Time Series Evolution for Integrating Developmental ~~Burglary~~ Processes

Sebastian von Mammen and Melanie Däschinger

Organic Computing, University of Augsburg, Germany
s.vonmammen@gmail.com
MelanieD@hotmail.de

Abstract

Material and virtual entities alike undergo developmental processes. If they replicate, they evolve and adapt to a variety of developmental processes. Based on the notion that these processes rarely constitute a direct translation from one state to the next, we have taken a step towards formally capturing evolutionary development of time series. We iteratively extend the Knapsack Problem to consider time, personal preferences, contradicting goals, external events, and changing environments and we show how the evolution of time series can be driven to address these various challenges. The presented work constitutes a small step towards a more rigorous approach hinted at towards the end of the paper.

Introduction

When facing evolutionary challenges, organisms need to adapt—behaviourally within a lifetime, see for instance Noë and Laporte (2014), or genetically over the course of generations, as comprehensively investigated in the context of the evolution of the eye (Lamb (2011)). We consider adaptation in terms of properties of the organism, in terms of its anatomy or physiology. Yet, properties, just like environmental challenges, emerge one after the other and evolve over time (Gilbert and Burian (2003)). Accordingly, there is no fixed phenotype of an organism but a series of phenotypical states over time. This notion is supported by the non-linear order of expression of genotypical information (Dang (2014)). In order to take a step towards this notion of organismal development, we have subjected a time series representation to evolutionary processes. Hereby, the transition to the next point in time of the organism's state depends on the previous one. At the same time, we assign such time-based individuals fitness values integrated over time, also considering changing environmental conditions.

In the remainder of this paper, we briefly touch upon preceding approaches of evolutionary development, first. Next, we present our concept of time series evolution (TSE) in the context of the knapsack problem. We extend the problem to dynamic processes in terms of filling the backpack and in terms of multivariate external challenges. Step by step, we expand the representation of an individual and its fitness evaluation to arrive at a generic approach to time series evolution in dynamic environments. Experiments are presented to back up and to illustrate our rationale. We conclude our presentation with a summary and an outlook at ongoing and potential future work.

Related Work

There have been numerous computational approaches to condense the principles of growth and evolution. Stanley (2014) provide a valuable overview to this field. Abstract data structures have been automatically, interactively, immersively bred to take on a multiplicity of challenges (Sayama (2014),Von Mammen and Jacob (2009)). For instance, the morphology of soft robots and their behaviour have been subjected to evolutionary algorithms (e.g. Rieffel et al. (2014)), establishing accurate models of plant growth (e.g. Henke et al. (2014)), and the design of computational hardware has been supported by evolutionary approaches (e.g. Bhattacharjee et al. (2015)). Typically, such computational evolutionary developmental (or EvoDevo) approaches focus on a desirable end product. Yet, the product itself will be embedded in a context, it will be used, be worn, and vanish at some point in time. Similarly, its production does not entail a direct mapping from a concept to an artefact. Instead, it grows one step after the next—its existence changes over its lifetime (Gilbert and Burian (2003)). Hence, development is a proactive state of being rather than a phase with a well-defined beginning and end. Therefore, the field of adaptive computing approaches, including autonomic computing, which is tailored towards the continuous adaptation of computing systems (Wódczak (2014)) represents an important reference to the work presented in this paper. Even more so, do organic computing (Bernard et al. (2014)) and morphogenetic engineering (Kowaliw et al. (2014)) which target rather generic adaptive systems. However, the focus of this work still varies—it investigates the interplay of time series as generic developmental representations and genetic algorithms to allow their generational adaptation.

The 0-1-Knapsack Problem and TSE

The concept of time series evolution is motivated by the modelling challenges of developmental processes (Gilbert and Burian (2003)). Therefore, we identified the Knapsack Problem (KP) to be a theoretical problem which (a) considers development, which can (b) be expanded towards multi-objectivity, and (c) allows the introduction of the aspect of time.

The 0-1-Knapsack Problem can be defined as follows (Plateau and Nagih (2010)). n items of weights $w_1, ..., w_n$ and of values $v_1, ..., v_n$ are given, alongside a backpack with a weight limit of L. Valid solutions to solving this problem are vectors $a_1, ..., a_n$, with $a_i \in \{0, 1\}$ denoting whether or not the corresponding item is put inside the backpack and the constraint that the summed weight of all items does not exceed the limit, i.e. $\sum_{i=1}^{n} a_i w_i \leq L$. The best solution to the problem is the combination of items with the greatest overall value.

