An Information-oriented View of Multi-Scale Systems

Ada Diaconescu
Computer Science & Networks
Telecom Paris, LTCI, IPP
diacones@telecom-paris.fr

Louisa Jane Di Felice
Institute of Environmental Science & Technology
Autonomous University of Barcelona
louisajane.difelice@uab.cat

Patricia Mellodge
Electrical & Computer Engineering
University of Hartford
mellodge@hartford.edu

***Abstract*—Multi-scale configurations are essential for dealing with system complexity. Yet, a theory for facilitating their cross-domain implementation is missing. We set the basis for such theory, by demystifying the core concepts of 'scale' and 'multi-scale' from an information perspective. In this view, we define and apply *information abstraction entropy* as a measure to assess and compare system scales.**

***Index Terms*—multi-scale, information, abstraction, entropy**

I. Motivation & Objective

Multi-scale structures are prevalent in natural and artificial systems, as they can handle increasing complexity [23], [16], [17]. Engineered systems, including autonomic, self-integrating and intelligent systems, can draw key benefits from such designs. Yet, transferring multi-scale concepts to these fields requires a multi-scale system theory, focusing on feedback systems. Our long-term goal is to develop such theory. This position paper aims to set the required conceptual basis, by demystifying the key concepts of 'scale' and 'multi-scale' via their various usages in relevant contexts.

Several terms are employed almost interchangeably across application domains relative to the multi-scale notion – e.g., hierarchy, holarchy, multi-level/-layer, nested, embedded, micro-macro, blurring, or coarse graining [8] [2] [19] [14]. While the concrete meanings behind these terms may differ slightly, several core commonalities persist across all cases. We highlight these common features here, focusing on the special role of *information* as a unifying concept across both physical and computational systems. To render these generic concepts more concrete, we suggest quantifying them in terms of their information content, impact, and cross-scale abstraction.

II. Previous Work

Many fields have studied multi-scale systems from different perspectives. Hierarchy theory is concerned with the epistemology of the scale concept, viewed as an observer-dependent construct [1]. It has been mostly applied to the field of ecology, where scales tend to map onto nested spatial features (e.g. tree patches nested into forests) [25]. In control engineering, hierarchical control theory characterises interactions between multiple control layers [20]. The scale question is central to many social science theories, which focus on *micro-macro linkages*, i.e., how the micro-scale affects the macro-scale, and vice-versa [6]. Here, as in hierarchy theory, the scale notion is not fixed [13], although generally micro-scales refer to individual behaviours, while macro-scales describe social patterns. The question of scales is also prevalent in neuroscience, where the brain is often described as a multi-scale system (e.g., the visual cortex architecture [24] [11]). In philosophy of information, Floridi's *method of levels of abstraction* argues that abstraction levels can be identified with respect to levels of explanation, organisation, or conceptual schemes [10]. We aim to identify the commonalities of these specific studies as a basis for a generic theory of multi-scale (feedback) systems.

In previous work, we highlighted the shared features of multi-scale structures in natural and artificial domains [5], and started formalising these via a generic design pattern – multi-scale abstraction feedbacks (MSAF) [4]. In brief, we model multi-scale systems as ensembles of information flows, which are merging and splitting to generate various information abstractions (micro-to-macro) and information reifications (macro-to-micro). In [15], we study the impact of inter-scale timing on the macro-properties of multi-scale systems.

III. What is Scale?

From across dictionary entries[1,2,3], the term "scale" has two main (interrelated) meanings that are relevant to our study:

1) Related to measurement – either the units of measurement (e.g., metric scale) or the instruments of measurement (e.g., weighting scale).
2) Related to a ratio between a real object and a model of the object (e.g., a map scale). Here, scale can also be used as a verb – i.e., to scale-up or -down – meaning that something pertaining to the scaled object increases or decreases proportionally with that with respect to which it is scaled. E.g., scaling-up a business, or a computing process, means that it can deal with more incoming requests while using a proportional amount of resources.

