Expert Systems with Applications 41 (2014) 2496–2513

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Solving system-level synthesis problem by a multi-objective estimation of distribution algorithm

Ling Wang [a,*], Chen Fang [a], Ponnuthurai Nagaratnam Suganthan [b], Min Liu [a]

[a] *Tsinghua National Laboratory for Information Science and Technology (TNList), Department of Automation, Tsinghua University, Beijing 100084, China*
[b] *School of Electrical and Electronic Engineering, Nanyang Technological University, 639798 Singapore, Singapore*

ARTICLE INFO

Keywords:
Estimation of distribution algorithm
Probability model
System-level synthesis problem
Project scheduling
Image compression standard

ABSTRACT

In this paper, the system-level synthesis problem (SLSP) is modeled as a multi-objective mode-identity resource-constrained project scheduling problem with makespan and resource investment criteria (MOMIRCPSP-MS-RI). Then, a hybrid Pareto-archived estimation of distribution algorithm (HPAEDA) is presented to solve the MOMIRCPSP-MS-RI. To be specific, the individual of the population is encoded as the activity-mode-priority-resource list (AMPRL), and a hybrid probability model is used to predict the most promising search area, and a Pareto archive is used to preserve the non-dominated solutions that have been explored, and another archive is used to preserve the solutions for updating the probability model. Moreover, specific sampling mechanism and updating mechanism for the probability model are both provided to track the most promising search area via the EDA-based evolutionary search. Finally, the modeling methodology and the HPAEDA are tested by an example of a video codec based on the H.261 image compression standard. Simulation results and comparisons demonstrate the effectiveness of the modeling methodology and the proposed algorithm.

1. Introduction

With the development of VLSI (Very-large-scale integration) technology, semiconductor companies like Intel can build large-scale, complex electronic systems which contain millions of transistors on a single chip. Meanwhile, due to the increasing system complexities, the system-level synthesis problem (SLSP) has emerged (Gerstlauer et al., 2009). There is a need for moving to the system level of abstraction in order to increase productivity in electronic system design. Different from high-level synthesis which is devoted to mapping behavioral description to resistor transistor logic (RTL) (Rosado-Munoz, Bataller-Mompeán, Soria-Olivas, Scarante, & Guerrero-Martínez, 2011), system-level synthesis considers the system hardware and software design simultaneously. Recently, some tools have emerged to realize and support the system-level synthesis process, such as Daedalus (Nikolov, Stefanov, & Deprettere, 2008; Nikolov et al., 2008), SoC Environment (SCE) (Dömer et al., 2008), and SystemCoDesigner (Keinert et al., 2009).

To solve the SLSP, the mixed integer linear programming (MILP) is widely used (Schwiegershausen, Kropp, & Pirsch, 1996; Niemann & Marwedel, 1997; Nagaraj Shenoy, Banerjee, & Choudhary, 2000). However, it has some disadvantages in solving the SLSP with the MILP. First, the MILP can only solve the small-scaled problems (no more than 20 tasks) in a reasonable computation time since the SLSP is NP-hard in a general case (Mann & Orbán, 2003). Second, it is difficult for the MILP to solve the SLSP with multiple objectives. As a result, it usually adopts the MILP to solve one chosen objective, and then uses the high-level synthesis tools to solve other objectives. But the MILP-based procedure still cannot guarantee the Pareto optimal solutions.

Since many real world problems are difficult to solve by traditional methods, soft computing has gained much attention during recent years in many fields, such as controller design (Wang & Li, 2011), engineering design (Zhao & Wang, 2011; Zhao, Wang, Zeng, & Fan, 2012), steelmaking scheduling (Pan, Wang, Mao, Zhao, & Zhang, 2013), and economic load dispatch (Wang & Li, 2013). During the past few years, evolutionary algorithm (EA) has also been used to solve the SLSP. Blickle (1996) first developed a single-objective EA, and later Blickle, Teich, and Thiele (1998) introduced a Pareto-ranking technique into the single-objective EA. Fan, Wang, Achiche, Goodman, and Rosenberg (2008) introduced the flow of a structured Micro-Electro-Mechanical Systems (MEMS) design process to emphasize the system-level lumped-parameter model synthesis. To trade off the predefined behavioral specifications for designers, at the system level an approach combining bond graphs and genetic programming can yield satisfactory design candidates. Zitzler and Thiele (1999) developed a multi-objective algorithm named SPEA to solve the SLSP by combining several features of

---

* Corresponding author. Tel.: +86 10 62783125; fax: +86 10 62786911.
E-mail address: wangling@mail.tsinghua.edu.cn (L. Wang).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.09.049

previous multi-objective EAs in a unique manner. Compared to the MILP-based procedures, the EA-based procedures can deal with the large-scaled and multi-objective problems. However, the specific search operators should be designed for the EAs to solve the SLSP due to the complicated constraints. So, it is important to develop novel methodologies to model the problem reasonably as well as powerful solution algorithms to solve the problem effectively.