From 0-1-KP to Development

Now consider a burglar carrying the backpack during his raid. Thereby, the Knapsack Problem is modified from a static combinatorial planning challenge to a development process. It maintains the original goal of maximising the backpack's contained value but it allows for consideration of additional constraints. These can, for instance, be the burglar's varying physical condition, lock-picking challenges and other external factors.

We define an according *developmental series* as a vector $\vec{d} = (y_1, ..., y_n)^T$ which describes a number of decisions (encoded as numeric values) taken at the corresponding points in time $t \in \{1, ..., n\}$. Such developmental series represent solutions, or individuals in the context of evolutionary optimisation, given their sequences are valid. A function $g(t)$ may serve as generator of the developmental series, and for integrating the solution's state at an arbitrary point in time.

Heuristic 0-1-KP Solving

Before adding any further constraints, we briefly show exemplary how a developmental series can approximate the optimal solution to the 0-1-Knapsack Problem. In particular, we consider the collectible items' value/mass-ratios and decide which one to pick up and which one to leave behind. In order to calculate a series' integrated value at a specific point in time, $g(t)$ would therefore be defined as follows (Eqn. 1). Figure 1 shows an evolved solution for a limit $L = 57$ for picking up a subset of 20 items that occur in descending order of their value/weight-ratios. Table 1 shows the given items.

$$g(t) = \begin{cases} g(t-1) + \frac{v_t}{w_t} & , if a_t = 1 \\ g(t-1) & , if a_t = 0 \end{cases} \qquad (1)$$

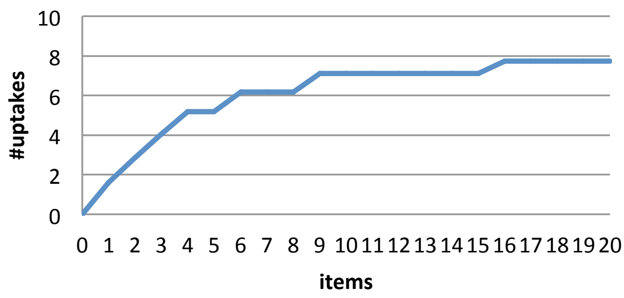

Figure 1: Uptake of items ordered according to their value/mass-ratios.

i	*v*	*w*	v/w	*i*	*v*	*w*	v/w
1	8	5	1,6	11	10	12	0,83
2	5	4	1,25	12	9	12	0,75
3	12	10	1,2	13	5	7	0,714
4	17	15	1,13	14	21	30	0,7
5	15	14	1,071	15	2	3	0,667
6	3	3	1,0	16	5	8	0,625
7	6	6	1,0	17	6	10	0,6
8	10	10	1,0	18	4	8	0,5
9	11	12	0,917	19	2	5	0,4
10	6	7	0,857	20	2	9	0,22

Table 1: Exemplary set of items for the 0-1-Knapsack Problem with indices i, values v and weights w.

Introducing Evolution

The generational evolution of developmental series can be realised by a Genetic Algorithm (Holland and Reitman (1977)). We need to assume that certain successions of states describe more successful developmental processes than others. Therefore, without loss of generality, we provide fitness values for each state of a developmental process, which results in a *fitness series* $\vec{f} = (z_1, ..., z_n)^T$ that defines the optimum. We rely on the mean quadratic error $\frac{1}{n} \sum^{n}{}_{i=1}(y_i - z_i)^2$ to derive a single fitness value for each developmental series. With an according set of genetic operators, the binary representation of the developmental series can be extended to arbitrary numeric values.

For the experiments presented throughout this paper, we deployed mutations with a chance of 10%. 30% of new offspring emerged from recombination of preceding specimen. We relied on fitness proportionate selection and $1-$elitism to keep the single best solution in the pool. Our tests regarding the population size included $20, 25, 40,$ and 50 individuals. Eventually, we stuck with 32, as it provided the best results. We let our experiments run for 40 generations, whereas the outcome typically converged after 10 generations.