We associate scale to the measurement of a system of interest, where such measures can then be used to create system models, at a certain ratio. Hence, we define scale as *the granularity of observation of a targeted object* (Figure 1). Granularity can represent, e.g., an interval, range, abstraction

[1]Merriam-Webster online: https://www.m-w.com/dictionary/scale
[2]Math: https://everydaymath.uchicago.edu/teachers/6th-grade/glossary/
[3]Collins Dictionary: https://www.collinsdictionary.com/dictionary/english/scale

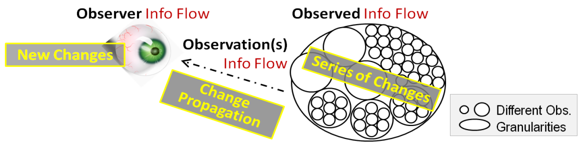

Fig. 1. Scale: granularity of observation of a targeted object by an observer – a system view focused on information flows

level, or frequency. Importantly, scale is thus a property of the observation (which includes representation) and not of the observed object. Moreover, scale necessarily implies the existence of an observer, performing the observation. The observer can be any entity that can acquire information from the system and change based on that information (Cf. Hierarchy Theory [1]) – e.g., a human observer, and external system or an internal system entity.

In this view, an information scale is related to the kind of information acquired by an observer about an observed object. A "higher" scale refers to a larger, or coarser, granularity of observation of a targeted object (more abstraction); a "lower" scale implies a smaller, or finer, granularity of observation of the same object (more detail). In multi-scale systems, it is often the case that higher scales provide more abstract information about larger scopes of the observed object; while lower scales provide more detailed information about narrower scopes of the observed object. This limits the amount of information observed, stored and processed at each scale – thus distributing the information load across multiple scales.

Importantly, this does not mean that higher scales contain less information or are less "complex" than lower scales. Nor does it mean that they contain more information or complexity. They merely contain less detailed information about each micro-entity; and often compensate by observing multiple such micro-entities and their interrelations. Hence, the amount of information held by a macro-entity may be larger, smaller, or equal to that held by each of its observed micro-entities.

In the following, we highlight the key role of information as a unifying factor across scales and application domains.

IV. Information Systems

Observation is the acquisition of information about an object of interest. Generally, such observation may or may not lead to the analysis of the acquired information, and to some sort of action or change in the observer (the receiver of that information). As we focus on a phenomenological system view, we are mostly interested in cases where observation leads, eventually, to an observable change (even if after a delay, e.g., via storage and later processing).

Hence, for the purpose of our study, we consider the definition of information as *an observable difference or change in an object that propagates and triggers change in an observer*. This view is consistent with the semantic definitions of information [12] [9], viewing information not only as a material flow (*data*), but as something to be received and interpreted (*data + meaning*). An information flow, thus, is a series of events, or changes, that lead to changes in an observer (the recipient of that information flow).

Thus, we can model our generic system of observed objects, observations, and observers (Figure 1) as an ensemble of information flows influencing each other: an observed information flow propagates changes via an observation information flow leading to changes in an observer information flow. We link this informational system model to physical systems as follows (extension of considerations in [5]).

Fundamentally, a physical object is some sort of *process* – i.e., a series of changes in observable variables happening in some order [18]. When a process keeps its identity, tied to a set of variables and their values, unchanged for a relatively long period (from the observer's perspective) we consider the process to be an *entity*; and consider the few variables that do change to define the entity's state (e.g. a stone, keeping its structure static for a long period, while changing its position and speed variables, in the eye of its observers).

Information flows are linked to physical processes, or entities, by the fact that any piece of information must rely upon, or be encoded onto, a physical substrate (even if the same information may be encoded onto various physical substrates). This is consistent with the findings in [3], where a piece of information about a physical object requires a minimum amount of energy (or negative entropy – negentropy) to be extracted. This information can then be employed to create some energy in return (negentropy), while of course abiding to the second law of thermodynamics.

