Evolution-in-Materio of a dynamical system with dynamical structures

Odd Rune Lykkebø[1], Gunnar Tufte[1]

[1]Norwegian University of Science and Technology
lykkebo@idi.ntnu.no, gunnart@idi.ntnu.no

Abstract

Evolution-in-Materio aims to exploit real-world physics of materials to achieve computation by a combination of external stimulus and interpretation of the state of materials through measurements and observations. In a majority of Evolution-in-Materio work the dynamics of the material is filtered out, or the problem is defined in a way that the sought solution is a point attractor. In this work we explore the dynamics of materials. Within the assumption that suited materials include rich behavior emerging from the underlying physical processes there should be observable behavior similar to Dynamical Systems with Dynamical Structures ($(DS)^2$). Such behavior result in systems with a possibility of inducing perturbations to their own dynamics. Further, the importance of the *observation level* used when observing and interpreting the state of the materials is discussed and related to dynamics in Evolution-in-Materio systems.

Introduction

Evolution-In-Materio (Miller et al., 2014) (EIM) can be seen as a method to explore unconventional computation, i.e. a computer operating outside of the traditional Turing/von Neumann (Turing, 1937; von Neumann, 1993) computational model and architecture, exploiting the power of evolution, i.e. Computer Controlled Evolution (CCE) to manipulate a physical system to search for regimes where the intrinsic properties of materials provide useful computation.

The concept of EIM is, as stated by Miller et al. (2014): ”to exploit the intrinsic properties of materials, or “computational mediums”, to do computation, where neither the structure nor computational properties of the material needs to be known in advance (Miller and Downing, 2002). In this way evolution is a bottom-up design process that can exploit natural physical processes to do useful computation.”

Herein the bottom-up design concept is further investigated so as to gain a deeper insight toward exploiting physical systems such as materials for computation: ”where neither the structure nor computational properties of the material needs to be known in advance”. Both the structure and the intertwined underlying physics leading to computation are products of the bottom-up design approach taken.

When a bottom-up process such as evolution acts on physical systems, there may be intrinsic processes in the computational medium that influence the computational function as a result of underlying properties resulting in a non-static structure and thereby non-static functionality, i.e a two way coupling between dynamic structure and functionality. A system with the possibility of inducing perturbations to their own dynamics as a function of their system states, enables state space trajectory changes and topological reconfigurations of the state space (Omholt, 2013).

In this paper we show that such behaviours, present in living systems, can also be found in EIM systems making EIM a physical realisations of Dynamical Systems with Dynamical Structures ($(DS)^2$) (Spicher et al., 2004). In such systems state transition functions and the set of state variables can change over time.

Reaction diffusion systems as in Adamatzky (2009) work exploit massively parallelism of state updates in growing patterns where information processing take place. The reaction-diffusion computer is based on local interactions and change of spatial properties over time. Growth of patterns change the state and topological properties of the machine providing a similarity to the ($(DS)^2$) dynamical reconfigurations of the state space.

Further, it can be useful to compare the concept of EIM with morphological engineering (ME) Doursat et al. (2013). Central to ME is the concept of *agents* and the mechanisms involved in controlling them. EIM places less focus on the interaction of the individual agents, and more focus on exploitation of the emergent behaviours of the physical systems under study. Though local interaction between smaller parts of the physical matter (i.e. electro-chemical interaction between molecules or concentrations of chemicals) is crucial, EIM attempts to take a step away from the agent-focus and consider emergent properties a primary unit of study. However, one of the central aspects of ME, i.e. ”endowing physical systems with information” Doursat et al. (2013) does capture EIM. 'Invisible' dynamics of physical systems can be manipulated without direct access to the individual agents' rule sets by giving the systems the ability to filter or react to information and energy presented through physical interfaces.

In this article, an EIM system is observed in a $(DS)^2$ system view. The results show that such a view is applicable and can be used to gain further insight in the working of EIM systems. As a result EIM may share several properties with self-organizing systems. Herein artificial developmental systems (Kumar and Bentley, 2003) are used as a starting point for exploring and exploiting computational mediums where computation is a product of underlying dynamical physical processes.