Multi-objectivity

To give credit to the fact that a single optimisation criterion rarely captures the many facets of developmental processes, we extend an individual to a set of developmental series. Relating to the example above, the burglar might, for example, want to gather great value and achieve financial security fast. He might also want to maintain low weight of the backpack, especially for the first hours of the raid. In summary, we consider a population of individuals which consist of sets of developmental series DS. The elements $\vec{d_i} \in DS$ describe the development of some attribute i over n points in time. A function $g_i(t)$ can generate the respective values. As a result of the multi-series extension, the overall fitness of an individual can be calculated as the total of fitness values of all its development series over a given period of time. In analogy to $g_i(t)$, the fitness target values can be provided by functions $f_i(t)$. This generic representation not only allows for arbitrary fitness evolutions but may also serve for balancing the relative weights of the considered attributes.

Synthesis of Fitness Series

As can be expected, we observed that optimisation towards individual criteria, such as stalling weight growth for as long as possible, converges well. Loading up items from Table 1 with indices greater than 13, would, for instance, result in a mass development series $\vec{d}_{mass} = (0, ..., 30, 33, 33, 43, 43, 48, 57)^T$. Given the function in Equation 2, which rewards a late mass increase, yields a relatively high fitness value of $222,75$.

$$f_{mass}(t) = 0,012 \cdot t^2 \cdot |\vec{d}_{mass}(t) - \vec{d}_{mass}(t-1)| \quad (2)$$

Adding the early financial security criterion f_{value} (Eqn. 3 and 4), which qualitatively inverts function f_{mass}' reward policy, would effectively work against this development. As a consequence, the given individual would perform very poorly in terms of the total value of $2,1$ of considered fitnesses. Accordingly, the influence of several fitness series needs to be automatically balanced to reach a global optimum.

$$\Delta\vec{d}_{value} = |\vec{d}_{value}(t) - \vec{d}_{value}(t-1)| \quad (3)$$

$$f_{value}(t) = \begin{cases} (4 - 0.07 \cdot (t-1)^2) \cdot \Delta\vec{d}_{value}, if\ t < 9 \\ (0.05 \cdot \Delta\vec{d}_{value}, if\ t < 9 \end{cases} \quad (4)$$

Multiple Criteria & Diversity

In the following, we present results from applying multiple fitness criteria, including a fast increase in value (Eqn. 4), the maximisation of the value/weight ratio, and the slow accumulation of weight (Eqn. 2). We further considered the total weight and total value, and some inherited knowledge $f_{experience}$ that works as a generational memory for good decisions but is independent of any items' properties.

An individual optimised to address the 0-1-KP in accordance with Bellman's optimality equation (Montrucchio (1986)) would simply pick up the first seven items in Table 1. Yet, considering the given number of fitness series, such an optimised specimen would only achieve a final value of 66 and receive an overall fitness of $339,51$ ($v/m = 82,55, v = 212,41, m = 16,55, experience = 14,0, other = 14,0$). Instead, the best GA-bred individual to address all the given factors achieved an overall fitness value of $380,95$. Figure 2 shows its genotype and the relative fitnesses.

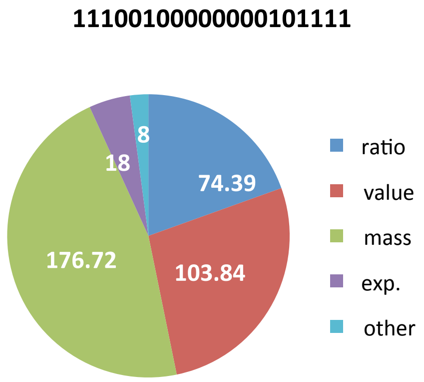

Figure 2: Top: The genotype of the best evolved specimen. Bottom: Its relative phenotypic fitness values.

We also found multiple specimen that achieved similarly high overall fitness ratings of about 300, relying on fundamentally different pick-up strategies. Figure 3 shows two rather diverse examples. Individual A achieves high overall fitness picking up items early on, whereas individual B focusses on the relative maximisation of value.

