at different levels; he suggested a *physical level*, a *design (or functional) level*, and an *intentional level*. At the physical level, we use physics and physical properties (such as colour and size) to describe and understand the world. At the design level, we abstract away from physics and use such relations as *transform, move, attach, emit, sense, interpret* and *inform* (such relations have been employed in the Consensus system [21]). Using intentional level we ascribe *beliefs* (and other *attitudes*) to systems, for instance, we might say that a thermostat-controlled air-conditionning system has a belief "it's too hot in here" [22] (we should see such a system as an autonomous system capable of handling the temperature *perturbations*). Dennett has also mentioned a "person level"—but maybe "autonomy level" would be more appropriate, with *metabolic autonomy* understood as the capability to handle (various types of) perturbations. If Dennett's scheme was extended by adding the metabolic autonomy level, then the required concepts would include *pathways, homeostasis* and *perturbations*. Such an extension might be necessary if we want our robots to be robustly autonomous (i.e., resilient to perturbations).

The metabolic approach provides the highest level of the conceptualisation: we could consider a *metaphysical level* for *space, time, matter* (without physical boundaries), then Dennett's *physical, design* and *intentional level.* At the intentional level, we can have beliefs and intentions of single agents, but we can also have attitudes—and transfers of attitudes (including *influences* and *delegations*)—at the social level (of groups of agents). Finally, we have a *metabolic autonomy level* (with pathways, homeostatic control and perturbations). The following list provides some *conceptual relations* at those levels:

1) *metaphysical:* spatially / temp. connects, orients;
2) *physical:* temperature, size, rigidity, toxicity;
3) *design:* senses, moves, transforms, interprets;

 ISBN: 978-1-61208-787-0

TABLE IV. SELF-ADAPTING METABOLIC COMPLEX NETWORK $\omega(\Theta, I, O)$ MODIFYING ITS NODES AND FLOWS.

```
    intends(a,  intends(b,  flow([b,{c,d}], [t1,t2], 16)))
[t1,t2]
    intends(b,  flow([b,{c,d}], [t1,t2], 16))
    intends(b,  intends(c, flow([b,c], [t1,t2], 10))) % ................................. push([b,c], 10)
    intends(b,  intends(d, flow([b,d], [t1,t2],  6))) % ................................. push([b,d],  6)
        intends(c, flow([b,c], [t1,t2], 10))       intends(d, flow([b,d], [t1,t2], 6))
        believes(c, capability(c, 8))              believes(d, capability(d, 8))
        believes(c, state-curr(c, t1, 0))          believes(d, state-curr(d, t1, 0))
        believes(c, state-next-flw(c, t2, 10))     believes(d, state-next-flw(d, t2, 6))
        believes(c, state-next-cap(c, t2,  8))     believes(d, state-next-cap(d, t2, 6))
        believes(c, state-next-max(c, t2,  8))     believes(d, state-next-max(d, t2, 8))
        believes(c, effective(c, [t1,t2], 8/10))   believes(d, effective(d, [t1,t2], 6/6))
        believes(c, efficient(c, [t1,t2], 8/8))    believes(d, efficient(d, [t1,t2], 6/8))
        believes(c, ineffective(c, [t1,t2]))       believes(d, inefficient(d, [t1,t2]))
        splits-ineffective-node(c, [c, [c1, c2]])  splits-inefficient-node(d, [d, [d1, d2]])
        intends(c,  flow([b,c1], [t1,t2], 8))      intends(d, flow([b,d1], [t1,t2], 6))
        intends(c,  flow([b,c2], [t1,t2], 2))      intends(d, flow([b,d2], [t1,t2], 0))
        believes(c1, effective-efficient(c1)*)     believes(d1, effective-efficient(d1)*) % pull([c1,b], 8)
        believes(c2, ineffective(c2, [t1,t2]))     believes(d2, inefficient(d2, [t1,t2])) % pull([d1,b], 6)
        believes(c2, capability(c2, 0))            believes(d2, capability(d2, 2)
        intends(c2,  flow([b,c2], [t1,t2], 2))     intends(d2, flow([b,d2], [t1,t2], 0))
        believes(c2,t1,effective(c2,[t1,t2],0))    believes(d2, t1, effective(d2, [t1,t2], 1))
        believes(c2,t1,efficient(c2,[t1,t2],1))    believes(d2, t1, efficient(d2, [t1,t2], 0))
        believes(c2,t1,ineffective-w0c(c2,[t1,t2])) believes(d2, t1, inefficient-w0f(d2, [t1,t2]))
        commands(c2,d2,offer(flow(c2,d2,t*,2)))
                                                   accepts (d2, c2, offer(flow(c2,d2,t*,2)))
        agrees(c2, t1, merge([c2,d2],e,[t1,t2]))   agrees(d2, t1, merge([c2,d2],e[t1,t2]))
            merges-0-cap-nodes-with-0-flw-nodes([c2,d2], [[c2, d2], e])
            believes(e, capability(e,      2))
            intends(e,  flow([b,e, [t1,t2], 2)) ........................................% pull([e,b],  2)
```