Based on the above considerations, we model a system as a network of information flows, which change, observe, and adapt to each other repeatedly. Each information flow may be observed by other information flows, via yet other information flows. The system is populated by information flow triplets: i) an observed information flow, ii) an observation information flow, and iii) an observer information flow (Figure 1). Information scales may vary across such information flow triplets within a system. To simplify, we only refer in the following to observer-observed pairs, implicitly including the observation flow connecting them. While we keep the three kinds of information flows separate, we can consider the system as a network of information flows that run in parallel, merge into each other, and split at various points (e.g., just like we distinguish between rivers with different names, whereas they merely represent a large network of water flows). As information flows can represent changes in processes, entities, or systems, we use these terms interchangeably next.

V. Kinds of Scales

What kinds of scales are we talking about? The kinds of scales of interest are related to the kinds of observed variables (of interest to the observer). This, in turn, depends on the targeted domain of application or study, and on the appropriate scale for that application or study. E.g., the observed variables of interest for most physical objects are related to space and time. The variables of interest for a computing system may be related to the usage of processing, storage, and communication resources; to the number of component instances and their interrelations (i.e., structural models); or to the client-triggered call-paths through the system (i.e., behavioural models). These computing variables are all information-related, even if ultimately traceable to physical processes.

In physical studies, considering space as an observable dimension already implies a minimum abstraction, or granularity of observation, as this dimension won't be found at the quantum scale [18]. Thus, space already represents an abstract variable only perceivable by observers at "larger" scales. At smaller scales, we may consider instead the frequencies or energy needed for observation (out of scope). We can still consider space as an observable variable for most physical systems, unless we analyse quantum (computing) systems. Hence, it is useful to consider spatial scales. Similarly, temporal scales are relevant for all processes, physical or computational, as a local means of ordering observations.

Spatial scale typically represents a unit interval, area, or volume over which observations are made. Observations here are about spatial properties, structures, or shapes. Thus, spatial scale refers to *what* is observed. Temporal scale refers to the frequency of observation. It refers to *when* observations are made, yet not to what is observed (e.g., spatial properties). At "higher" scales, information acquired via an observation can, in itself, represent variables of further observation, at even higher scales. This is the case when monitoring information variables of computing processes (e.g., the state of a Java Object or of an e-shopping basket). This means that we can talk about information scales, or abstraction levels.

What does it mean for a system to 'operate' at a certain scale? We understand by this that the system's variables are observed at a certain granularity. The observers, which may or may not be part of the same system, may operate at the same scale, or at different scales. In a single-scale system, all processes observe each other at similar scales, and change accordingly. A system (or system-of-systems) where different parts operate at different scales is referred to as a *multi-scale system*, i.e., a system observed at multiple granularities. It implies the existence of observers that perform some sort of mapping, or translation, between scales: observing variables of entities that operate at a lower scale and abstracting these for entities that operate at a higher scale; or the opposite, from higher to lower scales. We refer to an entity operating a higher scale as a higher-scale entity (macro) and to an entity operating at a lower scale as a lower-scale entity (micro). Higher/lower and macro/micro are relative to each pair of scales.

Generalising across physical and informational objects, and in line with the above definition of information, we reduce our focus to information scales only (and associated time scales of observation). Hence, multi-scale systems are ensembles of information flows that observe each other and that may be observed externally at different scales (micro and macro). The scale of observation represents the interval, range, granularity, or abstraction level at which information is acquired.

VI. Multi-Scale Feedback Systems

When information cycles between scales of observers, it forms multi-scale feedback loops [5]. Information about a lower scale (micro) is abstracted onto entities at a higher scale (macro), and this abstraction, in turn, is observed at the lower scale (micro), leading to adaptation. Hence, the higher scale observes the lower scale, and the lower scale observes the resulting abstraction from the higher scale (Figure 2). The observers of such multi-scale feedback systems are internal to the system (i.e., entities or processes that belong to the system).