An implication of "exploiting the intrinsic properties of materials" is that self-organizing processes and the resulting computation do not necessary comply with our mostly used computational paradigm of digital computers. EIM can be seen as a hybrid variant of Analog Computation (AC) (MacLennan, 2007). In AC, the fact that a mathematical function provides a model of the *observed* behaviour of a physical system is used 'conversely'; a physical system can be used to calculate a mathematical function since a physical system can be parameterized and adjusted to match a large number of mathematical functions. Configurable Analogue Processors (CAP) or Field Programmable Matter Arrays (FPMA) (Miller and Downing, 2002) are recent example of this paradigm. The hybrid approach of EIM include the computational matter, e.g. CAP or FPMA, in a mixed signal system using a digital computer to configure and communicate with the material.

This is an approach that enables the computational power of the material with the ease of programmability of digital computers (Broersma et al., 2012). In a hybrid approach observability is a core issue. Ensuring that the data from the material are observable and sound without using more computational power for the observation then the actual computation (Bremermann, 1962). A practical implication of observability is that there is a need for choosing what the smallest possible change is to be on the top-level. Top-level being the level that the observable computational result(s) emerge from the underlying physical processes.

Shannon (1941) provided an early theoretical model of analogue computers, the General Purpose Analog Computer (GPAC). Though application of the GPAC model on EIM seems un-natural since it relies on chaining together components (reminiscent of agents with particular tasks) and defining bigger expressions out of this. Neural networks is yet another potential model and link between the physical EIM system and a theoretical model– this link is further discussed in Broersma et al. (2012).

Figure 1 shows an EIM system using a digital computer to host an EA to configure a material for computation. The material operates in the analogue/physical domain and the computer responsible for input/output mapping and configuration will operate in the digital domain.

As stated, our focus is to explore EIM in a $(DS)^2$ view.

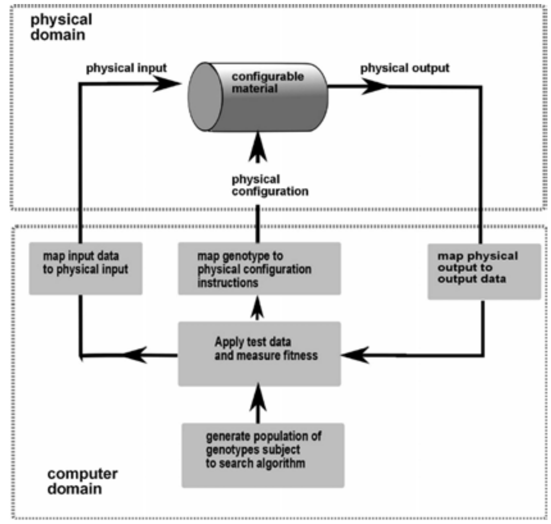

Figure 1: Principle of EIM using a hybrid approach. Taken from Miller et al. (2014).

In Figure 1 the configurable material is the self-organising system. The material inhibits dynamical properties and respond to perturbations from the input and configuration signals. To observe dynamical properties, the trajectory of the system is used as a quantifiable measurement of behaviour. In artificial development similar measurements have been used as a measurement of evolved complexity (Nichele and Tufte, 2013) and to show intertwined influence of structure and computation ($(DS)^2$) on artificial developmental systems (Tufte, 2009).

The systems herein are all observed at an electrical level. The underlying physical (and electrical) processes may change over time on the microscopic level, however our observations is on the 'digital level'. As such, the goal herein is to be able to exploit the power of EIM without a high cost in ensuring the correctness of the observation, i.e. a underlying rich physical system observed in the constrained digital domain.

Background

Evolution-in-materio extended to $(DS)^2$

Gordon Pask's pioneering work in EIM(Pask, 1959) is an interesting piece of work if viewed in a $(DS)^2$ setting. Pask observed (by eye) and evaluated (by ear) the growth of neural like structures using an electrochemical device made of a dish with electrodes covered by a metal salt solution. By adjusting the current between electrodes Pask was able to grow iron connections that responded to different frequencies. Pask's system show the principles of a $(DS)^2$ EIM system. The growth of the connections depend on current, when current passes the connection grow. By adjusting the current in different regions a structure emerges and are adjusted toward the evaluator's (here Pask) goal. The

(a) Block diagram of the Mecobo hardware interface.

(b) Picture of the Mecobo motherboard with mixed signal daughter board.

(c) The material bay filled with salt crystals (red arrow).