4) *intentional:* believes, intends, obligates, influences;
5) *metabolic:* perturbs, homeo-allostatically controls.

[For information on the conceptualisation cf. Appendix B.]

We could consider the following examples of metabolic streams at the above five levels: (1) streams of gravitational forces; (2) flows of water through a city; (3) streams of aircraft's radar readings transformed into movement maneuvers; (4) streams of beliefs, intentions, influences and obligations during a country's elections; (5) streams of modifications to city's infrastructure systems' homeostatic control mechanisms. It should be noted that many conceptual relations can be found at multiple levels in the above conceptualisation—consider e.g., *connects, senses* or *strikes* (with a *weapon*). It should also be noted that a metabolic system, in order to keep functioning (or keep living) should be continuously monitoring the environment in order to detect and handle all perturbations that have the potential to push the system out of the equilibrium state—it seems that this process of monitoring and handling perturbations constitutes the system's top-level goal. To test, verify and progress with such a framework, multiple case studies should be performed (such case studies could vary widely w.r.t. both domains and scale).

We are also interested in *non-monotonic ontology evolution,* cf. the last paragraph of Section VIII.

VIII. FUTURE WORK

There is a growing need to design and build robust autonomous systems; some domains, research areas and mathematical tools related to autonomy are listed in Table V.

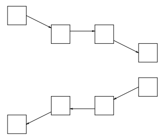
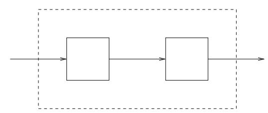
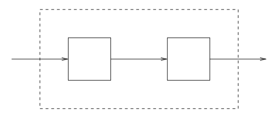
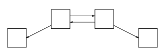
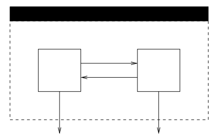
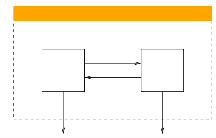
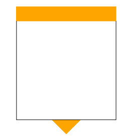
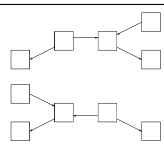
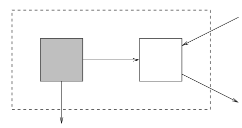
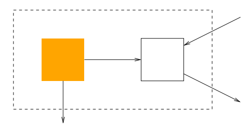
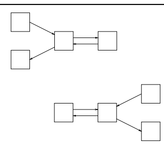
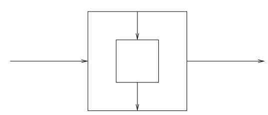
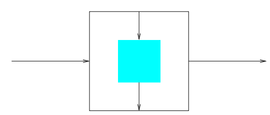
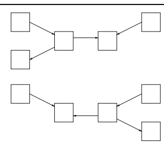
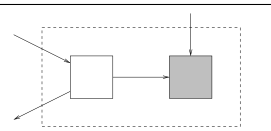
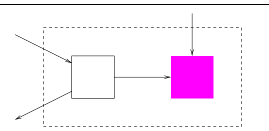
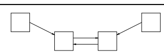
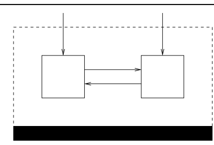
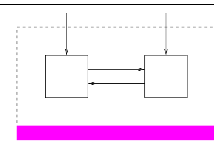
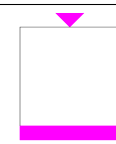
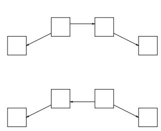
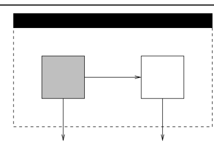
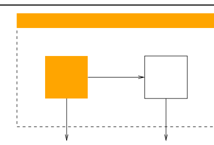
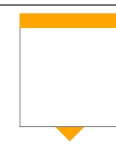
TABLE V. DOMAINS AND TOOLS FOR AUTONOMY.

