Chapter 3.19
Intelligent Business Process Execution using Particle Swarm Optimization

Markus Kress
University of Karlsruhe, Germany

Sanaz Mostaghim
University of Karlsruhe, Germany

Detlef Seese
University of Karlsruhe, Germany

ABSTRACT

In this chapter, the authors study a new variant of Particle Swarm Optimization (PSO) to efficiently execute business processes. The main challenge of this application for the PSO is that the function evaluations typically take a high computation time. They propose the Gap Search (GS) method in combination with the PSO to perform a better exploration in the search space and study its influence on the results of our application. They replace the random initialization of the solutions for the initial population as well as for the diversity preservation method with the GS method. The experimental results show that the GS method significantly improves the quality of the solutions and obtains better results for the application as compared to the results of a standard PSO and Genetic Algorithms. Moreover, the combination of the methods the authors used show promising results as tools to be applied for improvement of Business Process Optimization.

DOI: 10.4018/978-1-60566-705-8.ch003

INTRODUCTION

The success of a business enterprise highly depends on its business processes as they are the basis of the economic collaboration within an enterprise and between enterprises. Many companies are facing a large variety of issues such as faster time-to-market, shorter product life cycles, and increasing competition due to the globalization. One of the methods to take up these challenges is to optimize the running business processes, among others.

Business process management (BPM) methods must provide the required flexibility. In order to make use of standard BPM tools for the execution of business processes, the process models must be completely defined in the sense that every aspect and all the exceptions must be known and modelled in advance.

The so called Executable Product Model (EPM) approach proposed by Kress et al. (2007) combines a compact model with an intelligent control flow mechanism. The approach belongs to the product model based approaches such as those studied by Küster et al. (2007), Müller et al. (2007), and van der Aalst et al. (2001). The EPM is based on the product data model, which is extended in order to make it directly executable without having to derive a business process first.

The EPM provides a compact representation of the set of possible execution paths of a business process by defining information dependencies. It has been shown in Kress & Seese (2007a) and Kress & Seese (2007b) that multi-agent methods can be used to execute an EPM.

Intelligent agents take advantage of the flexibility provided by the EPM and select their actions based on relational reinforcement learning using a probabilistic policy. The agents individually learn how the EPM is efficiently executed based on the current situation and the objectives under consideration (e.g. the minimization of the cycle time). The probabilities of policies in such a model must be known or estimated using an adequate heuristic. Genetic Algorithms (GAs) have been successfully used to find the probability values (Kress & Seese, 2007a; Kress & Seese, 2007b). However, the main drawback of using GAs is that they require a high computation time to find the optimal values. Executing the business processes on a simulation model typically takes a high computation time and when combined with GAs the computation time drastically increases.

In this chapter, we study Particle Swarm Optimization (PSO) proposed by Kennedy & Eberhart (1995) to find the probability values of the policies for executing the business process using EPM. PSO is known to be successful in solving many real-world applications (Engelbrecht, 2006). Here, we study a new PSO using a Gap Search (GS) method. Inspired from the Binary Search (BS) introduced by Hughes (2003), GS is proposed to explore the large gaps which exist in the search space. We intentionally combine GS and PSO as the function evaluations in our application take a high computation time and the GS method is known to be reasonably quick in exploring the search space. Note that dealing with expensive function evaluations can also be done in several other ways such as Parallelization or meta-modelling (Alba & Tomassini, 2002; Cantu-Paz, 2000; Jones et al., 1998). Here, we select the GS mechanism as it is and show that it can easily be integrated into our approach.

An approach very similar to GS is a subdivision method, proposed by Schütze et al. (2003), where the search space is divided into different boxes. Through several iterations, the non-optimal boxes are removed and the good boxes are again divided into smaller boxes. This method requires a very high computation time for relatively low dimensional search spaces. The advantage of GS is that it has a very simple structure and can easily be implemented. We combine GS in several parts of PSO such as initialization, diversity, and feasibility preservation methods and study their influences on the results of the PSO.

The rest of the chapter is structured as follows. In the next section, we briefly present PSO and GS methods. We then explain the hybrid approach in Section 3. Section 4 describes the intelligent business execution model and the main application of this chapter. Sections 5 and 6 are dedicated to the experiments and conclusions, respectively.