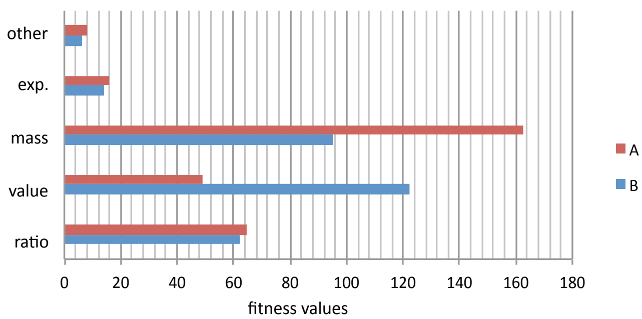

Figure 3: In this diagram, two individuals A and B are compared in terms of their relative fitness scores. Both individuals have achieved rather high overall scores of about 300.

Dynamic Environments

So far, our model considers several developmental time series but only one set of ordered items. Although helpful for model development itself, the latter restriction is not adequate when considering complex developmental processes. Metaphorically speaking, the burglar may not be able to choose his raiding route upfront, not know what items to expect along the way in detail, nor would it be possible to plan in unforeseeable events, e.g. the appearance of a police patrol.

Based on these deliberations, one goal could be to find strategies that optimally fit a broad range of item spaces (and orderings). Successful, fixed genotypes might reveal significant overlaps, in terms of picking up specific items, item properties or qualitative pick up sequences, that indicate preferable behaviours. When changing the order of the items, we discovered that the three items—originally indices 11, 13, and 19 from Table 1—were picked by the winners of two consecutively performed breeding experiments.

Dynamic Internal States

Another important modelling aspect is the condition of the burglar, or its internal state. Burglary, like everything else, requires energy. Therefore, without loss of generality, we assumed three breaks for snacks throughout the raid, with an overall decrease of recovery, see the blue columns in Figure 4. Their energetic maxima (at times $t = 2, 9, 17$) coincide with a tendency to pick up items. The minima (at times $t = 7, 14, 15, 20$), on the other hand, indicate exhaustion which nullifies the ability to pick up items. Energy is generally a rewarding dimension when considering process optimisation. For instance, it could be used to handle the aforementioned encounter with a police patrol—decreasing the specimen's supply at the time of the event (indicated by the red columns in Figure 4). Such strong constraints, of course, need to be considered during the evolutionary runs in order to provide valid solutions. The representation of external factors would encompass one n-dimensional vector $\vec{a}$ that quantifies the impact, a reference to the targeted developmental series $\vec{ts}$, and a set of constraints C that describe the relative impact of $\vec{a}$ on $\vec{ts}$

Dynamic System^2

Combining the concepts of dynamic system states and dynamically changing environments, we extend our model to consider DS^2, dynamic systems with a dynamical structure (Michel et al. (2009)): Breaks cannot be accurately anticipated during a raid, police officers do not patrol neighbourhoods at regular times. Accordingly, in an experimental run comprising 25 generations, we offset the occurrence of police by 0 to 3 units and the occasion for breaks by 1 to 2 units at each other generation. The results can be seen in Table 2. As expected, the population adapts to the interval shifts. We recognise a corresponding cyclic pattern.

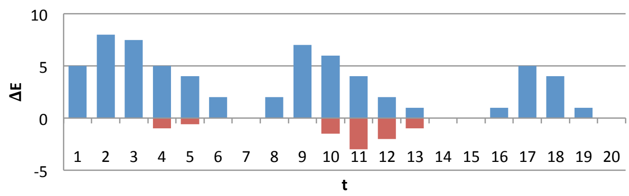

Figure 4: We define the internal state of the developing organism to depend on external events. In our example, a burglar replenishes his energy level three times throughout the raid (in blue). Encounters with a police patrol may cost energy (in red).

gen.	f	v	m	f_{max}-genotype	e/p
0-2	316	52	56	11101000000000001110	-
3-4	279	53	57	01110100010000001100	1/1
5-6	288	51	56	00111100000000000011	1/1
7-8	290	53	55	11101010000010000001	2/2
9-10	230	59	57	01101111010000100000	2/3
11-12	211	54	57	00101100110000110000	1/1
13-14	213	45	57	00001100001100100110	2/2
15-16	212	39	57	00000110001110100011	1/0
17-18	233	48	56	11000000110010111000	2/1
19-20	260	46	57	11000000011000111100	1/1
21-22	252	43	55	01100000000100111100	1/1
23-24	289	36	57	00100000000010011111	1/1
25-26	268	37	56	00001100000001000111	1/2

Table 2: Every other generation (gen.), a timing offset was introduced regarding energy intake (e) and policing (p) events. The best fitness values f_{max} indicate the adaptation of the population, the shifting pattern in item uptakes reflects the change of external events.