curvature & entropy	Forman-Ricci Ollivier-Ricci	$\overset{d \leftarrow c}{\longleftarrow}$	Ricci curvature (on manifolds)
maths	geometry		geometry
measures	■ generic entropy/curvature-based measures ↓ ■ specific performance measures (cf. generic meas.) ↑		
concepts	robustness resil. to pert. ↕ perturbations self-adaptation autonomy	$\overset{?}{\longleftarrow}$ ≡ ≡ ≡	robustness ↕ perturbations self-adaptation autonomy
maths sub-area area	graph th. & topology complex networks networks	$\overset{appr}{\longrightarrow}$	diff. equations complex dyn. sys. dynamical systems
domain	discrete		continuous

However, there is no consensus on the concepts of autonomy or robustness; sometimes robustness is considered to be a property that a system is said to possess if it does not fail some *proposed performance tests.* Autonomy is usually understood as ability to function without external control—but it has to be added that this ability should survive certain *changes (perturbations)* occurring in the complex (dynamic, uncertain) environment. Robustness, understood as *resilience to perturbations,* is closely related to self-adaptation; and it is the notion of perturbations that links robustness with adaptation—as indicated in the *concept rows* of Table V.

 ISBN: 978-1-61208-787-0

TABLE VI. TWO COMPONENT SYSTEMS: LABELS, CODES, GRAPHS AND TOPOLOGIES (PART 1 OF 3).

(1) label l	(2) code c	(3) graph J_c ($\subseteq J$)	(4) topology T_l	(5) topology T_l^c	(6) topology T_l^m
g_4	10 10 01 01 01 10				
a_3	11 01 01				
e_3	10 01 11 01 11 01				
b_3	11 11 00 11 00 11				
g_3	10 11 10 01 10 11				
d_3	11 10 10				
c_4	10 01 01 01 01 01				

Given a functioning system ω, it seems appropriate to say that: *autonomous*$(\omega) \leftrightarrow$ *self-adaptable*$(\omega) \leftrightarrow$ *robust*(ω). Furthermore, *robustness* has been linked to *curvature* and *entropy* [4]; therefore, entropic curvature-based measures could be applied to assess systems' *robustness*. It should be noted that curvatures are geometric notions; they have been recently considered appropriate for quantifying functionality and robustness of networks [4]. The top part of Table V shows that *discretisations* (changing from *continuous* to *discrete*) have been applied to Ricci curvature to produce (discrete) Forman-Ricci and Ollivier-Ricci curvatures. The second line of Table V, labelled *measures,* indicates that a promising line of research should investigate: (a) entropy/curvature based measures that should apply to systems in general; (b) performance/functionality related measures for specific systems (such as *supply networks*); (c) relating general entropic measures (of (a)) to specific performance measures (of (b)) for selected classes of systems.

We also have work underway looking at *non-monotonic ontology evolution.* Ontologies have traditionally been developed monotonically in the sense of them being built by additively including additional concepts. This assumes that the problem is fixed and we can build, eventually, a full and correct account of the universe of discourse. This does not suffice for a changing problem environment where we have to adapt. The insight is that what is required is a set of bounding conditions within which we allow non-monotonic ontology change; these bounding conditions would be related to regimes of behaviour picked out by order parameters as mechanisms of abstraction.

