Ecological Complexity 31 (2017) 34–49

Contents lists available at ScienceDirect

Ecological Complexity

journal homepage: www.elsevier.com/locate/ecocom

Original Research Article

Emergence and complex systems: The contribution of dynamic graph theory

Jacques Gignoux [a,*], Guillaume Chérel [b], Ian D. Davies [c], Shayne R. Flint [d], Eric Lateltin [a]

[a] Institute of Ecology and Environmental Sciences, CNRS UMR 7618, 4 place Jussieu, bât. 44-45, CC 237, 75005 Paris, France
[b] Institut des Systèmes Complexes Paris Ile-de-France, 113 rue Nationale, 75013 Paris, France
[c] Fenner School of Environment and Society, The Australian National University, Canberra ACT 2601, Australia
[d] Research School of Computer Science, The Australian National University, Canberra ACT 0200, Australia

ARTICLE INFO

Article history:
Received 31 August 2016
Received in revised form 31 January 2017
Accepted 20 February 2017
Available online 16 March 2017

Keywords:
Hierarchy
Ontology
Feedback loop
Causality
Computational irreducibility

ABSTRACT

Emergence and complex systems have been the topic of many papers and are still disputed concepts in many fields. This lack of consensus hinders the use of these concepts in practice, particularly in modelling. All definitions of emergence imply the existence of a hierarchical system: a system that can be observed, measured and analysed at both macroscopic and microscopic levels. We argue that such systems are well described by mathematical graphs and, using graph theory, we propose an ontology (*i.e.* a set of consistent, formal concept definitions) of dynamic hierarchical systems capable of displaying emergence. Using graph theory enables formal definitions of system macro-state, micro-state and dynamic structural changes. From these definitions, we identify four major families of emergence that match existing definitions from the literature. All but one depend on the relation between the observer and the system, and remind us that a major feature of most supposedly complex systems is our inability to describe them in full. The fourth definition is related to causality, in particular, to the ability of the system itself to create sources of change, independent from other external or internal sources. Feedback loops play a key role in this process. We propose that their presence is a necessary condition for a hierarchical system to be qualified as complex.

1. Introduction

Interest in the concepts of *complex system* and *emergence* has been pursued over many years in many fields. In such circumstances, it is inevitable that conflicting definitions will arise. When attempting a definition, it is important to be wary of including concepts that are either themselves poorly defined or are merely correlated with the concept at hand (Jax, 2007). With these caveats in mind, we propose a formal definition of a system from which we can formalise the circumstances under which emergence may arise.

In doing this, we build an *ontology* of useful and rigorous concepts related to emergence. An ontology (in computer science) is a set of formal definitions of concepts and their relationships, that can lead to automatic processing for the construction of formal grammars and software (Guarino, 1995). It is our hope that this methodology will underpin a broadly applicable clarification of these concepts (*e.g.* an application to ecology in Gignoux et al., 2011). Following Jax (2007), we begin by using mathematical notation to define the concept of a 'system'. Our notation provides for, but does not impose, the possibility of emergent properties, based on the commonalities between most definitions of emergence. In so doing, we extract generic properties of 'systems with emergence'.

Despite fundamental differences, all definitions of emergence share a common assumption: emergence arises only in systems that can be described at both *macroscopic* and *microscopic* levels (de Haan, 2006; De Wolf and Holvoet, 2005; Bedau, 2003). A system with such properties is usually called a *hierarchical system* (Allen and Hoekstra, 1992; Ahl and Allen, 1996; O'Neill et al., 1986). In contrast, a *non-hierarchical system*, also called *atomic system*, is one which cannot be divided into sub-systems; it is atomic in the sense that we have no knowledge of a microscopic representation (see Definition 24).

How can we formally define a hierarchical system in a generic way? In one of its most commonly accepted definitions (Carnot, 1824), a system is *'the part of the world under consideration for a particular purpose'*. Implicit in this definition is the existence of an observer, someone or thing for which a part of the world is extracted for consideration to some end. The ecosystem, as initially

* Corresponding author.
E-mail address: jacques.gignoux@upmc.fr (J. Gignoux).

http://dx.doi.org/10.1016/j.ecocom.2017.02.006
1476-945X/

defined by Tansley (1935), falls within the scope of this definition, just as do, for example, thermodynamic systems and systems of social organisation. In the field of systems thinking, Jordan (1981) finds that nothing more specific can be said in defining the term 'system' (the fundamental concept in the author's discipline) other than that *'a system is composed of identifiable entities and their relationships'*. This definition is just as applicable to concrete objects as it is to *virtual* or *conceptual* objects. For emergence to occur, the system must be characterised as *hierarchical*, in the sense that we can provide both a macroscopic and a microscopic description. We will therefore define a hierarchical system as *an object composed of components in interaction*. This is close to some definitions of a complex system, but we make no assumption about emergence as this is precisely what we wish to explore.

