Journal of Scientific & Industrial Research
Vol. 66, December 2007, pp. 989-994

Case-based evolutionary design approach for satellite module layout

Bo Jin[1] and Hong-Fei Teng[2]*

[1]Department of Computer Science Engineering, [2]School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, P R China

Received 29 May 2006; revised 07 September 2007; accepted 17 September 2007

A case retrieval algorithm has been developed for reuse of previous stored design solutions and to optimize layout of satellite module using prior knowledge and evolutionary approach. Case-based approach has been found more powerful than a non-case used evolutionary computation. A prototypical of a 2-D layout design of a satellite was used to show effectiveness and efficiency of this approach.

Keywords: Case-based reasoning, Case retrieval, Evolutionary algorithms, Layout design

Introduction

Various attempts have been made to apply evolutionary algorithms (EAs) to complex engineering layout designs[1]. EAs are powerful optimisation techniques inspired by genetics and natural selection while lacking some important characteristics (prior knowledge) that make other classical and numerical search algorithms very effective[2].

Case-based reasoning (CBR) technique[3] is to solve the problem by retrieving previously solved cases, which are similar to the current problem. CBR has been used in various optimization problems[4], such as, layout design problems and nesting problems. In CBR process, retrieved cases should be adapted and then reused as solutions for new problems[5]. Therefore, key to achieving the best performance from system is to combine retrieved cases and algorithms with the adaptation process. Pearce[6] analyzed effect of injecting cases to the population of EAs. Ramscy[7] proposed a method of using cases as initial population of EAs. Rosenman[8] used cases not only in the initial population, but also in the process of EAs, and applied the method to building design problems. Sun[9] proposed a modified genetic algorithm to generate an initial layout scheme for layout design. A human-algorithm-knowledge based layout design method[10] (HAKD) has been proposed to afford prior knowledge solutions for EA.

This study aims to improve convergence time of EAs and discusses how CBR techniques can be used to extract reusable cases, which will be injected into evolutionary process of design circuits. Taking layout design of a satellite module[11] (Fig. 1), this paper presents a similarity algorithm for layout case retrieval. A prototype of a 2-D layout design of satellite module is used to test the efficacy and efficiency of this approach.

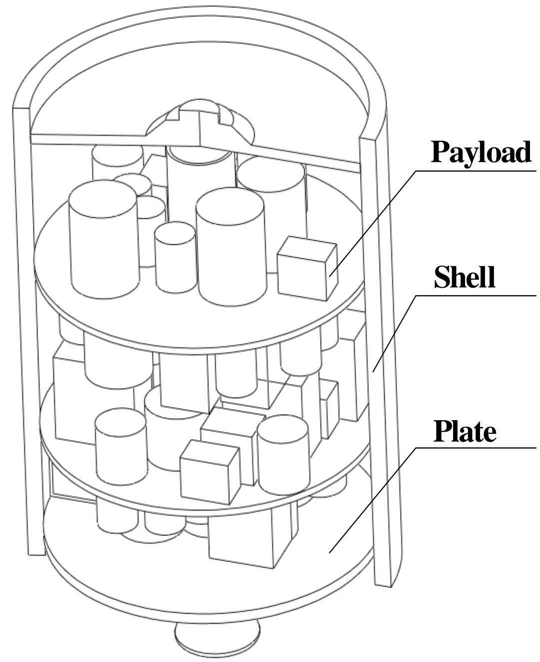


Fig. 1—Schematic layout pattern of a simplified satellite module

*Author for correspondence
E-mail: tenghf@dlut.edu.cn

System Architecture

In this study, case-based evolutionary reasoning layout (CERL) system mainly has two modules: i) case retrieval module (CRM); and ii) evolutionary computation module (ECM). For case retrieval, similarity between problems and cases can be defined over the space of problems or the space of cases. Classic CBR systems operate by defining similarity over the space of problems and hence are problem or domain dependent. CRM uses similarity metrics over the case space. A case adaptation process is employed for case-based EAs after cases are retrieved. EAs are employed as layout optimization approach for ECM. EAs are sufficiently powerful to optimize layout of satellite structures and to produce desired patterns. ECM is found by applying the simple genetic algorithms[10].

Similarity Algorithm for Layout Case Retrieval

In CERL system, developed for layout design of satellite module, in general, each satellite module consists of following three main components: 1) performance index; 2) layout space; and 3) set of payloads. Layout design problem for the satellite module can be divided into three sub-problems. Let a layout design problem P be a 3-tuple (I_C, S, L), where I_C is similarity metric of performance indices, S is shape similarity of layout space, and L is match degree between payload sets. Similarity algorithm for layout case retrieval is defined as:

$$SimC = \alpha \cdot I_C + \beta \cdot S + \gamma \cdot L \qquad \text{...(1)}$$

where α, β and γ are weighting factors.

Generally, similarity measurement between cases is defined using the rate at which the values provided from corresponding indices match[12]. Let c and c' be the performance indices of two cases. Similarity metric I_C is defined as:

$$I_C = \frac{1}{n} \times \sum_{i=1}^{n} w(v_i) \frac{c(v_i) - c'(v_i)}{c(v_i) + c'(v_i)} \qquad \text{...(2)}$$

where $c(v_i)$ and $c'(v_i)$ are the values of performance indices. $w(v_i)$”$[0, 1]$ is the weighting factor of value v_i.

Shape similarity is to calculate similarity metric between different graphics in the aspect of topology structure, geometric shape, and expressing function[13]. Features of shape are the foundation for judging similarity between graphics. Suppose graphic of a layout space is a simple closed curve and consists of basic geometric elements (lines, arcs, etc.), shape similarity S is defined as:

$$S = \frac{1}{M \times N} \sum_{i=1}^{M} w_i \sum_{j=1}^{N} \beta_{ij} q_{ij} \qquad \text{...(3)}$$

where M is number of features, N is number of feature elements. w_i and β_{ij} are weighting factors. q_{ij} is value of feature elements, which can be formulated as:

$$q_{ij} = \frac{|n_a - n_b|}{n_a + n_b} \qquad \text{...(4)}$$

where n_a and n_b are feature elements of shape.

Payloads of satellite module are simplified into regular shapes. Then using simplified skeleton technology[14], payload set can be transformed into strings. Let a and b be two strings, which emerge from the simplified skeletons of payload sets, match degree L is defined as:

$$L = \frac{1}{n} \sum_{i=1}^{n} \omega_i \times D(a_i, b_i) \qquad \text{...(5)}$$

where ω_i is weighting factor. $D(a_i, b_i)$ is edit-distance of two strings. Edit-distance[15] between two strings of characters is the minimum number of operations required to transform one of them into the other. For example, adaptation from “*delet*” to “*delete*” just needs one operation of inserting an “*e*”, therefore, their edit-distance is one. Edit-distance is widely used in information theory and computer science, such as computational biology, text processing, and web searching[16]. In this study, calculation of edit-distance is dealt with in the dynamic programming algorithm[17], which parses two strings and uses a cost matrix to determine the minimum cost sequence.

Experimental Details

Case base is composed of simplified satellite module schemes (Fig. 1). There are 100 cases in the case base.

Proposed Module

There are 52 payloads (Table 1) to be located on the four bearing surfaces in the satellite module. The first 31 objects are cylinders and the others are cuboids. Suppose each payload is a rigid body and the quality is equably distributed. Technological requirements for a

Table 1—Dimensions and masses of payloads

ID	Mass, kg	Dimension, mm		ID	Mass, kg	Dimension, mm		
		r_i	h_i			a_i	b_i	h_i
1	15.08	100	240	32	15.00	250	150	200
2	15.08	100	240	33	15.00	250	150	200
3	15.08	100	240	34	15.00	250	150	200
4	15.08	100	240	35	15.00	250	150	200
5	15.08	100	240	36	15.00	250	150	200
6	11.31	100	180	37	15.00	250	150	200
7	11.31	100	180	38	20.00	200	200	250
8	11.31	100	180	39	20.00	200	200	250
9	11.31	100	180	40	20.00	200	200	250
10	11.31	100	180	41	11.25	150	150	250
11	12.56	100	200	42	11.25	150	150	250
12	12.56	100	200	43	11.25	150	150	250
13	12.56	100	200	44	6.00	150	100	200
14	12.56	100	200	45	6.00	150	100	200
15	3.53	75	100	46	3.00	100	100	150
16	3.53	75	100	47	3.00	100	100	150
17	3.53	75	100	48	11.10	200	185	150
18	3.53	75	100	49	11.10	200	185	150
19	3.14	50	200	50	8.00	200	100	200
20	3.14	50	200	51	8.00	200	100	200
21	3.14	50	200	52	9.00	200	100	200
22	3.14	50	200					
23	3.14	50	200					
24	3.14	50	200					
25	3.39	60	150					
26	3.39	60	150					
27	2.04	45	160					
28	2.04	45	160					
29	11.31	100	180					
30	11.31	100	180					
31	12.56	100	200					

final layout scheme of the whole module are given as anticipant position of whole module centroid (*0*, *0*, *780*) mm, allowance for amortization of centroid position $\Delta x = \Delta y = \Delta z = \Delta 5\ mm$, and allowance for amortization of dynamic equilibrium degrees $\Delta_{\theta x} = \Delta_{\theta y} = \Delta_{\theta z} = 0.1\ rad$. Basic problem is to optimize objective function, while satisfying technological constraints (overlap, dynamic equilibrium degrees, static equilibrium).

Experiment consists of CRM and ECM. Under CRM, similar layout cases are retrieved from the case base with the given layout scheme as input. First, calculating the similarity of cases using similarity algorithm, and then judging whether the cases can be retrieved or not. ECM optimizes the layout of satellite module with EAs using retrieved cases as artificial individuals. The cases are used in the entire process (initial population and algorithm population) of EAs. A comparison experiment is performed to compare the two situations of the same algorithm; EAs with cases (artificial individuals) used and EAs with no cases (random individuals) used.

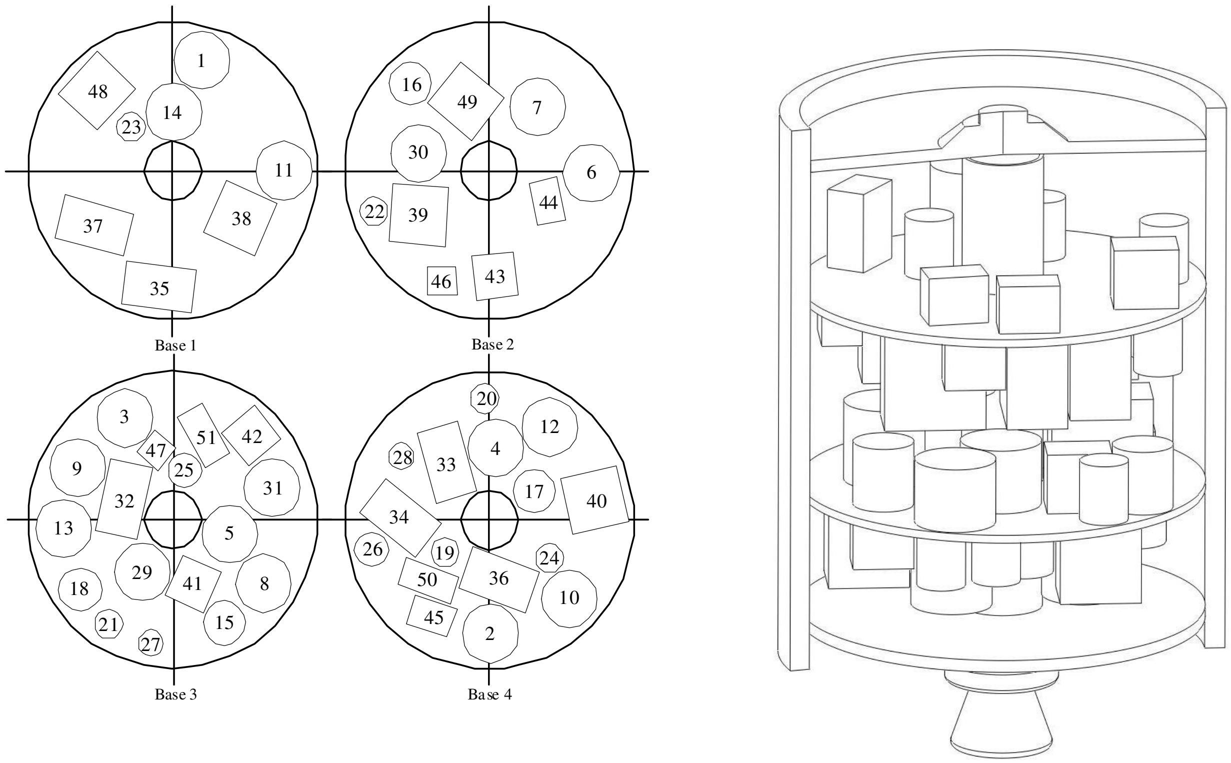


Fig. 2— Retrieved layout case

Process of Solution

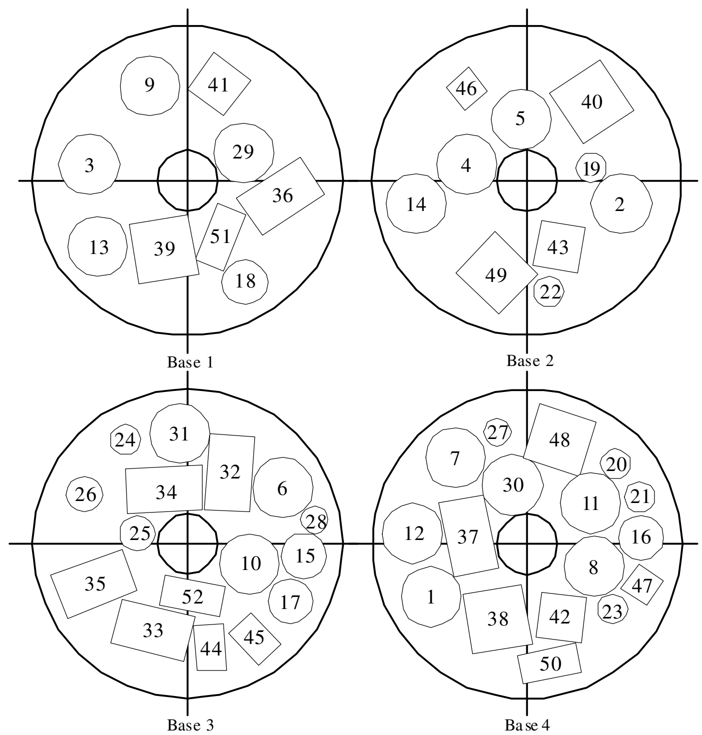
In the experiment, retrieved cases should be put into the individual base as artificial individuals. At the beginning of EAs, initial population consisted of these cases. At the meantime of EAs, these cases should be picked out from individual base and injected into the population of EAs in a certain situation, such as, when the algorithm has operated certain generations ($\Delta K = 2000$), or obviously no improvement occurs for the fitness function.

Computation of overlap is simplified into two-dimensions, while other constraints should be calculated in three-dimensions. The population size $P_{size} = 20$, and largest generation $K_{max} = 10000$. The end rule is that the generation reaches maximum number $K_{max} = 10000$, or the variety of overlap in xOy surface $\Delta S \leq 10\ mm^2$, or the variety of fitness function $|f_K - f_{K-1}|/f_K \leq 10^{-6}$. The experiment is operated on a computer with Intel P4 1.5 GHz CPU and 512M memory.

Results and Discussion

In this study, threshold value is set to 0.5. If similarity metric of a case is larger than threshold value, the case could be retrieved as a proposed solution. The most similar layout case that has been retrieved from the case base consists of 51 payloads (Fig. 2). Its layout space and performance indices are similar to that of the task layout scheme and the similarity metric is 0.813. Performances of CRM are as follows: recall, 63%; precision, 70%; and time cost, 0.17 sec. Retrieved cases need to be adjusted before injecting them into the population of EAs, for example, adding a payload to the retrieved case (Fig. 2). A comparison is made between random individuals using ECM (Fig. 3a) and artificial individuals using ECM (Fig. 3b). Resulting criteria (Table 2) are obtained after many experiments (50 times).

Proposed similarity algorithm can retrieve the reusable cases effectively, and the performance of EA is improved with retrieved cases (Table 2). With the average results from 50 calculations, rate of success reaches 42% by using the cases as artificial individuals, whereas, the traditional approach can obtain 18% success rate. The reason is that prior knowledge and experience is used, and the size of the search space is extended. This work is closely related to the cases in the case base. It is important to maintain a case base to increase the quality of cases in addition to increasing the case base coverage. Furthermore, case-based optimization



a) Result of random individuals used evolutionary computation b) Result of artificial individuals used evolutionary computation

Fig. 3—Optimal layout scheme of satellite

Table 2— Statistical data for 50 times experiments

Performance	Random individuals x	Random individuals y	Random individuals z	Artificial individuals x	Artificial individuals y	Artificial individuals z
Static equilibrium, [illegible] /mm	[illegible]	8.033×10^{-2}	9.858×10^{-1}	3.701×10^{-2}	6.850×10^{-2}	1.301×10^{-1}
Dynamic equilibrium degree, $\theta(x)$, [illegible] $\theta(z)$/rad	[illegible]	1.839×10^{-2}	-7.140×10^{-4}	-3.493×10^{-4}	1.132×10^{-2}	6.279×10^{-4}
Moment of inertia, $J(x)$, [illegible] /$Kg\cdot m^2$	[illegible]	305.094	267.188	162.389	301.966	260.297
Overlap in xOy surface, ΔS /mm^2		782.024			309.1834	
Cost time*, t/s		336.484			327.336	
Success rate, %		18			42	

*Cost time of artificial individuals [illegible] evolutionary computation does not include the cost time of case retrieval. In a certain layout design [illegible] time of case [illegible] occurs [illegible] once and usually lasts scores of milliseconds. Its value depends on the performance of case [illegible] algorithm and size of the case base.

ap[illegible] cannot assu[illegible] solution every time [illegible] there woul[illegible] of precocity [illegible]gence in the evolutionary process. These problems can potentially harm the performance of CERL system.

Conclusions

Engineering practice shows the feasibility and effectiveness of prior knowledge reuse for complicated engineering problems. Taking a layout design of a satellite module as an example, this work provides the user with a powerful tool for retrieving layout cases from the case base in order to optimize layout with these retrieved cases. This task was accomplished with two methods. First, a similarity algorithm for layout case retrieval was developed. Second, evolutionary algorithm was used for optimizing layout of payloads in the satellite module. This work has a limitation of cases in the case base. Thus, there is a need to extend the case base space in the future.

Acknowledgements

This work was supported by National High-tech R&D Program of P R China (Grant No. 2006AA04Z109), National Natural Science Foundation of P R China (Grant Nos 60674078, 50575031) and National Key Technologies R&D Program during the 11th Five-Year Plan Period of P R China (Grant No. JPPT-115-189).

References

1 Cagan J, Shimada K & Yin S, A Survey of computational approaches to three-dimensional layout problems, *Comput Aid*, **34** (2002) 597-611.

2 Poli R & Logan B, On the relations between search and evolutionary algorithms, in *Technical Report CSRP-96-7* (School of Computer Science, University of Birmingham) March 1996.

3 Aamodt A & Plaza E, Case-based reasoning: foundational issues, methodological variations, and system approaches, *AI Commun*, **7** (1994) 39-59.

4 Dipankar D & Zbigniew M, Evolutionary algorithms in engineering applications, *Evol Optimiz*, **10** (1999) 93-94.

5 Kokku R, Riche T L, Kunze A, Mudigonda J, Jason J &Vin H M, A Case for run-time adaptation in packet processing systems, *Proc 2nd Workshop on Hot Topics in Networks* (MIT, Cambridge) 2003, 107-112.

6 Pearce M, Goel A K, Kolodner I L, Zimring C, Sentosa L & Billington R, Case-based design support: a case study in architectural design, *IEEE Expert*, **7** (1992) 14-20.

7 Ramsey C L & Grefenstette J J, Case-based initialization of genetic algorithms, *Proc 5th Int Conf on Genetic Algorithms* (Morgan Kaufmann Publishers, San Mateo) 1993, 84-91.

8 Rosenman M, Case-based evolutionary design, *AI EDAM*, **14** (2000) 17-29.

9 Sun Z G & Teng H F, Optimal layout design of a satellite module, *Eng Optimiz*, **35** (2003) 513-529.

10 Zhang B, Teng H F & Shi Y J, Layout optimization of satellite module using soft computing techniques, *Appl Soft Computing*, (in press).

11 Liu Z W & Teng H F, Human-computer cooperative layout design method and its application. *Comput Ind Eng*, (in press).

12 Surma J & Braunschweig B, Case-based retrieval in process engineering: supporting design by reusing flow sheets, *Eng Appl Artif Intel*, **9** (1996) 385-391.

13 Tan J R, Yue X L & Lu G D, Basic principle, method of graphic similarity and its application to structure pattern recognition, *Chinese J Comput*, **25** (2002) 1-9.

14 Ma J, Teng H F & Liu D Q, A nesting method based on master drawings and their similarity retrieval, *Chinese J Software*, **11** (2000) 1685-1691.

15 Ristad E S & Yianilos P N, Learning string-edit distance, *IEEE PATT A*, **20** (1998) 522-532.

16 Reis D C, Golgher P B, Silva A S & Laender A F, Automatic web news extraction using tree edit distance, *Proc 13th Int Conf on World Wide Web* (ACM Press, New York) 2004, 502-511.

17 Kim S R & Park K, A dynamic edit distance table, *J Discrete Algorithms*, **2** (2004) 303-312.