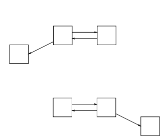
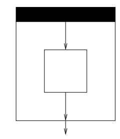
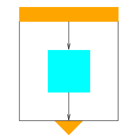
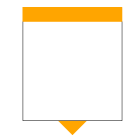
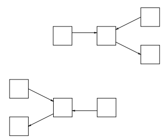
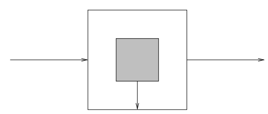
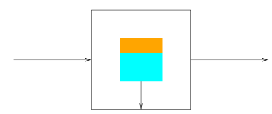
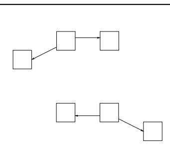
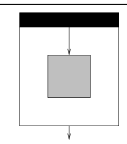
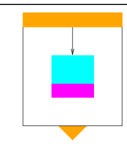
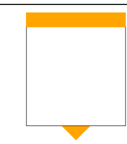
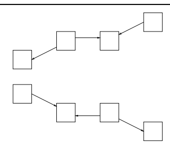
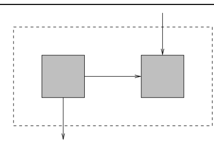
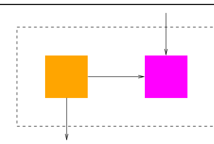
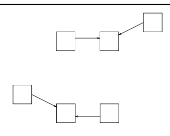
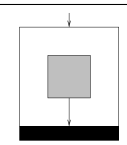
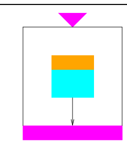
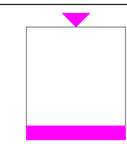
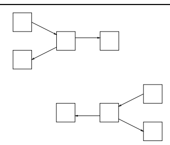
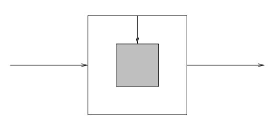
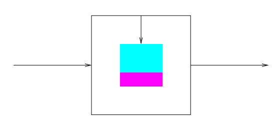
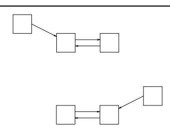
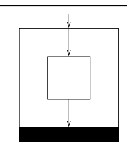
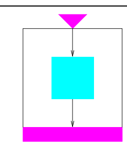
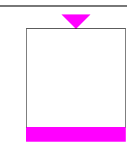
IX. CONCLUSION

This paper provides a description of the *Metabolic Complex Networks (MCN)* framework and an analysis of *supply networks* from the perspective of the MCN framework; the obtained set of *homeostatic control mechanisms* $\Psi = \{\psi_{\text{split}}, \psi_{\text{merge}}\}$ is appropriate. Research on adaptive supply networks [7][19][20][23]–[24] and large scale implementation (using Python and NetworkX) are underway.

The essential components of the framework are:

 ISBN: 978-1-61208-787-0

TABLE VII. TWO COMPONENT SYSTEMS: LABELS, CODES, GRAPHS AND TOPOLOGIES (PART 2 OF 3).

(1) label l	(2) code c	(3) graph J_c ($\subseteq J$)	(4) topology T_l	(5) topology T_l^c	(6) topology T_l^m
a_4	11 01 00 11 00 01				
f_4	10 00 11 01 11 00				
e_5	10 01 00 01 00 01				
d_4	10 01 10 01 10 01				
g_5	10 00 10 01 10 00				
e_4	10 11 00 01 00 11				
b_4	11 10 00 11 00 10				

I. ONTOLOGY: conceptual understanding of the domain;
II. METABOLISM: homeostatic handling of perturbations;
III. CURVATURE: entropic quantifying of robustness.

Future work—sketched in Section VIII—will focus on a metabolic analysis of evacuation operations expanded by an entropic analysis of robustness of the selected systems.

ACKNOWLEDGMENT

La démarche de chercher à comprendre les réseaux complexes comme des systèmes métaboliques est née lors du séjour du premier auteur à Rouen, en France, en décembre 2018 et janvier 2019. Merci à Thérèse et Daniel Caillemet pour la création d'un environnement métabolique excellent, impossible à trouver ailleurs.