Summary & Future Work

In this paper, we have presented the concept of evolving time series in the context of developmental processes. Step by step, we extended the Knapsack Problem, turning it into a metaphorically understood burglar raid to suit the challenges faced in actual developmental processes. We started by merely evolving *developmental series*—in the given example a decision strategy for stealing specific items at particular points in time. Predetermined ideal progression was provided by *fitness series*, evolution of the developmental series ensured their approximation.

Next, partially contradicting desires by the decision maker were put to the test, necessitating prioritisation. Finally, we started considering the internal (physical) state of the burglar, introducing the notion of a generic time-dependent fitness criterion (energy) that is tightly interwoven with the environment. Our last experiments showed how time series evolution successfully adapts to dynamic environmental challenges.

Although it is already partially incorporated in the presented model, a major challenge is the accessible description and efficiently resolvable computation of constraints (a) among different developmental series and (b) across time steps. Depending on the resulting overhead, we hope to deploy our approach to biological developmental modelling and prediction. Currently, we are working on the architectural deployment of time series evolution as we can easily choose a manageable level of abstraction and since we expect actually applicable results.

References

Bernard, Y., Klejnowski, L., Bluhm, D., Hähner, J., and Müller-Schloer, C. (2014). Self-organisation and evolution for trust-adaptive grid computing agents. In Evolution, Complexity and Artificial Life, pages 209–224. Springer.

Bhattacharjee, D., Banerjee, A., and Chattopadhyay, A. (2015). Evodeb: Debugging evolving hardware designs. In VLSI Design (VLSID), 2015 28th International Conference on, pages 481–486. IEEE.

Dang, C. V. (2014). Gene regulation: Fine-tuned amplification in cells. Nature, 511(7510):417–418.

Gilbert, S. F. and Burian, R. M. (2003). Development, evolution, and evolutionary developmental biology. Keywords and concepts in evolutionary developmental biology, pages 61–8.

Henke, M., Huckemann, S., Kurth, W., and Sloboda, B. (2014). Reconstructing leaf growth based on non-destructive digitizing and low-parametric shape evolution for plant modelling over a growth cycle. SILVA FENNICA, 48(2).

Holland, J. H. and Reitman, J. S. (1977). Cognitive systems based on adaptive algorithms. ACM SIGART Bulletin, (63):49–49.

Kowaliw, T., Bredeche, N., Chevallier, S., and Doursat, R. (2014). Artificial neurogenesis: An introduction and selective review. In Growing Adaptive Machines, pages 1–60. Springer.

Lamb, T. D. (2011). Evolution of the eye. Scientific American, 305(1):64–69.

Michel, O., Spicher, A., and Giavitto, J.-L. (2009). Rule-based programming for integrative biological modeling. Natural Computing, 8(4):865–889.

Montrucchio, L. (1986). Optimal decisions over time and strange attractors: An analysis by the bellman principle. Mathematical Modelling, 7(2):341–352.

Noë, R. and Laporte, M. (2014). Socio-spatial cognition in vervet monkeys. Animal cognition, 17(3):597–607.

Plateau, G. and Nagih, A. (2010). 0–1 knapsack problems. Paradigms of Combinatorial Optimization, 2nd Edition, pages 215–242.

Rieffel, J., Knox, D., Smith, S., and Trimmer, B. (2014). Growing and evolving soft robots. Artificial life, 20(1):143–162.

Sayama, H. (2014). Guiding designs of self-organizing swarms: Interactive and automated approaches. In Guided Self-Organization: Inception, pages 365–387. Springer.

Stanley, K. O. (2014). Generative and developmental systems tutorial. In Proceedings of the 2014 conference companion on Genetic and evolutionary computation companion, pages 765–794. ACM.

Von Mammen, S. and Jacob, C. (2009). The evolution of swarm grammars-growing trees, crafting art, and bottom-up design. Computational Intelligence Magazine, IEEE, 4(3):10–19.

Wódczak, M. (2014). Autonomic computing and networking. In Autonomic Computing Enabled Cooperative Networked Design, pages 3–16. Springer.