Figure 2: Overview of the Mecobo hardware interface.

growth of the connections are governed by self-organization and perturbations. The system is constantly changing making the response to perturbations depending on the systems present state. In Pask's work this property of change by self-organization and perturbations was exploited toward the goal of growing "an artificial ear".

In the work of Clegg et al. (2014) a material of Carbon Nano Tubes (CNT) was exploited to solve the travelling salesperson problem by evolving statical configurations that manage to solve the problem. i.e. the material's dynamic properties was filtered out by allowing the system to reach a steady state, or point attractor. Thompson's experiments using a Field Programmable gate Array (FPGA) as a material (Thompson et al., 1999) evolved a static configuration defining the internal circuit architecture. The input signal (similar to Pask's) was two frequencies that changed the output value depending on the input frequency (a frequency discriminator). The static configuration in Thompson's FPGA experiments does not allow for dynamic structures. The input signal perturbed the system to an binary observable output. However, the underlying dynamics (internal state transitions) for the system was untested.

In Harding and Miller's EIM work utilizing Liquid Crystal Displays (LCDs), the material is viewed as a type of static device with an evolved configuration that enable the LCD to compute, e.g. a frequency discriminator (Harding and Miller, 2004). However, the LCD is not a static device. The behaviour change if it is disconnected and reconfigured. To regain previous behaviour a short re-evolution was required.

Systems that explicitly exploit dynamic structures such as slime moulds (Adamatzky, 2016) or the combination of CNTs and liquid crystals (Massey et al., 2015b) show a clear connection between dynamic structure and computation similar to a $(DS)^2$ system.

As in Pask's work, the work on explicit dynamic structures and the shown change in response on configuration for the LCD it can be argued that a closer look at EIM systems as a $(DS)^2$ system are fruitful as to achieve more complex behaviour in such computational medium. More over, including more knowledge of the underlying bottom-up processes ($(DS)^2$) enables an increased understanding and insight in evolutionary exploitation of EIM systems toward complex computational tasks.

Materials

Recently the NASCENCE (NAnoSCale Engineering for Novel Computation using Evolution) project (Broersma et al., 2012) has provided a variety of material samples based on CNTs (Massey, 2013), mix of liquid crystals and CNTs (Massey et al., 2015b) and gold nano particles (Boses et al., 2015) for EIM research. Studies of Single walled carbon nano tubes(SWCNT) (Massey, 2013; Massey et al., 2015a) have shown that these CNTs have novel electrochemical properties and have the potential to do computation (Massey et al., 2015b).

The results of NASCENCE (e.g. (Mohid et al., 2014; Clegg et al., 2014; Massey, 2013)) have shown that computational results can be achieved without explicitly exploiting the dynamics of physical systems. Here the exploration of $(DS)^2$ expand our target behaviour of materials to a physical system with rich dynamic properties and complexity at many scales, which is a point frequently brought up in the complex systems literature. Often one divides a system into two broad categories relative to the observation level or scale, (Sayama, 2015) (Bar-Yam, 1997) (Fromm, 2004). Microscopic properties or behaviours are those that potentially give rise to emergent macroscopic properties. At different levels of observation, different macroscopic behaviours (such as emergence and self organization) exist, and thus the complexity of a system depends on the level of observation; as Simon (1962) writes, "How complex or simple a structure is depends critically upon the way in which we describe it."

A core idea in EIM is that it is should be possible to manipulate the microscopic behaviour of the material by providing input and 'configuration' energy (which potentially affects all scales of the system) such that the emerging or self-organizing behaviour is both observable and useful in terms of computation.

To further drive the point of the richness the materials used in the experiments are chosen to be very different. Single walled carbon nano tubes in a static physical configuration, i.e. electrical charge change cause dynamics, used and exploited for computation within the NACSENCE project show electrical observable response from underlying electrical networks exploited by evolution to emerge at the macroscopic level. As a second material common kitchen table salt, a material that is conductive when mixed with water. The crystalline form of salt is for the most part non-conductive since the ions are bound up in a crystal lattice structure, but by adding a tiny amount of water to the crystals conductivity is achieved. The salt structure is in contrast to the CNT material not static.