As a novel evolutionary algorithm, estimation of distribution algorithm (EDA) can be regarded as a general framework of statistical learning based optimization algorithm (Larrañaga & Lozano, 2002). Unlike genetic algorithm (GA) which explicitly generates new individuals by crossover and mutation, the EDA tries to predict of the movement of population in the search space and estimates the underlying probability distribution of the encoded variables of the elite individuals so as to generate new individuals. So far, the EDA has been applied to solve a variety of optimization problems in academic and industrial fields, such as feature selection (Armañanza et al., 2011), shop scheduling (Wang, Wang, Xu, Zhou, & Liu, 2012), nurse rostering (Aickelin, Burke, & Li, 2007), hybrid electric vehicle charging (Su & Chow, 2012), multi-speed planetary transmission (Simionescu, Beale, & Dozier, 2006), knapsack problem (Wang, Wang, & Xu, 2012), and software testing (Sagarna & Lozano, 2005). However, to the best of our knowledge, there is no work about EDA to solve the SLSP.

In this paper, the SLSP is solved by adopting the project scheduling concept-based model and using the EDA-based search method. First, the SLSP is modeled as a multi-objective mode-identity resource-constrained project scheduling problem with makespan and resource investment criteria (MOMIRCPSP-MS-RI). Then, a hybrid Pareto-archived estimation of distribution algorithm (HPAEDA) is proposed to solve the problem. The activity-mode-priority-resource list (AMPRL) is used to encode individuals, and a hybrid probability model is designed to predict the promising search area. During the search procedure, a Pareto archive is employed to preserve the non-dominated solutions that have been explored, and another archive is used to preserve the solutions for updating the probability model. Specific sampling and updating mechanisms are designed to make the evolution process track the most promising search areas. The modeling methodology and the proposed HPAEDA are tested with the example of a video codec based on the H.261 image compression standard. Simulation results and comparisons demonstrate the effectiveness of the modeling methodology and the proposed HPAEDA.

The remainder of the paper is organized as follows: In Section 2, the system-level synthesis problem is introduced. In Section 3, the project scheduling model for the system-level synthesis problem is described. Following the original EDA introduced in Section 4, the HPAEDA is presented in details in Section 5. An example of a video codec design based on the H.261 image compression standard is provided in Section 6. Finally, the paper is ended with some conclusions and future work in Section 7.

2. System-level synthesis problem

The system-level synthesis problem (SLSP) can be described using the "double roof" model, which is illustrated in Fig. 1.

The "double proof" model (Gerstlauer et al., 2009) describes the top-down hardware and software design process of electronic system in an ideal case. One side of the roof corresponds to the software design process; while the other side corresponds to the hardware design process. Both sides contain different abstract layers. A design specification is transformed into an implementation on each abstract layer (vertical arrow). The implementation of each abstract layer is transferred to the next abstract layer as the corresponding design specifications (horizontal arrow).

The SLSP is the top level of the electronic design, which is concerned with how to map the task specification onto the related hardware architecture so as to provide an electronic design specification with high performance and low cost. The specification of electronic design contains three parts (Blickle et al., 1998):

(1) Software behavioral design specification. The software specification of an electronic system is defined by two kinds of tasks, i.e., function tasks and communication tasks. The function task defines the related function module of the electronic system, and the communication task defines the data flow between the function modules.
(2) Hardware architecture design specification. The hardware architecture consists of different hardwares, such as Reduced Instruction Set Chip (RISC), Digital Signal Processor (DSP), Application Specific Integrated Circuit (ASIC), BUS, Random-access Memory (RAM), and so on. Each hardware $k = 1, 2, \ldots, K$ has a cost c_k.
(3) Mapping. The mapping between software and hardware can be defined by a set of Boolean functions. If $map(j,k) = 1$, the task j can be carried out on hardware k; else, the task j cannot be carried out on hardware k. Additionally, the delay function $lt(j,k)$ defines the delay when task j is carried out on hardware k, which is the execution time of task j on hardware k.

The goal of the SLSP is to find an implementation of electronic system to minimize the hardware cost and the system delay (Teich, 2000). The implementation of electronic system can be defined by the hardware implementation and the binding implementation.

(1) Hardware implementation: the hardware set to carry out the electronic system.
(2) Binding implementation: the binding between hardware and software, which is how to choose hardware to carry out the software tasks.

An example of the SLSP is illustrated in Fig. 2 by slightly modifying the example in Blickle et al. (1998). The left hand of Fig. 2 is the design specification of the SLSP. The behavioral design specification is described by a directed graph containing 7 nodes. The white nodes represent function tasks, and the gray nodes represent communication tasks. The hardware architecture contains one RISC, one DSP, one ASIC, and two BUS (BUS1 and BUS2). The arcs between hardware and software define the mapping. For example, task 7 can be executed on any chip (RISC, DSP, ASIC); task 1 can only executed on RISC. The number on each arc is the related delay time. The right hand of Fig. 2 is an electronic system implementation. This implementation adopts all the hardware except BUS2. All the communication tasks are carried out on BUS1.

3. Project scheduling model

The resource-constrained project scheduling problem (RCPSP) is concerned with single-item or small batch production where scarce resources have to be allocated to dependent activities over time (Brucker, Drexl, Möhring, Neumann, & Pesch, 1999). The RCPSP has many extensions, such as multi-mode RCPSP (Wang & Fang, 2011), multi-objective RCPSP (Ballestín & Blanco, 2011), stochastic RCPSP (Ballestín, 2007). The RCPSP comes from practice. The construction of Maya temples in Central and South America and the pyramids of ancient Egypt can be considered as the earliest project scheduling problem (Demeulemeester & Herroelen, 2002). Nowadays, the RCPSP and its extensions widely exist in various industries and service fields, such as medical research experiments