The advantage of such multi-scale feedback loops lies in allowing individual micro-entities to access abstract information about the entire state of a set of micro-entities, possibly including themselves. Using abstract information instead of detailed information lowers the amount of resources needed to communicate, store, and process it [4]. Hence, it allows for a large set of micro-entities to coordinate based on abstracted information about their collective state, thus using a limited amount of resources (i.e., addressing H. Simon's "bounded rationality" problem [22]). Abstracted information can also form a stable variable for the decision-making of micro-entities, changing slowly as to reduce the uncertainty that micro-entities have about the state of the (abstracted) macro-entity [7]. This allows multi-scale systems to accommodate increasing levels of complexity (which does not mean that higher levels are more complex, only that the system as a whole can become more complex – able to handle more information flows and coordinate in the face of more changes).

VII. How to Quantify Scales

A. Quantifying Abstraction

Various approaches may increase the abstraction of higher scales, from lower abstractions at lower scales – e.g., via *sampling* (at every granule interval, e.g. a unit area for space) or *aggregating* (over each granule interval, e.g., by averaging). This results in an abstraction of information from lower scales (micro) to higher scales (macro), where information details about the micro scale are lost at the macro scale. But how can we measure this information gap, or abstraction, between scales?

We propose using the notion of information *abstraction entropy* to quantify information loss between micro and macro scales. The abstraction entropy of a macro-entity's value is a function of the number of micro-state combinations that

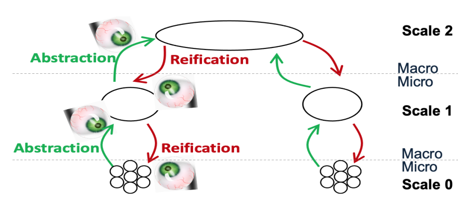

Fig. 2. Multi-scale feedback system: observers are internal system processes

could result in that macro-value (possibly weighted by the occurrence probabilities of these micro-state combinations). The function must be monotonically increasing with the number of micro-state combinations. Similarly to Boltzman's equation for ideal gases, this function can be logarithmic: $S = k_B * \ln(W)$, with S the entropy, k_B Boltzman's constant and W the number of micro-states corresponding to the gas' macro-state.

E.g., consider a macro-variable that takes the sum value of four binary micro-variables (see decentralised task-distribution in [4]). If that macro-value took value 2, then its abstraction entropy would be a function of the six possible combinations of micro-values that could produce that sum: {0,0,1,1}, {1,1,0,0}, {0,1,0,1}, {1,0,1,0}, {1,0,0,1}, or {0,1,1,0}. Similarly, consider that the macro-entity samples the sum value every 2 time steps versus every 10 steps. Then the less frequently sampled sum would include micro-variable information across a larger interval of time and thus would have higher abstraction entropy compared to the case of more frequent sampling. Hence, the abstraction entropy measures *the amount of uncertainty about the micro-scale that is hidden at the macro-scale*.

B. Value Abstraction Entropy

We suggest formalising these ideas via entropy formulae from statistical physics (Boltzmann Equation[4]) and Shannon's Mathematical Theory of Communication (MTC) [21]. We define abstraction entropy for a macro-variable's value based on the values of the micro-variables that it is abstracted from. Consider N micro-variables $\mathbf{x} = (x_0, x_1, .., x_{N-1})$ which are abstracted via a function f_a to a macro-variable $y = f_a(\mathbf{x}) \in \{v_k | k = 0..K-1\}$, where K is the number of distinct values in the range of f_a. Denote the macro-variable's *value abstraction entropy* as $S(v_k)$, which for binary variables is defined by:

$$S(v_k) = \log_2 W(v_k) \tag{1}$$

Here, $W(v_k)$ is the number of combinations of micro-variable values that result in $f_a(\mathbf{x}) = v_k$. Consider the previous example of four binary micro-variables $x_i \in \{0,1\}, i = 0..3$, occurring with equal probability and a sum abstraction function leading to a macro-variable $y = \sum_{i=0}^{3} x_i$. Then for each macro-value $y \in \{0,1,2,3,4\}$:

