How can we test for artificial life?

Hector Zenil, Gordon Ball and Jesper Tegnér

Department of Medicine, Karolinska Institutet, Unit of Computational Medicine
Centre for Molecular Medicine, Solna, Stockholm, Sweden.
{hector.zenil,gordon.ball,jesper.tegner}@ki.se

A problem in artificial systems modelling is related to the estimation and control of how many of the intended target self-assembled objects one can expect from a particular self-assembling system (this problem is ubiquitous in the biochemical sciences). The observation that "self-assembly and computation are linked by the study of mathematical tiling" Rothemund (2011) has produced a step change in the way in which molecular self-assembly is dealt with, for example using Wang tiles. These connections between self-assembly and computation have been explored before (see Terrazas et al. (2013)).

We address the question of what kind of test could be implemented to establish whether an artificial system is living or not with a computability and programmability test as advanced in Zenil (2013). We claim that necessary conditions for life are non-linear behavioural variability and high and rapid sensitivity to external stimuli. The approach can be applied to all kind of artificial systems modelling living systems and it finds similarities with Alan Turing's approach to the question of intelligence and his imitation game.

The idea of ultimately programming a system is related to the non-linear variability of the system. Along these lines, we advance a concept of programmability as a combination of behavioural change and external control. We propose a Turing-test type approach to the problem of testing for artificial life where in the place of the interrogator there is a complexity evaluator (e.g. a lossless compression algorithm testing for Kolmogorov randomness). A programmability model with associated information-theoretic metric is advanced. We look into evaluating, classifying and discriminating biological models from non-biological models, where an important overlapping is expected, with a case study of simulations of biological models taken from the BioModels Database (`http://www.ebi.ac.uk/biomodels-main/`), a centralized database of curated quantitative models of biochemical interest whose dynamical space, time-evolution and model's reaction to their "environment" (sensitivity) is studied, leading to questions such as the robustness of the systems to drastic changes. Similarity metrics and information distances are applied and results are presented in the direction of characterisations of the behaviour of physical and natural systems, allowing the classification of qualitative properties and the assessment of simulation results in terms of algorithmic information content with special focus on phase transitions, clustering distances, variability and parameter discovery.

The framework promises to constitute an alternative testing approach based on a system capability to react to external stimuli and transfer information as reflected in the proposed measures of programmability. The programmability measure indicates the susceptibility of a system to be (efficiently) programmed where we can ask questions such as whether programmability differences originate from either (a) a structural difference in the modelling equations, or (b) differences in how much parameters are allowed to vary, and thereby having a greater/smaller flexibility to respond to external stimuli.

The proposed model is also intended to help accelerate the iterative process in synthetic biology from *in silico* simulation to *in vivo* testing, potentially contributing to a workflow for generating hypothesis of biomedical interest similar to Gomez-Cabrero et al. (2011).

References

Gomez-Cabrero, D., Compte, A., and Tegnér, J. (2011). Workflow for generating competing hypothesis from models with parameter uncertainty. *Interface Focus*, 1:438–449.

Rothemund, P. (2011). Using lateral capillary forces to compute by self-assembly. *Proceedings of the National Academy of Sciences*, 97(3):984–989.

Terrazas, G., Zenil, H., and Krasnogor, N. (2013). Exploring programmable self-assembly in non-dna based molecular computing. *Natural Computing*, (submitted).

Zenil, H. (2013). What is nature-like computation? a behavioural approach and a notion of programmability. *Philosophy & Technology*, (published online Springer-Link).