Method

The overall goal of EIM is to find methods and materials that can serve as complex computing systems. Methods should be capable of exploring and exploiting materials toward achieving useful computation, and materials should have inherent properties that enable this. The systems are physical systems, hence the material will operate in a real environment, and in particular, the environment will be part of the system. The main method is the bottom-up design approach of evolution. In a complexity setting evolution is argued to be a process that builds complexity (Holland, 2012). The building of complexity can also be considered as a case where the system learns to program itself, third of Brian Arthur's methods of complexity growth (Arthur, 1993). Further, the concept of growth of complexity fits well into a $(DS)^2$ setting; a system that evolve toward more complex dynamic behaviour by perturbations from the environment and the dynamics of the system itself (Nichele and Tufte, 2013).

An experimental approach is taken to investigate and explore the relation between EIM and $(DS)^2$. Exploiting evolution to configure materials with a behaviour that show induced perturbations to it's own dynamics. The experimental setting is in principle as shown in Figure 1. The experiments are designed to unveil the intertwined influence between the dynamics of underlying physics the input data and configu-

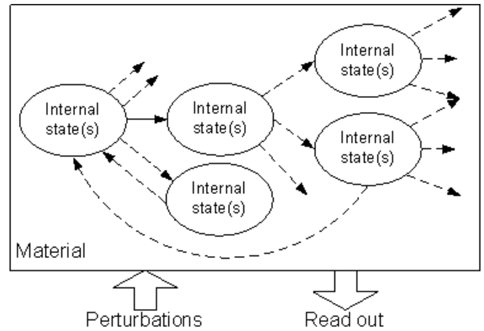

Figure 4: System view of the experiments. The material is considered as a $(DS)^2$ system capable of inducing perturbations to it's own dynamics. The external perturbations indicated as an input arrow include input data and configuration signals for the material. The output arrow indicates the external observation of the system state.

ration signals. To put the system in a $(DS)^2$ setting input and configuration signals are considered as external perturbations to the system. Figure 4 illustrates the experimental setting at a system level.

The material in the figure include a set of state transitions. The state transitions can be perturbed externally. Each state in the figure is observable through the read out signal. If the system inhibits $(DS)^2$ behaviour it should be possible to detect different trajectories (induced perturbations to it's own dynamics) whilst the external perturbation is unchanged, illustrated by dotted arrows. In the figure each state is as indicated one or several internal states (internal state(S)) as to illustrate the property of topological reconfigurations of the state space.

In literature on artificial (and biological) evolution (AE) many terms are used for breaking down the various parts of a search. In a physical and 'real' system such as the EIM-based ones it is not immediately clear where the boundaries go between these parts, such as *genotype*, *phenotype* and the *genotype-phenotype map*. The definition of these terms is context-dependant, and EIM mixes two contexts; the context of artificial evolution and the context of physical, real life systems. For the purposes of EIM it is sufficient when discussing these terms to simply note that when we discuss the *phenotype* of a system, we are talking about the observed entity interacting with the physical environment, e.g. the electrical current flowing through the material, and observed as state space trajectories, i.e. the observable read out in Figure 4. Since the material is a part of the environment by virtue of being a physical object, there is inevitable interaction between the material and the environment. The genotype can still be treated as is common in AE; the entity operated upon by genetic operators such as mutation and crossover.

The relation between the genotype and phenotype is the genotype-phenotype map, This map takes a genotype as in-

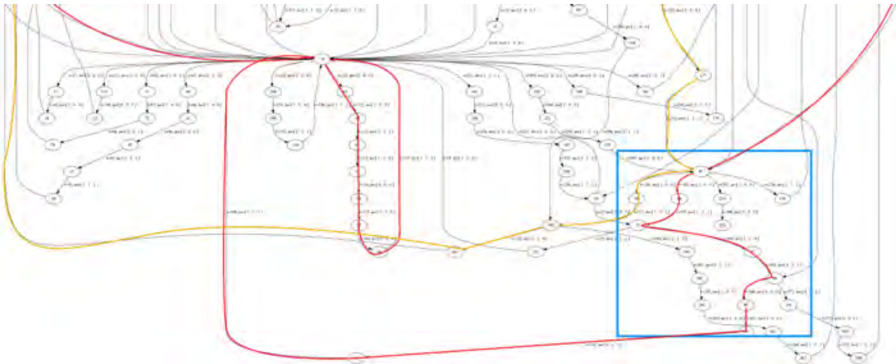

Figure 3: As an illustration of the experimental results a section of the full state space trajectory for an experimental run on salt/water solution with . A cut out in blue highlight state space trajectory changes (presented in Figure 5).

put and transforms it to a real-world entity with real-world physics and in particular includes the interface used to produce the desired manipulative phenomena, such as digital-to-analog converters. It is however worth noting that only what is observable to the EA (phenotypic behaviour or environmental effects of it's existance) can be used as input to a fitness function.