- $S(0) = S(4) = \log_2 1 = 0$
- $S(1) = S(3) = \log_2 4 = 2$
- $S(2) = \log_2 6 = 2.585$

Note that these values are the same irrespective of the number of intermediate scales between the bottom and top scales considered above (assuming perfect micro-macro abstraction information) – to be extended in future work. Generalising to N binary micro-variables, the number of combinations of micro-values resulting in any $V_{sum} \in \{0, 1, .., N\}$ is:

$$W(V_{sum}) = \frac{N!}{V_{sum}! * (N - V_{sum})!} \tag{2}$$

Hence, the value abstraction entropy for each V_{sum} is:

$$S(V_{sum}) = \log_2 W(V_{sum}) \tag{3}$$

[4]https://scienceworld.wolfram.com/physics/BoltzmannEquation.html

Similarly, if abstraction were achieved via a max function, $y = \max_{i \in \{0..N-1\}}(x_i) \in \{0,1\}$, then $W(0) = 1$ and $W(1) = 2^N - 1$. Thus there are two possible values for S, $S(0) = 0$ and $S(1) = \log_2(2^N - 1)$.

The definition of $S(v_k)$ can be extended to include cases in which the macro-scale operates at a slower rate than the micro-scales. First consider the situation where a macro-entity samples the micro-values every J cycles before applying the abstraction function, leading to the non-sampled micro-values being lost. For N binary micro-variables, the number of combinations that result in v_k is $W(v_k) \times 2^{N(J-1)}$ and the value abstraction entropy of the macro-values becomes:

$$S_J(v_k) = N(J-1) + \log_2 W(v_k) \tag{4}$$

Considering the previous example of four binary micro-variables with the sum abstraction function, the value abstraction entropy for each value of $y \in \{0,1,2,3,4\}$ when sampled every 2 cycles are:

- $S_2(0) = S_2(4) = \log_2 16 = 4$
- $S_2(1) = S_2(3) = \log_2 64 = 6$
- $S_2(2) = \log_2 96 = 6.585$

Similarly, when sampled every 3 cycles the values are:

- $S_3(0) = S_3(4) = \log_2 256 = 8$
- $S_3(1) = S_3(3) = \log_2 1024 = 10$
- $S_3(2) = \log_2 1536 = 10.585$

Note that the macro-value abstraction entropy becomes higher with larger sampling intervals due to more information loss.

Now consider the situation where micro-variables are averaged over the J cycles before the abstraction function is applied. Denoting this value abstraction entropy as $S_{J,ave}(v_k)$, one can expect that $S(v_k) \leq S_{J,ave}(v_k) \leq S_J(v_k)$. However, there are more values for v_k due to the averaging. E.g., for N=4, J=2, $y \in \{0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4\}$, the value abstraction entropy is:

- $S_{2,ave}(0) = S_{2,ave}(4) = \log_2 1 = 0$
- $S_{2,ave}(0.5) = S_2(3.5) = \log_2 8 = 3$
- $S_{2,ave}(1) = S_{2,ave}(3) = \log_2 28 = 4.807$
- $S_{2,ave}(1.5) = S_{2,ave}(2.5) = \log_2 56 = 5.807$
- $S_{2,ave}(2) = \log_2 70 = 6.129$

Note that $S(v_k) \leq S_{2,ave}(v_k) \leq S_2(v_k)$ for $v_k = 0, 1, 2, 3, 4$ (no comparison is possible for $0.5, 1.5, 2.5, 3.5$). Also note that $S_{2,ave}(0) = S_{2,ave}(4) = 0$ because there is no uncertainty about the values of the micro-variables, even at the in-between samples. All micro-values must be 0 (or 1) for $y = 0$ (or 4).