APPENDIX

A. *FCA analysis of two component systems*

This appendix provides details on the FCA analysis of Section IV-A. Tables VI–VIII show labels, codes and ways to visualise the systems and facilitate the construction of FCA contexts (colours *magenta/orange/cyan* indicate *sink/source/closed* ($n_\circ$/$r_\circ$/$c_\circ$), respectively). In Tables VI, VII and VIII, the following columns are used: (1) label l, with labels $a_1, \ldots, h_6$ corresponding to the cells $(a, 1), \ldots, (h, 6)$ of Table I; (2) code c, identifying the subset J_c of J—these codes have been collected in Table I; (3) graph J_c ($\subseteq J$), showing input–output connections between the two components of the system and the environment—note that Figure 4 shows the graph J_{111111} (of system S_{111111}) for code $c = 111111$; (4) topology T_l, representing the connection topology of graph J_c, with the association between the label l and the code c provided in Table I (note that multiple codes can be associated with a single label, as multiple graphs can have the same connection topology); these topologies have been shown in Figures 5 and 6; (5) topology T_l^c, visualising topology T_l differently, using colour; (6) topology T_l^m, a modification of T_l^c.

The contexts K_b and K_c mentioned in Section IV-A are provided in Tables IX and X; their concept lattices $\mathcal{L}_b$ and $\mathcal{L}_c$ were provided in Figure 6.

TABLE VIII. TWO COMPONENT SYSTEMS: LABELS, CODES, GRAPHS AND TOPOLOGIES (PART 3 OF 3).

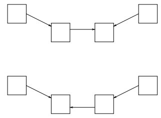
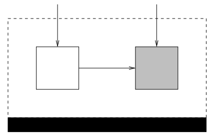
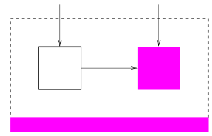
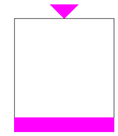
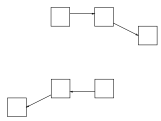
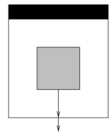
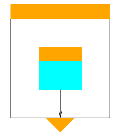
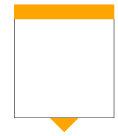
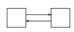
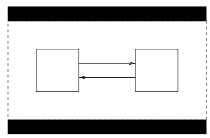
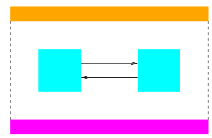
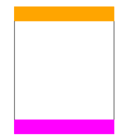
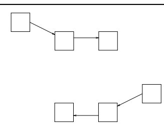
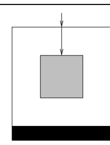
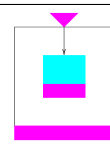
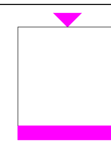
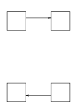
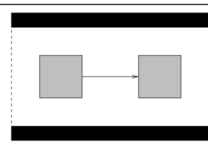
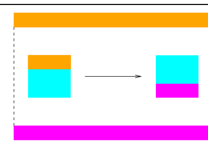
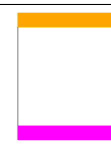
(1) label l	(2) code c	(3) graph J_c ($\subseteq J$)	(4) topology T_l	(5) topology T_l^c	(6) topology T_l^m
h_4	10 10 10 01 10 10				
f_5	10 00 01 01 01 00				
a_5	11 00 00				
h_5	10 10 00 01 00 10				
h_6	10 00 00 01 00 00				

TABLE IX. AN FCA CONTEXT K_b.

	$n_\circ$	$n_\bullet$	$r_\circ$	$r_\bullet$	$c_\circ$
g_4					
b_3					×
a_3			×	×	
d_3	×	×			
a_4			×	×	×
b_4	×	×			×
a_5	×	×	×	×	×
e_3			×		
g_3	×				
c_4			×	×	
d_4	×		×		
e_4	×				×
f_4			×		×
h_4	×	×			
e_5	×		×	×	×
f_5			×	×	×
g_5	×	×	×		×
h_5	×	×			×
h_6	×	×	×	×	×

TABLE X. AN FCA CONTEXT K_c.

	$n_\circ \vee r_\circ$	$n_\bullet \vee r_\bullet$	$c_\circ$
g_4			
b_3			×
a_3	×	×	
d_3	×	×	
a_4	×	×	×
b_4	×	×	×
a_5	×	×	×
e_3	×		
g_3	×		
c_4	×	×	
d_4	×		
e_4	×		×
f_4	×		×
h_4	×	×	
e_5	×	×	×
f_5	×	×	×
g_5	×	×	×
h_5	×	×	×
h_6	×	×	×

B. Conceptualising metabolic systems

Regarding the conceptualisation, the following list of *conceptual relations* was presented in Section VII.