The experimental platform

The experimental results in this paper were achieved using the Mecobo platform (Lykkebo et al., 2014), a hardware/software implementation of an EIM system. The Mecobo platform is shown in Figure 2.

Figure 2(a) show the overall design. Configuration specification, i.e. genotypes, are loaded from a PC to Mecobo over the USB port. The micro controller communicates with the USB interface and with an FPGA on an internal bus. The FPGA can interface directly to materials or as in the figure use a daughter board to extend the signal range as shown in Figure2(b).

Mecobo is capable of controlling close to 100 individual configurable input/output signals (pins) that connects to the material. Each signal are described by parameters at a given point in time., e.g. recording pin from time 0, output frequency pin from time 0 to 10 or output pin voltage level 2.7V from time 0 etc., see (Lykkebo et al., 2014) for a detailed presentation of the Mecobo hardware and software.

The material samples are placed in a material bay, as seen in 2(c), pointed to with a red arrow, and connected to the Mecobo platform.

A standard genetic algorithm with tournament-based selection of size 3, a population size of 30 and a mutation probability of 0.2 was used. The mutation is drawn from a Gaussian distribution with $\sigma = 1$ and $\mu = 0$. The genome consists of 3 floating point numbers, from 0 to 1 which are scaled to integer *square wave* frequencies by 10^6 during the genotype-phenotype mapping process.

Each individual is run 5 times and stability (i.e. repeata-

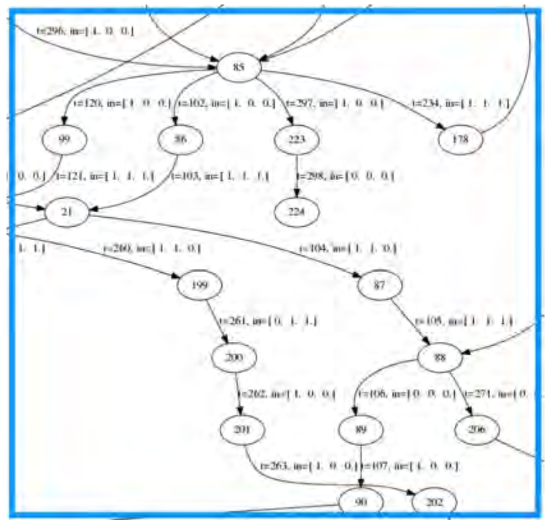

Figure 5: Zoomed region in blue in fig 3, showing the time-dependant behaviour of a driven evolved system in salt.

bility of the measured) states is part of the fitness. One run collects T state vectors $S(t)$ in a vector $B_r, r \in [0,4]$.

The fitness function counts the number of unique states measured during one time period T, where unique means that for all pairs $(t_1, t_2) \in [0, T]$ $S(t_1) \neq S(t_2)$ and divides the number of unique states by T. Finally, we take the cosine between all pairs of B_r-vectors and multiply the fitness by this number, ensuring that systems who has very similar trajectories in state space get awarded a high fitness.

The material bay has room for 60 connectors, of which we used 50 to connect to the Mecobo platform. 3 pins were then selected as designated input (*to the material*) pins; spread out in the material, one pin was selected as a current sink and the remaining 46 pins were used as material output pins, whose *digital* values were recorded over a time period of 200 uS at 20KHz.

The output pins are connected directly to the FPGA input buffers and further directly connected to FPGA-internal flip-flops with a triggering voltage of 1.7V. We define a *state* as a vector of size n $S(t) = (o_1(t), o_2(t), ..., o_n(t))$ where $o_i(t)$ is the value of flip flop i in the FPGA at time t.

We do not claim that the targeted system does useful computation. The purpose of this example is to demonstrate the existence of potentially complex behaviour in a driven physical system, which gives *potential* for useful computations. The chosen trajectory metric for the fitness evaluation is mainly chosen to be able to investigate for $(DS)^2$ behaviour. However the metric is in accordance with an abstract measurement of complexity as used by Langton (1991), Wolfram (1984) and for developmental systems (Kowaliw, 2008; Nichele et al., 2016).