C. Variable and Micro-Macro Abstraction Entropy

We now aim to quantify the overall entropy of a macro-variable y, considering all its values v_k and their underlying micro-values x_i. We employ the Shannon entropy to quantify the macro-variable's entropy (as a measure of its values' variability). We then use the difference between this 'macro' entropy and the 'micro' entropy of the underlying micro-variables to quantify the variable abstraction entropy (i.e., entropy difference between the macro and micro scales).

Thus, considering N binary micro-variables connected to a macro-variable via an abstraction function $y = f_a(\mathbf{x})$, we first define *variable entropy* as:

$$H_{f_a}(y) = -\sum_{k=0}^{K-1} p(v_k)\log_2 p(v_k) \tag{5}$$

with $p(v_k)$ the probability that $y = v_k$.

When $N = 4$ and f_a is the sum function, $y = \sum_{i=0}^{3} x_i$ and $y \in \{0, 1, 2, 3, 4\}$, and assuming each x_i has value 0 or 1 with equal probability, we have:

- $W(0) = W(4) = 1$, hence $p(0) = p(4) = \frac{1}{16}$
- $W(1) = W(3) = 4$, hence $p(1) = p(3) = \frac{4}{16} = \frac{1}{4}$
- $W(2) = 6$, hence $p(2) = \frac{6}{16} = \frac{3}{8}$

Thus, the variable entropy for the sum function becomes:

$$\begin{aligned} H_{sum}(y) &= -\frac{2}{16}\log_2\frac{1}{16} - \frac{2}{4}\log_2\frac{1}{4} - \frac{3}{8}\log_2\frac{3}{8} \\ &= \frac{1}{8}*4 + \frac{1}{2}*2 + \frac{3}{8}*1.415 = 2.03 \end{aligned} \tag{6}$$

For a scale with several independent variables, we can define the *scale entropy* as the sum of $H_{f_a}(y)$ taken over all variables at that scale. E.g., given a scale with multiple variables $\mathbf{y} = (y_0, y_1, .., y_{N-1})$, its scale entropy is given by: $H_{f_a}(\mathbf{y}) = \sum_{i=0}^{N-1} H_{f_a}(y_i)$. In the above example, assuming four variables x_i each having value 0 or 1 with equal probability, the scale entropy at the micro-scale is:

$$\begin{aligned} H(\mathbf{x}) &= -\sum_{i=0}^{3}\left[p(0)\log_2 p(0) + p(1)\log_2 p(1)\right] \\ &= -\sum_{i=0}^{3}\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 4 \end{aligned} \tag{7}$$

Note the absence of the "f_a" subscript on H due to the lack of an abstraction function generating the x_i.

We define the *variable abstraction entropy* as:

$$\Delta H_{sum}(\mathbf{x}, y) = H(\mathbf{x}) - H_{sum}(y) = 4 - 2.03 = 1.97 \tag{8}$$

Similarly, for the max abstraction function $y = \max_{i\in\{0..N-1\}}(x_i)$,

$$H_{max}(y) = -\frac{1}{16}\log_2\frac{1}{16} - \frac{15}{16}\log_2\frac{15}{16} = 0.34$$

We notice here that $H_{max}(y) = 0.34$ is considerably smaller than $H_{sum}(y) = 2.03$. This makes sense as the distribution of macro-values is more varied for the sum (higher entropy) than for the max (lower entropy). The variable abstraction entropy when using max abstraction function is:

$$\Delta H_{max}(\mathbf{x}, y) = H(\mathbf{x}) - H_{max}(y) = 4 - 0.34 = 3.66 \tag{9}$$

We note that the value 3.66 obtained via max is larger than the value 1.97 obtained via sum. This makes sense as max "loses" more information from the micro scale than sum does (i.e., macro variable y only takes two values when using max and four values when using sum).

