Time Series Evolution for Integrating Developmental Burglary Processes

Sebastian von Mammen and Melanie Däschinger

Organic Computing, University of Augsburg, Germany
s.vonmammen@t-online.de
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. Within the interplay of genes and environment individuals shall develop who can survive optimally. 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—behaviorally within a lifetime, see for instance Noé 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 first reflect upon preceding approaches of evolutionary development. Afterwards, we present our concept of time series evolution (TSE) in the context of the knapsack problem. We extend the problem to a dynamic burglary process 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. Finally, this approach is supposed to be used in complex systems, which consist of numerous interrelated processes, in order to find individuals who personate good and robust solutions. 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 behavior 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 multiobjectivity, and (c) allows the introduction of the aspect of time.

The 0-1-Knapsack Problem can be defined as follows (Plateau and Nagih, 2010). Let n items with weights $w_1, ..., w_n$ and of values $v_1, ..., v_n$, be 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 $d^t = (y_1, ..., y_n)^T$, 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 describe an example to show 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.

\[
g(t) = \begin{cases} \frac{v_i}{w_i}, & \text{if } a_t = 1 \\ g(t-1), & \text{if } a_t = 0 \end{cases}
\]

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

<table>
<thead>
<tr>
<th>i</th>
<th>v</th>
<th>w</th>
<th>v/w</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>5</td>
<td>1.6</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>4</td>
<td>1.25</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>10</td>
<td>1.2</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
<td>15</td>
<td>1.13</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>14</td>
<td>1.071</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>11</td>
<td>12</td>
<td>0.917</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>7</td>
<td>0.857</td>
</tr>
</tbody>
</table>

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 a developmental process, which results in a fitness series $f = (z_1, ..., z_n)^T$ that defines the optimum. We rely on the mean quadratic error $\frac{1}{n} \sum_{i=1}^{n} (y_i - z_i)^2$ 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 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 $d_i \in DS$ 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 $f_j(t)$. 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 $\vec{d}_{\text{mass}} = (0, ..., 30, 33, 34, 43, 48, 57)^T$. Given the function in Equation 2, which rewards a late mass increase, yields a relatively high fitness value of 222.75.

$$f_{\text{mass}}(t) = 0.012 \cdot t^2 \cdot |\vec{d}_{\text{mass}}(t) - \vec{d}_{\text{mass}}(t - 1)|$$  \hspace{1cm} (2)

Adding the early financial security criterion $f_{\text{value}}$ (Eqn. 3 and 4), which qualitatively inverts function $f_{\text{mass}}$, 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.

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

$$f_{\text{value}}(t) = \begin{cases} (4 - 0.07 \cdot (t - 1)^2) \cdot \Delta \vec{d}_{\text{value}}, & \text{if } t < 9 \\ 0.05 \cdot \Delta \vec{d}_{\text{value}}, & \text{if } t \geq 9 \end{cases}$$  \hspace{1cm} (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 $f_{\text{experience}}$ 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 ($v/m = 82.55, v = 212.41, m = 16.55, \text{experience} = 14.0, \text{other} = 14.0$). 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.](image)

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.

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...
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.

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).

Table 2: Every other generation (gen.), a timing offset was introduced regarding energy intake ($\epsilon$) 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. Fi-
nally, 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