Material Samples

The 'material cup', in fact a Multichannel Systems micro electrode array (MEA) model (60MEA100/10iR-Ti), pointed to in red in Figure 2(c), holds salt crystals formed by letting a solution 10mL of water with 1mL of kitchen table salt dry out, and before each evolutionary run these salt crystals are mixed with 50μL more of water to allow charge movement in the crystals. A second such MEA was filled with single walled carbon nano tubes (SWCNT) in a PMMA polymer solution. The same experiment is run on both. For control purposes, a fully conductive carbon plate was used.

Results and discussion

The Genetic Algorithm (GA) was used to provide data to be analyzed in a $(DS)^2$ setting. The resulting phenotypes from the evolutionary runs were analyzed by examining the state space traversal. In all results, $(DS)^2$ behavior was found. Figure 3 shows parts of a full state space traversal on dry salt crystals, visualized in graph form. Each node represents one state, and each arc between states one state transition. Figure 5 shows a zoomed version of the box marked by blue in figure 3. Figures 5 shows out-takes of the more interesting dynamic (time-dependent) behaviors in the run. Each edge is marked with the time-step (t) that the transition occurred on, along with the input as a tuple of 3 binary values (a,b,c). For a given input and a state one could expect that a transition should go to the same state, but as we can see for instance in state 85 of figure 5 this is not the case: at t=102, in=(1,0,0) there is a transition to state 86, whereas for t=120 the transition goes to a different state, 99. The red line in 3 traces out the full path of this branching behavior, turning orange at the second pass through state 85. This demonstrates a time-dependent behavior where the traversal of the state space depends on previously seen states.

The same method and set-up was used on the carbon nano tube(CNT) material sample. Time-dependent behavior whilst traversing the state space was present and found in all runs with the carbon nano tube material sample as well.

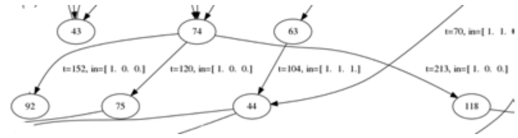

Figure 6: A state graph of a time-evolved carbon nano tubes driven state graph.

Figure 6 shows an out-take with dynamic (time-dependent) behaviors from the full state space traversal graph.

The GA typically achieves a fitness of 0.75 out of a 1.0 max within 50 generations, meaning that it finds a stable behavior that generates roughly $3/4$ unique states relative to the chosen observation level. This result is achieved in all cases tested. The number of transitions that show the time-dependent branching behavior discussed in relation to figure 6 is typically around 5. The chosen genome most often would map to frequencies around 100 KHz., which is higher than the Nyquist-rate relative to our sampling frequency of 50KHz, meaning that we cannot fully reconstruct the input signal from the sampling rate, however this simply underlines our previous points relating to where one sets the observation level– we are not concerned with signal reconstruction, but rather the systems ability to produce behavior in the $(DS)^2$ context.

This behavior is further shown in figure 7 for the salt crystals, and the plot is similar when using CNT as a material. The plot shows periodic behavior interspersed with spurious stable (in the sense that they meet our stability criteria previously defined), states. We stress that the nature of these states can have several reasons, (i.e. metastability in the flip-flops) and we cannot fully rule out the possibility of the sampling apparatus 'interfering' with the dynamics of i.e. the stimulus of the salt crystals, however runs with a fully conductive carbon plate as material gives us no such observed dynamics.

The vertical axis on this figure indicates the time as the voltage is applied to the salt crystals. On the left hand side we see the input to the system as it is captured by the same method that captures the rest of the state data– it is of a more regular nature (though some sampling artifacts).

The complexity of the input compared to the complexity of the output is currently under investigation, however for the purposes of this article the fact that the number of observable states (2^3) is much lower than the number of potential observable output states (2^{46}) demonstrates that there is potential for underlying $(DS)^2$ dynamics. Input is also verified as 'stable' in the same sense as output in that it is also based on a number of repeats of the same run and compared.

Conclusion

Analyzing the behavior of the evolved systems in a $(DS)^2$ setting show that both material samples explored show the property of inducing perturbations to their own dynamics as a function of their system states. In the state space traversal graph this behavior is visualized and show that the underlying dynamics of the materials enables changes in the trajectory in the state space. For the CNT sample the change in trajectory is a product of electrical effects that emerges as observable $(DS)^2$ traversal of the state space. For the salt crystal/water sample topological reconfigurations are possible but not detectable directly. The chosen observation level is here also only based on sampled voltages, a result of currents and charge in the material.