Importantly, the above formulae only consider abstractions f_a occurring at an instant t. If a macro-value were abstracted from micro-values collected over an interval Δt, its abstraction entropy would increase with that interval's length, depending on the frequency of micro-value changes during that interval. To asses this latter case, we again consider interval sampling and averaging aggregates. In the worst case, with independent micro-values across cycles, the micro-scale entropy is:

$$H(\mathbf{x}) = -JN\left[p(0)\log_2 p(0) + p(1)\log_2 p(1)\right] \tag{10}$$

where N is the number of binary micro-variables and samples taken every J cycles. At the macro-scale, the values $H_{f_a}(y)$ remain the same when sampling occurs because, although the number of occurrences resulting in a given v_k increases, the probability of those occurrences is unchanged. However, if averaging is used, the number of occurrences and their probability change, as well as the number of possible macro-values. Specifically, for N=4 binary micro-variables and a sum abstraction, the abstraction entropy is given in Table I.

Sampling	$H(\mathbf{x})$	$H_{sum}(y)$	$\Delta H_{sum}(\mathbf{x}, y)$
Every cycle	4	2.03	1.97
Every 2 cycles	8	2.03	5.97
Every 2 cycles with averaging	8	2.54	5.46
Every 3 cycles	12	2.03	9.97
Every 3 cycles with averaging	12	2.84	9.16

TABLE I
ABSTRACTION ENTROPY (N=4, SUM f_a, $J \in \{1, 2, 3\}$)

Note that obtained values conform to expectations, as values of variable abstraction entropy obtained with less frequent sampling are larger compared to more frequent sampling (1.97 vs. 5.97 vs. 9.97). Also, averaging over each interval results in lower value (5.46 vs. 5.97 and 9.16 vs. 9.97). These results confirm that more information loss between levels leads to larger variable abstraction entropy values.

D. Quantifying Data and Information

The micro-macro abstraction entropy is different from (i) the amount of data available at each scale (an absolute measure) and (ii) the amount of information cycling across scales. When measuring information, Shannon's MTC quantifies the information content of a signal travelling from a source to a destination via the degree of "surprise" of the message (the higher the probability of that message, the lower its information content). The channel capacity, then, sets a limit for the maximum amount of information that can be carried through a medium per time.

However, this differs from the amount of *semantic* information [9]: while transmission of data across channels is independent of its interpretation by a receiver, an observer-dependent view focuses on information as *data + meaning*.

Hence, in terms of data quantification, we may employ Shannon's theory or simply consider the amount of resources required to encode, store, transmit, and process the data (e.g., [5]). Going beyond syntactic data, we are interested in the amount of semantic information cycling across scales.

For autonomic and self-integrating systems, it is essential to estimate acquired information in terms of its effects on the system's adaptation (with respect to its goals). Considering information in terms of change (in the receiver), semantic information in multi-scale systems is tied to *the amount of change generated by the received information with respect to an entity's current state and to the system goal*. Hence, we cannot say whether there is *more* or *less* information at a scale rather than another, but can only compare changes with respect to one another (at a scale). The magnitude of change at each scale depends on:

1) The amount of data received and processed (determined by the source's available data, the channel's capacity and the receiver's processing capacity);
2) How different the data received is from what is already known/stored (Shannon's information content);
3) The capacity of each entity to change its state, towards its goals (different from channel & processing capacity).

The amount of semantic information will also depend on the time over which this change takes place – we suggest that the amount of information is tied to the amount of change produced over the amount of time taken for the change. These initial considerations will be further formalised in future work.

VIII. Summary and Future Perspectives

This short paper aimed to clarify notions of scale and multi-scale as a conceptual basis for a theory of multi-scale (feedback) systems. It proposed an information-oriented approach to help transferability across application domains. We modelled systems as information flows, or change sequences, merging and splitting to form higher or lower scales of information abstraction. We also suggested several ways to quantify multi-scale information flows: abstraction entropy (micro-macro gap); data amount (encoding resources); and information amount (incurred change magnitude).

Future work will further refine and formalise these measuring concepts, and study their applicability to specific domains. A difficulty is that abstracted information, e.g., models, may be hard (or impossible) to quantify. A key related question is how to determine the viable or optimal abstractions, or scales, for each system, considering its particular context and goals.