1. metaphysical: spatially / temporally connects, orients;
2. physical: temperature, colour, weight, rigidity, toxicity;
3. design: senses, moves, transforms, interprets, informs;
4. intentional: believes, intends, obligates, influences;
5. metabolic: perturbs, homeo-allostatically controls.

We list selected *concepts* and *relations* for the above five levels.

1) metaphysical level
   a) `process, spatial, temporal`
   b) `connects, meets`
   c) `distance, between`
2) physical level
   a) `temperature` (+ physical props/rels)
   b) `toxicity` (+ chemical props/rels)

3) design level
    a) `transforms`
    b) `damages, repairs, enables`
    c) `attaches, moves`
    d) `emits, senses`
    e) `interprets, informs`
4) intentional level
    a) `believes, desires, intends`
    b) `permits, obligates`
    c) `influences, controls`
    d) `delegates`
5) metabolic level
    a) `perturbs`
    b) `detects-perturbation`
    c) `handles-perturbation`
    d) `homeostatically-controls`
    e) `allostatically-controls`

In the remainder of this appendix, we provide further comments on the above ontological concepts and relations by listing some frameworks and examples. For instance, (1b) clarifies that *connects* and *meets* are the primitive relations of the *Region Connection Calculus (RCC)* framework [25] and of Allen's *interval algebra* [26], respectively, while (3a) lists three examples of *transformation:* biological metamorphosis, cellular respiration ($\rightarrow$) & photosynthesis ($\leftarrow$), and transformation of mercury (Hg) into gold (Au).

Metaphysical level (1)—existence, space, time
(1b) `connects` [25] and `meets` [26]
(1c) *qualitative distance* (e.g., `near` and `far`) [27]
(1c) `between`: cf. *[Paris/1915, Berlin/1910, Moscow/1905]*
Physical level (2)—the domain of *physics* and *chemistry*
(2a) physical properties: temperat., colour, mass, size, texture
(2b) chemical properties: toxicity, flammability, chem. stabil.
Design level (3)—above the level of physics and chemistry
(3a) `transforms`: *metamorph.*, $C_6H_{12}O_6 \leftrightarrow CO_2$, $Hg \rightarrow Au$
Intentional level (4)—attitudes (information, pro, normative)
(4a) `believes`, `intends`: cf. $B_i \neg\alpha \wedge B_i I_j B_i \alpha$ [28]
(4b) `permits`: cf. $P_{k,i}\,[\text{protects}(i,i)] \leftarrow I_j\,[\text{kills}(j,i)]$
(4c) `influences`: *"Ann influences Ben to control Craig"*
Metabolic level (5)—functioning despite *perturbations*
(5a) `perturbs`: cf. earthquake, flood, socio-techn. systems
(5b) `detects perturb.`: cf. earthquake warning system
(5c) `handles perturb.`: cf. flood mitigation dams
(5d) `homeostatically controls`: cf. air-cond. system
(5e) `allostatically controls`: social adaptation [29].