In the experiment *dynamics* was explicitly targeted, in contrast to previous EIM work on similar CNT materials ((Clegg et al., 2014; Mohid et al., 2014)), as the results show the hybrid approach with a digital read out dynamic behavior are present. The presence and possibility to observe such behavior can expand the computational range of a relative simple EIM-set-up from problems requiring only feed forward networks, e.g. (Clegg et al., 2014), to computational tasks requiring memory.

The $(DS)^2$ behavior at our chosen observation level show that it is not given that the system will behave predictably in the sense that a given input and given state will produce the same output– the current state might look the same, but actually be a result of a different underlying dynamic process. This does not imply that it is useless to use this system as a basis for building computational systems or similar, but rather that care must be taken when choosing and applying stimulus to the system and also when we evaluate the output by using a fitness function in a artificial evolution-approach. We suggest allowing the computational system to 'unfold' over time, treating the apparent weakness of unpredictability more as a strength. A way of doing this would be to not consider the system using reductionism and divide it's functionality into smaller parts (i.e. individual gates), but rather consider the system as a whole as a 'basic component' and it's dynamic properties a way of achieving computation.

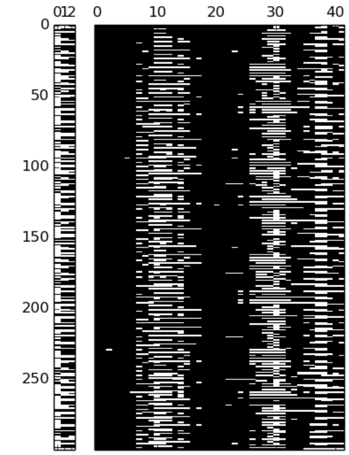

Figure 7: Salt crystals state plot

References

Adamatzky, A. (2009). *Encyclopedia of Complexity and Systems Science*, chapter Reaction-Diffusion Computing, pages 7548–7565. Springer New York, New York, NY.

Adamatzky, A. (2016). *Advances in Physarum Machines*. Springer International Publishing.

Arthur, W. B. (1993). On the Evolution of Complexity. Working Papers 93-11-070, Santa Fe Institute.

Bar-Yam, Y. (1997). *Dynamics of Complex Systems*. Studies in Nonlinearity. Westview Press.

Boses, S. K., Lawrence, C. P., Liu, Z., Makarenko, S., van Damme, R. M. J., Broersma, H. J., and van der Wiel, W. G. (2015). Evolution of a designless nanoparticle network into reconfigurable Boolean logic. *Nature Nanotechnology*, 10(12):1048–1052.

Bremermann, H. J. (1962). *Self-Organizing Systems-1962*, chapter Optimization through Evolution and Recombination, pages 93–106. Spartan Books.

Broersma, H., Gomez, F., Miller, J. F., Petty, M., and Tufte, G. (2012). Nascence project: Nanoscale engineering for novel computation using evolution. *International Journal of Unconventional Computing*, 8(4):313–317.

Clegg, K., Miller, J. F., Massey, M. K., and Petty, M. (2014). Travelling Salesman Problem solved in materio by evolved carbon nanotube device. In *Parallel Problem Solving from Nature – PPSN XIII: 13th International Conference*, Lecture Notes in Computer Science, pages 692–701. Springer.

Doursat, R., Sayama, H., and Michel, O. (2013). A review of morphogenetic engineering. *Natural Computing*, 12(4):517–535.

Fromm, J. (2004). *The Emergence of Complexity*. Kassel University Press.

Harding, S. L. and Miller, J. F. (2004). A Tone Discriminator in Liquid Crystal. In *Congress on Evolutionary Computation(CEC2004)*, pages 1800–1807. IEEE.

Holland, J. H. (2012). *Signals and Boundaries: Building Blocks for Complex Adaptive Systems*. MIT Press.

Kowaliw, T. (2008). Measures of complexity for artificial embryogeny. In *GECCO '08: Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 843–850. ACM.

Kumar, S. and Bentley, P. J., editors (2003). *On Growth, Form and Computers*. Elsevier Limited Oxford UK.

Langton, C. G. (1991). Computation at the edge of chaos: phase transitions and emergant computation. In Forrest, S., editor, *Emergent Computation*, pages 12–37. MIT Press.

