Towards performance-focused implementations of adaptive devices

Paulo Roberto Massa Cereda\textsuperscript{a,}\textsuperscript{*}, João José Neto\textsuperscript{a}

\textsuperscript{a}Escola Politécnica, Departamento de Engenharia de Computação e Sistemas Digitais, Universidade de São Paulo
Av. Prof. Luciano Gualberto, s/n, Travessa 3, 158, CEP: 05508-900 – São Paulo, SP – Brasil

Abstract
This paper presents instrumentation metrics for adaptive rule-driven devices as a means to obtain performance-focused implementations, from the underlying non-adaptive rule-driven device to the adaptive mechanism, as well as discussions regarding the adaptive behaviour and its corresponding operations, from theoretical and practical points of view.

1. Introduction
Adaptivity is the term used to denote a phenomenon in which a device spontaneously changes its internal behaviour in order to accommodate planned yet unexpected situations; these changes are triggered based solely on the device’s own rule set and input stimuli, without any external interference\textsuperscript{1,2}. A device is called adaptive if such feature is available to the model as a whole.

Although offering no computational power boost, adaptivity provides mechanisms for expressing abstractions more conveniently\textsuperscript{1,3,4}. As a direct consequence, several model improvements are made possible and practically viable, such as complexity reduction\textsuperscript{5}, problem partitioning\textsuperscript{6} and hierarchical solving\textsuperscript{7}, available at almost no sensible cost to the user\textsuperscript{3}.

Implementations of adaptive devices greatly vary, as well as the models themselves\textsuperscript{3,8,9}. An early work of Cereda and José Neto\textsuperscript{3} discussed potential bottlenecks and shortcomings on using common software engineering techniques as a means to implementing programs with adaptive characteristics, with drastic – and sometimes fatal – impacts on performance and stability. In this paper, we aim at extending the discussion to the adaptive mechanism, through fine-grained instrumentation of the adaptive behaviour and its corresponding operations, from theoretical and practical points of view.

Instrumentation is the capability of monitoring and recording a device behaviour and measuring performance during its life cycle\textsuperscript{10}. It plays a crucial role in evaluation and testing procedures, as the collected data provide basis for achieving better performance and model improvements\textsuperscript{11,12,13}. It is generally advisable to combine different metrics

\begin{footnotesize}
\textsuperscript{*} Corresponding author, +55 11 3091-5402.
\textit{E-mail addresses:} paulo.cereda@usp.br (Paulo Roberto Massa Cereda), jjneto@usp.br (João José Neto).
\end{footnotesize}
in order to obtain a more comprehensive representation of such collected data, in an attempt to reduce bias. However, producing traces incurs runtime overhead and therefore may interfere with the device’s timing and perturb its behaviour, so instrumentation has to be kept to a minimum. As to trace adaptive devices, we exposed their inner workings to analysis and gathered relevant data on queries and operations.

This paper is organized as follows: Section 2 formally introduces the concept of an adaptive rule-driven device. Section 3 presents presents the instrumentation metrics, as well as implementation aspects. Experiments and discussions are presented in Section 4. Finally, conclusions are presented in Section 5.

2. Adaptive rule-driven devices

This section formally introduces the concept of a general adaptive rule-driven device. It is important to observe that any non-adaptive rule-driven device may be enhanced in order to accommodate an adaptive behaviour while preserving its integrity and original properties. The adaptive mechanism acts as simple extension to the underlying non-adaptive device.

Definition 1 (rule-driven device). A rule-driven device is defined as \( ND = (\mathbf{C}, \mathbf{R}, \mathbf{S}, \mathbf{c}_0, \mathbf{A}, \mathbf{N}) \), such that \( ND \) is a rule-driven device, \( \mathbf{C} \) is the set of all possible configurations, \( \mathbf{c}_0 \in \mathbf{C} \) is the initial configuration, \( \mathbf{S} \) is the set of all possible input stimuli, \( \mathbf{e} \in \mathbf{S} \), \( \mathbf{A} \subseteq \mathbf{C} \) is the subset of all accepting configurations (respectively, \( \mathbf{F} = \mathbf{C} - \mathbf{A} \) is the subset of all rejecting configurations), \( \mathbf{N} \) is the set of all possible output stimuli of \( ND \) as a side effect of rule applications, \( \mathbf{e} \in \mathbf{N} \), and \( \mathbf{R} \) is the set of rules defining \( ND \) as a relation \( \mathbf{R} \subseteq \mathbf{C} \times \mathbf{S} \times \mathbf{C} \times \mathbf{N} \).

Definition 2 (rule). A rule \( \mathbf{r} \in \mathbf{R} \) is defined as \( \mathbf{r} = (\mathbf{c}_i, \mathbf{s}, \mathbf{c}_j, \mathbf{z}) \), \( \mathbf{c}_i, \mathbf{c}_j \in \mathbf{C} \), \( \mathbf{s} \in \mathbf{S} \) and \( \mathbf{z} \in \mathbf{N} \), indicating that, as response to a stimuli \( \mathbf{s} \), \( \mathbf{r} \) changes the current configuration \( \mathbf{c}_i \) to \( \mathbf{c}_j \), processes \( \mathbf{s} \) and generates \( \mathbf{z} \) as output. A rule \( \mathbf{r} = (\mathbf{c}_i, \mathbf{s}, \mathbf{c}_j, \mathbf{z}) \) is said to be compatible with the current configuration \( \mathbf{c} \) if and only if \( \mathbf{c}_i = \mathbf{c} \) and \( \mathbf{c} \) is either empty or equals the current input stimuli; in this case, the application of a rule \( \mathbf{r} \) moves the device to a configuration \( \mathbf{c}_j \), denoted by \( \mathbf{c}_i \Rightarrow \mathbf{c}_j \), and adds \( \mathbf{z} \) to the output stream.