REFERENCES

[1] S. H. Strogatz, "Exploring complex networks," *Nature*, vol. 410, pp. 268–276, 2001.
[2] X. F. Wang and G. Chen, "Complex Networks: Small-World, Scale-Free and Beyond," *IEEE Circuits and Systems Magazine*, pp. 6–20, 2003.
[3] A.-L. Barabási and E. Bonabeau, "Scale-free networks," *Scientific American*, vol. 288, pp. 50–59, 2003.
[4] A. R. Tannenbaum et al., "Graph curvature and the robustness of cancer networks," 2015, arXiv 1502.04512.
[5] J. Sia, E. Jonckheere, and P. Bogdan, "Ollivier-Ricci curvature-based method to community detection in complex networks," *Scientific Reports*, vol. 9, pp. 1–12, 2019.
[6] M. Magoni, "Resilience thinking and urban metabolism in spatial planning: which possible integrations," *City, Territory and Architecture*, vol. 4, pp. 1–8, 2017.
[7] P. G. Flikkema and M. Vigil-Hayes, "Self-adaptive and resilient urban networking infrastructure for disasters and smart city services," in *IEEE International Conference on Big Data*, pp. 4074–4079, 2017.
[8] C.-C. Ni, Y.-Y. Lin, F. Luo, and J. Gao, "Community Detection on Networks with Ricci Flow," *Scientific Reports*, vol. 9, pp. 1–12, 2019.
[9] B. Ganter and R. Wille, *Formal Concept Analysis. Mathematical Foundations*. Springer, 1996.
[10] D. C. Dennett, *The Intentional Stance*. The MIT Press, 1987.
[11] A. Samal et al., "Comparative analysis of two discretizations of Ricci curvature for complex networks," *Scientific Reports*, vol. 8, pp. 1–16, 2018.
[12] D. J. Reid, "An Autonomy Interrogative," in *Foundations of Trusted Autonomy* (H. A. Abbass, J. Scholz, and D. J. Reid, eds.), pp. 365–391, Springer, 2018.
[13] D. L. Nelson and M. M. Cox, *Lehninger Principles of Biochemistry*. W. H. Freeman and Company, 2008.
[14] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, 1999.
[15] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks," *Nature*, vol. 393, pp. 440–442, 1998.
[16] S. Okuda et al., "KEGG Atlas mapping for global analysis of metabolic pathways," *Nucleic Acids Research*, vol. 36, no. 1, pp. W423–W426, 2008.
[17] P. V. Sengbusch, *Metabolic Pathways*, 1998 (accessed January 26, 2020). https://s3.lite.msu.edu/res/msu/botonl/b_online/ e19_1/metabol.htm.
[18] *Biochemical Pathways, Part 1: Metabolic Pathways*. F. Hoffmann-La Roche AG, 2020 (accessed January 26, 2020). http://biochemical-pathways.com/#/map/1.
[19] E. J. S. Hearnshaw and M. M. J. Wilson, "A complex network approach to supply chain network theory," *International Journal of Operations & Production Management*, vol. 33, pp. 442–469, 2013.
[20] S. I. Mari, Y. H. Lee, and M. S. Memon, "Complex network theory-based approach for designing resilient supply chain networks," *International Journal of Logistics Systems and Management*, vol. 21, pp. 365–384, 2015.
[21] D. A. Lambert, A. Saulwick, and K. Trentelman, "Consensus: A comprehensive solution to the Grand Challenges of Information Fusion," in *Proceedings of the 18th International Conference on Information Fusion*, pp. 908–915, 2015.
[22] J. Searle, *Minds, brains and science*. Harvard University Press, 1985.
[23] S. S. N. Perera, M. G. H. Bell, and M. C. J. Bliemer, "Modelling supply chains as complex networks for investigating resilience: an improved methodological framework," in *37th Australasian Transport Research Forum*, pp. 1–16, 2015.
[24] A. Surana, S. Kumara, M. Greaves, and U. N. Raghavan, "Supply-chain networks: a complex adaptive systems perspective," *International Journal of Production Research*, vol. 43, pp. 4235–4265, 2005.
[25] D. A. Randell, Z. Cui, and A. G. Cohn, "A spatial logic based on regions and connection," in *3rd Int. Conf. on Knowledge Representation and Reasoning* (B. Nebel, C. Rich, and W. Swartout, eds.), pp. 165–176, Morgan Kaufmann, 1992.
[26] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983.
[27] D. Hernández, E. Clementini, and P. D. Felice, "Qualitative distances," in *Spatial Information Theory A Theoretical Basis for GIS. COSIT 1995* (A. U. Frank and W. Kuhn, eds.), pp. 45–57, Springer, 1995.
[28] W. van der Hoek and M. Wooldridge, "Towards a logic of rational agency," *L. J. of the IGPL*, vol. 11, pp. 133–157, 2003.
[29] J. Schulkin, "Social allostasis: Anticipatory regulation of the internal milieu," *Frontiers in Evolutionary Neuroscience*, vol. 2, pp. 1–15, 2011.

 ISBN: 978-1-61208-787-0