Lykkebo, O. R., Harding, S., Tufte, G., and Miller, J. F. (2014). Mecobo: A hardware and software platform for in materio evolution. pages 267–279.

MacLennan, B. J. (2007). A Review of Analog Computing. Technical Report UT-CS-07-601. Technical report, University of Tennessee, Knoxville.

Massey, M., Volpati, D., Qaiser, F., Kotsialos, A., Pearson, C., Zeze, D., and Petty, M. (2015a). Alignment of liquid crystal/carbon nanotube dispersions for application in unconventional computing. *AIP Conference Proceedings*, 1648(1).

Massey, M. K. (2013). *Electrical Properties of Single-Walled Carbon Nanotube Networks Produced by Langmuir-Blodgett Deposition*. PhD thesis, Durham University, UK.

Massey, M. K., Kotsialos, A., Qaiser, F., Zeze, D. A., Pearson, C., Volpati, D., Bowen, L., and Petty, M. C. (2015b). Computing with carbon nanotubes: Optimization of threshold logic gates using disordered nanotube/polymer composites. *Journal of Applied Physics*, 117(13):134903.

Miller, J. F. and Downing, K. (2002). Evolution in materio: Looking Beyond the Silicon Box. In *2002 NASA/DOD Conference on Evolvable Hardware*, pages 167–176. IEEE Computer Society Press.

Miller, J. F., Harding, S., and Tufte, G. (2014). Evolution-in-materio: evolving computation in materials. *Evolutionary Intelligence*, 7(1):49–67.

Mohid, M., Miller, J. F., Harding, S. L., Tufte, G., Lykkebo, O. R., Massey, M. K., and Petty, M. C. (2014). Evolution-in-materio: Solving function optimization problems using materials. In *2014 14th UK Workshop on Computational Intelligence (UKCI)*, pages 1–8. IEEE.

Nichele, S., Giskeødegard, A., and Tufte, G. (2016). Evolutionary Growth of Genome Representations on Artificial Cellular Organisms with Indirect Encodings. *Artificial Life*, 22(1):76–111.

Nichele, S. and Tufte, G. (2013). Evolution of Incremental Complex Behavior on Cellular Machines. In *ECAL 2013*, pages 63–70. MIT Press.

Omholt, S. W. (2013). From sequence to consequence and back. *Progress in Biophysics and Molecular Biology*, 111(2-3):75–82.

Pask, G. (1959). Physical analougues to the growth of a concept. In *Mechanisation of Thought Processes*, number 10 in National Physical Laboratory Symposium, pages 877–922. Her Majesty's Stationery Office, London, UK.

Sayama, H. (2015). *Introduction to the Modeling and Analysis of Complex Systems*. Open SUNY Textbooks.

Shannon, C. E. (1941). Mathematical Theory of the Differential Analyzer. *Journal of Mathematics and Physics*, 20:337–354.

Simon, H. a. (1962). The architecture of complexity. *American Philosophical Society*, 106(6):467–482.

Spicher, A., Michel, O., and Giavitto, J.-L. (2004). *Cellular Automata: 6th International Conference on Cellular Automata for Research and Industry, ACRI 2004, Amsterdam, The Netherlands, October 25-28, 2004. Proceedings*, chapter A Topological Framework for the Specification and the Simulation of Discrete Dynamical Systems, pages 238–247. Springer Berlin Heidelberg, Berlin, Heidelberg.

Thompson, A., Layzell, P., and Zebulum, R. S. (1999). Explorations in Design Space: Unconventional electronics design through artificial evolution. *IEEE Transactions on Evolutionary Computation*, 3(3):167–196.

Tufte, G. (2009). The Discrete Dynamics of Developmental Systems. In *Proc. of 2009 International Conference on Evolutionary Computation (CEC 2009)*, pages 2209–2216. IEEE.

Turing, A. M. (1937). On Computable Numbers, with an Application to the Entscheidungsproblem. In *Proceedings of the London Mathematical Society 1936-37*, volume 42 of *2*, pages 230–265. London Mathematical Society.

von Neumann, J. (1993). First draft of a report on the edvac. *IEEE Annals of the History of Computing*, 15(4):27–75.

Wolfram, S. (1984). Universality and Complexity in Cellular Automata. *Physica D*, 10(1-2):1–35.