Acknowledgment

We are grateful to Dr. P. Nelson and Dr. C. Diaconescu for the valuable discussions on entropy related concepts.

References

[1] T.FH Allen and T.B Starr. *Hierarchy: perspectives for ecological complexity*. Univ. of Chicago Press, 2017.

[2] KJ Astrom and RM Murray. *Feedback Systems-An Introduction for Scientists and Engineers, v 2.10 c*. 2010.

[3] L. Brillouin. "The negentropy principle of information". In: *J. of Applied Physics* 24.9 (1953), pp. 1152–1163.

[4] A. Diaconescu, L.J Di Felice, and P Mellodge. "Exogenous coordination in multi-scale systems: How information flows and timing affect system properties". In: *Future Gen. Comp. Sys.* 114 (2021), pp. 403–426.

[5] A. Diaconescu, L.J Di Felice, and P. Mellodge. "Multi-scale feedbacks for large-scale coordination in self-systems". In: *Intl. Cnf. on Self-Adaptive and Self-Organizing Systems (SASO)*. IEEE. 2019, pp. 137–142.

[6] E. Durkheim. *Suicide: A study in sociology*. 2005.

[7] J.C Flack et al. "Timescales, symmetry, and uncertainty reduction in the origins of hierarchy in biological systems". In: *Evolution cooperation and complexity* (2013), pp. 45–74.

[8] J.C. Flack. "Coarse-graining as a downward causation mechanism". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 375.2109 (2017), p. 20160338.

[9] L Floridi. "Semantic Conceptions of Information (stanford. library. sidney. edu./au/entries/computer-science)". In: *First published Wed Oct* 5 (2005).

[10] L. Floridi. "The method of levels of abstraction". In: *Minds and machines* 18.3 (2008), pp. 303–329.

[11] D.H Hubel and T.N Wiesel. "Ferrier lecture-Functional architecture of macaque monkey visual cortex". In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 198.1130 (1977), pp. 1–59.

[12] E. Jablonka. "Information: Its interpretation, its inheritance, and its sharing". In: *Philosophy of science* 69.4 (2002), pp. 578–605.

[13] FL Jones. "Micro-macro linkages in sociological analysis: theory, method and substance". In: *The Australian and New Zealand j. of sociology* 31.2 (1995), pp. 74–92.

[14] A. Koestler. "The ghost in the machine." In: (1968).

[15] P. Mellodge, A. Diaconescu, and L.J. Di Felice. "Timing Configurations Affect the Macro-Properties of Multi-Scale Feedback Systems". In: *ACSOS* (2021).

[16] H.H. Pattee. *Hierarchy theory*. Braziller., 1973.

[17] D. Pumain et al. *Hierarchy in natural and social sciences*. Springer, 2006.

[18] C. Rovelli. *The order of time*. Riverhead, 2019.

[19] A.P Sage and C.D Cuppan. "On the systems engineering and management of systems of systems and federations of systems". In: *Information knowledge systems management* 2.4 (2001), pp. 325–345.

[20] G.N Saridis. "Machine-intelligent robots: A hierarchical control approach". In: *Machine Intelligence and Knowledge Eng. for Robotic Apps.* Springer, 1987.

[21] C.E Shannon. "A mathematical theory of communication". In: *The Bell system technical journal* 27.3 (1948).

[22] H.A Simon. "Bounded rationality". In: *Utility and probability*. Springer, 1990, pp. 15–18.

[23] H.A. Simon. "The architecture of complexity". In: *Facets of systems science*. Springer, 1991, pp. 457–476.

[24] D.C Van Essen and J.HR Maunsell. "Hierarchical organization and functional streams in the visual cortex". In: *Trends in neurosciences* 6 (1983), pp. 370–375.

[25] J. Wu. "Hierarchy theory: an overview". In: *Linking ecology and ethics for a changing world* (2013).