A system comprising components and their interactions is well described by a mathematical graph (Diestel, 2000; Gross and Yellen, 1999). A graph is a set of nodes connected by edges. We propose to represent a hierarchical system as a mathematical graph: the 'interacting components' that produce the 'microscopic state' of the system are the nodes, the edges represent interactions, while the system as a whole is represented by the graph. Although the hierarchical relation between the graph and its components is not explicit at this stage, this representation allows us to consider both a *macroscopic* view of the system – the graph as a whole – and a *microscopic* view – the list of all its components *and their interactions*.

2. Formal definitions for a hierarchical system: an ontology

We first provide a minimal set of mathematical definitions to describe a system without any a priori knowledge of emergence.

2.1. The system

We postulate that a hierarchical system can be represented as a graph. We call the *world* $\mathcal{W}$, that set of objects from which an observer draws a subset to build a system *for some purpose*. Components of the system are defined as objects $\rfloor \in \mathcal{W}$, and interactions as relations between any two components of $\mathcal{W}$.

Proposition 1. *A hierarchical system S is defined as the graph:*

$$S := (C, R, \gamma)$$

where C is the set of components (nodes) of the system:

$$C := \{c_u\}_{u \le n_c < \infty}, \quad c_u \in \mathcal{W}$$

R is the set of relations (edges) between components of the system:

$$R := \{r_v\}_{v \le n_r < \infty}, \quad r_v \in \mathcal{W}^2$$

and γ is the incidence function, which assigns a relation to a pair of components:

$$\gamma : R \to C \times C$$

$$r_v \to (c_i, c_j)_{i \le n_c, j \le n_c}$$

n_c is the number of components and n_r the number of relations of the system; $\mathcal{W}^2$ is the set of applications from $\mathcal{W}$ to $\mathcal{W}$.

We make no assumption as to the type of graph used to represent S. It can be directed, undirected, a multigraph or any other kind of graph, hence the need for an explicit incidence function.

Where it may be ambiguous, we subscript sets C, R and function γ by the graph to which they belong.

Fig. 1 gives examples of systems represented as graphs.

For later simplification, its is convenient to define:

Definition 1. Components c_u and relations r_v are called *elements* of the system *S*. We denote them by $e_w \in E$, with $E = C \cup R$ and the

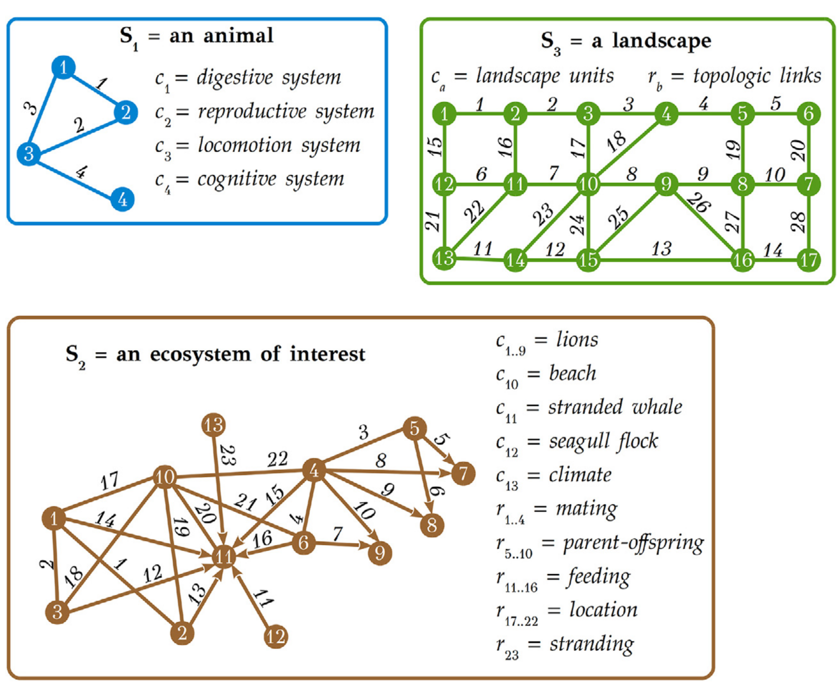

Fig. 1. Three different examples of systems represented as graphs using Proposition 1. Circles denote system components c_u and lines between them denote relations r_v.