Definition 3 (acceptance of an input stimuli stream by a rule-driven device). An input stimuli stream \( \mathbf{w} = w_1 w_2 \ldots w_n \), \( w_k \in \mathbf{S} - \{\mathbf{e}\}, k = 1, \ldots, n, n \geq 0 \), is accepted by a device \( ND \) when \( \mathbf{c}_0 \Rightarrow^{w_1} \mathbf{c}_1 \Rightarrow^{w_2} \ldots \Rightarrow^{w_n} \mathbf{c} \) (in short, \( \mathbf{c}_0 \Rightarrow^\mathbf{w} \mathbf{c} \)), and \( \mathbf{c} \in \mathbf{A} \). Respectively, \( w \) is rejected by \( ND \) when \( \mathbf{c} \in \mathbf{F} \). The language described by a rule-driven device \( ND \) is represented by \( L(ND) = \{w \in \mathbf{S}^* \mid \mathbf{c}_0 \Rightarrow^w \mathbf{c}, \mathbf{c} \in \mathbf{A}\} \).

Definition 4 (adaptive rule-driven device). A rule-driven device \( \mathbf{A} = (\mathbf{N}_0, \mathbf{AM}) \), such that \( \mathbf{N}_0 \) is a device and \( \mathbf{AM} \) is an adaptive mechanism, is said to be adaptive when, for all operation steps \( k \geq 0 \) (\( k \) is the value of an internal counter \( T \) starting in zero and incremented by one each time a non-null adaptive action is executed), \( \mathbf{A} \) follows the behaviour of an underlying device \( \mathbf{N}_0 \) until the start of an operation step \( k+1 \) triggered by a non-null adaptive action, modifying the current rule set; in short, the execution of a non-null adaptive action in an operation step \( k \geq 0 \) makes the adaptive device \( \mathbf{A} \) evolve from an underlying device \( \mathbf{N}_0 \) to \( \mathbf{N}_0^{k+1} \).

Definition 5 (operation of an adaptive device). An adaptive device \( \mathbf{A} \) starts its operation in configuration \( \mathbf{c}_0 \), with the initial format defined as \( \mathbf{A}_0 = (\mathbf{C}_0, \mathbf{R}_0, \mathbf{S}, \mathbf{c}_0, \mathbf{A}, \mathbf{N}, \mathbf{B}, \mathbf{A}) \). In step \( k \), an input stimuli move \( \mathbf{A} \) to the next configuration and starts the operation step \( k + 1 \) if and only if a non-adaptive rule is executed; thus, being the device \( \mathbf{A} \) in step \( k \), with \( \mathbf{A}_k = (\mathbf{C}_k, \mathbf{R}_k, \mathbf{S}, \mathbf{c}_k, \mathbf{A}, \mathbf{N}, \mathbf{B}, \mathbf{A}) \), the execution of a non-null adaptive action leads to \( \mathbf{A}_{k+1} = (\mathbf{C}_{k+1}, \mathbf{R}_{k+1}, \mathbf{S}, \mathbf{c}_{k+1}, \mathbf{A}, \mathbf{N}, \mathbf{B}, \mathbf{A}) \), in which \( \mathbf{A} = (\mathbf{N}_0, \mathbf{AM}) \) is an adaptive device with a starting underlying device \( \mathbf{N}_0 \) and an adaptive mechanism \( \mathbf{AM} \). \( \mathbf{N}_0 \) is an underlying device of \( \mathbf{A} \) in an operation step \( k \), \( \mathbf{N}_k \) is the set of non-adaptive rules of \( \mathbf{N}_0 \), \( \mathbf{C}_k \) is the set of all possible configurations for \( \mathbf{N}_0 \) in an operation step \( k \), \( \mathbf{C} \) is the subset of accepting configurations (respectively, \( \mathbf{F} = \mathbf{C} - \mathbf{A} \) is the subset of rejecting configurations), \( \mathbf{B} \) and \( \mathbf{A} \) are sets of adaptive actions (both containing the null action, \( \mathbf{e} \in \mathbf{B} \cap \mathbf{A} \)), \( \mathbf{N} \), with \( \mathbf{e} \in \mathbf{N} \), is the set of all output stimuli of \( \mathbf{A} \) as side effect of rule applications, \( \mathbf{R}_k \) is the set of adaptive rules defined as a relation \( \mathbf{A}_k \subseteq \mathbf{B} \times \mathbf{C} \times \mathbf{S} \times \mathbf{C} \times \mathbf{N} \times \mathbf{A} \), with \( \mathbf{R}_0 \) defining the starting behaviour of \( \mathbf{A} \). \( \mathbf{A} \) is the set of all possible adaptive rules for \( \mathbf{A} \), \( \mathbf{N}_k \) is the set of all possible underlying non-adaptive rules of \( \mathbf{A} \), and \( \mathbf{A} \) is an adaptive mechanism, \( \mathbf{A} \subseteq \mathbf{B} \times \mathbf{N} \times \mathbf{A} \), to be applied in an operation step \( k \) for each rule in \( \mathbf{N}_k \).
دریافت فوری
متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات