多目的進化型アルゴリズムにおける解の支配領域制御と多目的0/1ナップザック問題を用いた性能解析

佐藤 寛之, エルナン アギレ, 田中 清[1]

この論文では、問題に対する解の適切なランキングを促し、選択を強調し、組合せ最適化問題におけるMOEAの性能改善を図るために、解の支配領域を制御する方法を提案している。提案法は、利用者が定義するパラメータ S を用いて、解が支配する領域を拡大したり縮小したりする程度を制御できる。解の支配領域を変更することより、従来の支配とは異なる解のランキングを導く。目的数、探索空間および問題の実行可能度が変化したとき、解の支配領域の拡大・縮小によって生ずる解のランキングへの影響と、多目的最適化器の探索性能に与える影響を解析するために、この論文では多目的0/1ナップザック問題を用いる。解の支配領域の拡大・縮小によって、収束または多様性のどちらか一方が強調されることを示す。また、支配領域の最適値は、ここで解析するすべての要因：すなわち、目的数、探索空間および問題の実行可能度に強く依存することを示す。

Controlling Dominance Area of Solutions in Multiobjective Evolutionary Algorithms and Performance Analysis on Multiobjective 0/1 Knapsack Problems

Hiroyuki Sato, Hernán Aguirre and Kiyoshi Tanaka[2]

This work proposes a method to control the dominance area of solutions in order to induce appropriate ranking of solutions for the problem at hand, enhance selection, and improve the performance of MOEAs on combinatorial optimization problems. The proposed method can control the degree of expansion or contraction of the dominance area of solutions using a user-defined parameter S. We use 0/1 multiobjective knapsack problems to analyze the effects on solutions ranking caused by contracting and expanding the dominance area of solutions and its impact on the search performance of a MOEA when the number of objectives, the size of the search space, and the feasibility of the problems vary.

1. Introduction

Multiobjective evolutionary algorithms (MOEAs) [1, 2] are being increasingly investigated for solving multiobjective optimization problems. Some important features of the latest generation MOEAs are that selection incorporates elitism and it is biased by Pareto dominance and a diversity preserving strategy in objective space. Pareto dominance based selection is thought to be effective for problems with convex and non-convex fronts and has been successfully applied, especially in two and three objective problems. However, some current research reveals that ranking by Pareto dominance on problems with an increased number of objectives might not longer be effective [3] because the fronts become substantially denser. In this case, most sampled solutions at a given time turn to be non-dominated and Pareto selection weakens since it has to discriminate mostly based on diversity of solutions. Another factor that affects the density of the fronts is the complexity (ruggedness and number of local optima) of the individual single objective landscapes, which also affects the behavior and effectiveness of conventional Pareto selection.

In this work, we propose a method to control the dominance area of solutions in order to induce appropriate ranking of solutions for the problem at hand, enhance selection, and improve the performance of MOEAs on combinatorial optimization problems. The proposed method can control the degree of expansion or contraction of the dominance area of solutions using a user-defined parameter S. Modifying the dominance area of solutions changes their dominance relation inducing a ranking of solutions that is different to conventional dominance. Related works on relaxed forms of Pareto dominance are ϵ-dominance [4] and α-domination [5]. Contrary to ϵ-dominance and α-domination, the proposed method can strengthen or weaken selection by expanding or contracting the area of dominance and conceptually can be considered as a generalization of Pareto dominance.

In this work we analyze the effects on solutions

[1]信州大学工学部
[2]Shinshu University, Faculty of Engineering

ranking caused by contracting and expanding the dominance area of solutions and its impact on the search performance of a multi-objective optimizer when the number of objectives, the size of the search space, and the complexity of the problems vary. We chose NSGA-II as a representative elitist algorithm that uses dominance [1] and compare its performance with NSGA-II enhanced by the proposed method. We conduct our study on 0/1 multiobjective knapsack problems with $m = \{2, 3, 4, 5\}$ objectives varying the number of items n (size of search space is given by 2^n) and the feasibility ratio ϕ of the search space.

2. Proposed Method

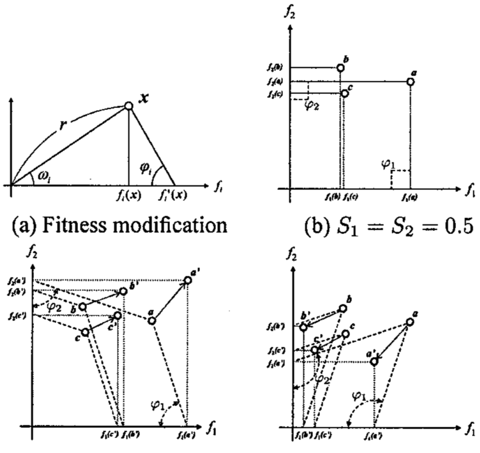
2.1 Contraction and Expansion of Dominance Area

In this work, we try to control the covered area of dominance. Normally, the dominance area is uniquely determined with a fitness vector $\boldsymbol{f}(\boldsymbol{x}) = (f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x}))$ in the objective space when a solution $\boldsymbol{x}$ is given. To contract and expand the dominance area of solutions, we modify fitness value for each objective function by changing the user defined parameter S_i in the following equation

$$f_i'(\boldsymbol{x}) = \frac{r \cdot \sin(\omega_i + S_i \cdot \pi)}{\sin(S_i \cdot \pi)} \quad (i = 1, 2, \cdots, m) \tag{1}$$

where $\varphi_i = S_i \cdot \pi$. This equation is derived from the Sine theorem. We illustrate the fitness modification in **Figure 1** (a), where r is the norm of $\boldsymbol{f}(\boldsymbol{x})$, $f_i(\boldsymbol{x})$ is the fitness value in the i-th objective, and ω_i is the declination angle between $\boldsymbol{f}(\boldsymbol{x})$ and $f_i(\boldsymbol{x})$. In this example, the i-th fitness value $f_i(\boldsymbol{x})$ is projected (increased) to $f_i'(\boldsymbol{x}) > f_i(\boldsymbol{x})$ by using $\varphi_i < \pi/2$ ($S_i < 0.5$). In case of $\varphi_i = \pi/2$ ($S_i = 0.5$), $f_i(\boldsymbol{x})$ does not change and $f_i'(\boldsymbol{x}) = f_i(\boldsymbol{x})$. Thus, this case is equivalent to the conventional dominance. On the other hand, in case of $\varphi_i > \pi/2$ ($S_i > 0.5$), $f_i(\boldsymbol{x})$ is projected (decreased) to $f_i'(\boldsymbol{x}) < f_i(\boldsymbol{x})$.

Such fitness modification changes the dominance area of solutions. We show an example in **Figure 1** (b)-(d), where three solutions $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$ are distributed in 2-dimensional objective space. In **Figure 1** (b), $\boldsymbol{a}$ dominates $\boldsymbol{c}$, but $\boldsymbol{a}$ and $\boldsymbol{b}$, and $\boldsymbol{b}$ and $\boldsymbol{c}$ do not dominate each other. However, if we modify fitness values for each solution by using **Eq.**(1), the location of each solution moves in the objective space, and consequently the dominance relationship among solutions changes. For example, if we use $S_1 = S_2 < 0.5$ as shown in **Figure 1** (c), the dominance area of solutions $\boldsymbol{a}'$, $\boldsymbol{b}'$ and $\boldsymbol{c}'$ is expanded from the original one of $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$. This causes that $\boldsymbol{a}'$ dominates $\boldsymbol{b}'$ and $\boldsymbol{c}'$, and $\boldsymbol{b}'$ dominates $\boldsymbol{c}'$. That is, expansion of dominance area by smaller $S_i (< 0.5)$ works to produce a more fine



(a) Fitness modification (b) $S_1 = S_2 = 0.5$

(c) $S_1 = S_2 < 0.5$ (d) $S_1 = S_2 > 0.5$

Figure 1: Fitness modification to change the covered area of dominance (a), conventional dominance (b) and examples of expanding (c) and contracting (d) the dominance area of solutions

grained ranking of solutions and would strengthen selection. On the other hand, if we use $S_1 = S_2 > 0.5$ as shown in **Figure 1** (d), the dominance area of solutions $\boldsymbol{a}'$, $\boldsymbol{b}'$ and $\boldsymbol{c}'$ is contracted from the original one of $\boldsymbol{a}$, $\boldsymbol{b}$ and $\boldsymbol{c}$. This causes that $\boldsymbol{a}'$, $\boldsymbol{b}'$ and $\boldsymbol{c}'$ do not dominate each other. That is, contracting the area of dominance by larger $S_i (> 0.5)$ works to produce a coarser ranking of solutions and would weaken selection.

2.2 Effects of Controlling Dominance Area

In this section we verify and illustrate the effect of expanding or contracting the dominance area on the distribution of the fronts changing the parameter S_i in **Eq.**(1). Here, we randomly generate 100 solutions in the 2-dimensional objective space of $[0, 1]^2$, calculate dominance among them after recalculating fitness with **Eq.**(1), and perform a non-domination sorting to obtain the fronts. We repeat the above steps a 1000 times and calculate the average number of fronts and solutions per front, for each value of S_i. In this work, we use a common parameter $S = S_i (i = 1, 2, \cdots, m)$ for all objective functions, because we assume that all objective functions are normalized. **Figure 2** shows the fraction of number of solutions per front varying S in the range $[0.25, 0.75]$ in intervals of 0.1 along with results for conventional dominance ($S = 0.5$).

From this figure, note that if we gradually expand the area of dominance by decreasing S below 0.5, the number of fronts increases and the ranking of solutions by non-dominance can be fine grained. Note that for maximum expansion of the dominance area $S = 0.25$ there is one solution per front. On the other hand, if

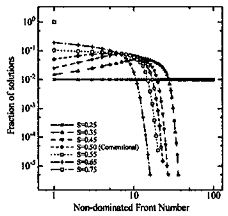


Figure 2: Solutions per front varying the parameter S

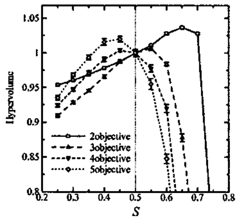


Figure 3: Hypervolume as we increase the number of objectives m for problems with n = 500 items and $\phi = 0.5$ feasibility ratio

we gradually contract the area of dominance by increasing S above 0.5, the number of fronts decreases and ranking of solutions by non-dominance becomes coarser. Note that for maximum contraction of the dominance area $S = 0.75$ there is only one front that contains all solutions. Since different rankings can be produced, we can expect that the optimum parameter S^* that yields maximum search performance exists for a given kind of problem.

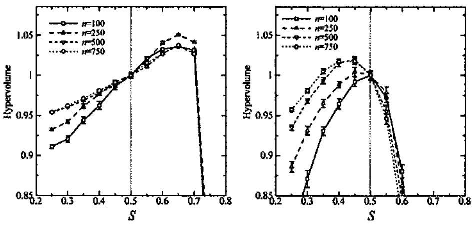
3. Experimental Results and Discussion

3.1 Benchmark Problems and Parameters

In this paper we use multiobjective 0/1 knapsack problems [6] with $m = \{2,3,4,5\}$ objectives, $n = \{100,250,500,750\}$ items and feasibility ratio of the search space $\phi = \{0.75, 0.5, 0.25\}$. We use a constant S for all objectives because the scale of each objective function is similar. In our study we compare the performance of a conventional NSGA-II [1] with NSGA-II enhanced by the proposed method. We adopt two-point crossover with a crossover rate $p_c = 1.0$ for recombination, and apply bit-flipping mutation with a mutation rate $p_m = 1/n$. We show the average performance with 30 runs, each of which spent 2,000 generations. Population size is set to $|P| = 200$ and the parent and offspring population sizes $|Q|$ and $|R|$ are set to half the population size $|P|$, i.e. $|Q| = |R| = 100$.

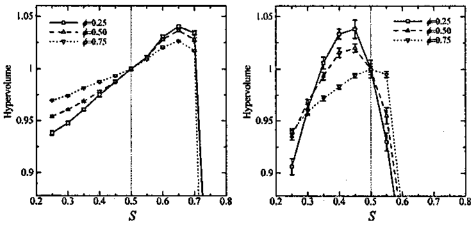
3.2 Performance Varying m

First, we observe the effect of varying S on problems with different number of objectives. **Figure 3** shows



(a) m = 2 objectives (b) m = 5 objectives

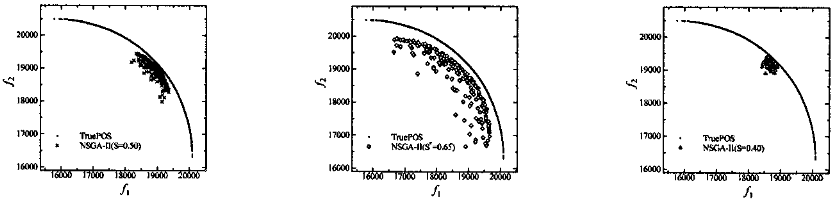
Figure 4: Hypervolume as we increase the number of items n for problems with $m = \{2, 5\}$ objectives and $\phi = 0.5$ feasibility ratio



(a) m = 2 objectives (b) m = 5 objectives

Figure 5: Hypervolume as we decrease feasibility ratio for problems with $m = \{2, 5\}$ objectives and n = 500 items

the values of the hypervolume achieved varying S in the range [0.25, 0.75] in intervals of 0.05 on problems with $m = \{2,3,4,5\}$ objectives, n = 500 items, and feasibility ratio $\phi = 0.50$. Note that in the figure the values of the hypervolume are normalized so that the value achieved at $S = 0.5$ is always 1.0. From this figure important observations are as follow. First, there is an optimum value S^* for each number of objectives that maximizes the hypervolume. Note however that the maximum value of hypervolume is not achieved by conventional dominance ($S = 0.5$) for any number of objectives. Second, to achieve the maximum value of hypervolume, the degree of expansion or contraction of dominance area of solutions should be adjusted accordingly to the number of objectives. Note that maximum values of the hypervolume are achieved for two and three objectives by contracting the dominance area of the solutions ($S > 0.5$), whereas for four and five objectives the maximum hypervolume values are achieved by expanding the dominance area of the solutions ($S < 0.5$). Third, as a general trend in problems with n = 500 items and feasibility ratio $\phi = 0.50$, we observe that the optimum value S^* reduces as we increase the number of objectives. That is, increasing the number of objectives the area of dominance should be expanded by using smaller values of S^* to achieve maximum hypervolume.



(a) $S = 0.5$ (b) $S^* = 0.65$ (c) $S = 0.40$

Figure 6: Obtained solutions by conventional dominance $S = 0.5$, contracting dominance $S^* = 0.65$, and expanding dominance $S = 0.4$ for $m = 2$ objectives, $n = 500$ items, and $\phi = 0.5$ feasibility ratio

3.3 Performance Varying n

Second, we observe the effects of varying S on problems with different number of items n. Note that the size of the search space is given by 2^n. **Figure 4** shows the hypervolume varying S on problems with $n = \{100, 250, 500, 750\}$ items and feasibility ratio $\phi = 0.5$ for $m = \{2, 5\}$ objectives. From **Figure 4** (a) we can see that in the case of $m = 2$ objectives the optimum S^* is similar for all n, around 0.65. However, we observe that increasing the number of items n produces a clear shift of the optimum S^* towards smaller values (greater expansion of area of dominance), especially in the case of $m = 4$ and $m = 5$ objectives as illustrated in **Figure 4** (b).

3.4 Performance Varying ϕ

Third, we observe the effects of varying S on problems with different feasibility ratio ϕ. **Figure 5** shows the hypervolume varying S on problems with feasibility ratio $\phi = \{0.75, 0.5, 0.25\}$ and $n = 500$ items for $m = \{2, 5\}$ objectives. From **Figure 5** (a)-(b) note that the effects on problems with different feasibility ratio ϕ resemble those observed on problems with different number of items. That is, in $m = 2$ objectives the optimum S^* is the same for all ϕ. However, reducing the feasibility ratio ϕ from 0.75 to 0.25, there is a shift of the optimum S^* towards smaller values, which becomes more notorious for $m = 4$ and $m = 5$ objectives.

3.5 Obtained Solutions

Figure 6 illustrates the obtained solutions in the final generation ($t = 2000$) for all 30 simulations by conventional dominance $S = 0.5$, contracting dominance $S^* = 0.65$, and expanding dominance $S = 0.4$ for $m = 2$ objectives, $n = 500$ items, and $\phi = 0.5$ feasibility ratio. Note that solutions obtained by conventional dominance are close to the true Pareto front but are clustered in a limited region of objective space. By contracting dominance with the optimum parameter $S^* = 0.65$, we can spread the obtained solutions showing the maximum hypervolume, although convergence of some of them seems to deteriorate. On the other hand, by expanding dominance with $S = 0.4$ showing worst spread, we can focus on convergence of solutions within a narrower region of objective space.

4. Conclusions

We have proposed a method that can control dominance area of solutions by a user defined parameter S. We showed that contracting or expanding the dominance area of solutions changes their dominance relation, modifying the distribution of solutions in the multiobjective landscape. Also, using 0/1 multiobjective knapsack problems we showed that the optimum value of S^* depends strongly on number of objectives, size of the search space, and feasibility ratio of the search space. Moreover, we showed that either convergence or diversity can be emphasized by contracting or expanding the dominance area rather than by using conventional dominance. As future works, we would like to investigate the effect of varying S_i for each objective and combine the proposed method with other selection methods to achieve higher convergence while covering the whole true Pareto front.

References

[1] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*, John Wiley & Sons, 2001.

[2] C. Coello, D. Van Veldhuizen, and G. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, 2002.

[3] E. J. Hughes, "Evolutionary Many-Objective Optimisation: Many Once or One Many?", *Proc. 2005 IEEE Congress on Evolutionary Computation*, Vol.1, pp.222-227, September 2005.

[4] M. Laumanns, L. Thiele, K. Deb and E. Zitzler, "Combining Convergence and Diversity in Evolutionary Multi-objective Optimization", *Evolutionary Computation*, Vol.10, No.3, pp.263-282, Fall 2002.

[5] K. Ikeda, H. Kita and S. Kobayashi, "Failure of Pareto-based MOEAs: Does Non-dominated Really Mean Near to Optimal?", *Proc. 2001 IEEE Congress on Evolutionary Computation*, Vol. 2, pp.957-962, May 2001.

[6] http://www.tik.ee.ethz.ch/~zitzler/testdata.html