Computers and Operations Research 90 (2018) 72–83

Contents lists available at ScienceDirect

Computers and Operations Research

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

Improved quick hypervolume algorithm

Andrzej Jaszkiewicz

Poznan University of Technology, Faculty of Computing, Institute of Computing Science, ul. Piotrowo 2, Poznan, 60-965, Poland

ARTICLE INFO

Article history:
Received 4 May 2017
Revised 10 August 2017
Accepted 14 September 2017
Available online 15 September 2017

Keywords:
Multiobjective optimization
Hypervolume indicator
Computational complexity analysis

ABSTRACT

In this paper, we present a significant improvement of the Quick Hypervolume algorithm, one of the state-of-the-art algorithms for calculating the exact hypervolume of the space dominated by a set of d-dimensional points. This value is often used as the quality indicator in the multiobjective evolutionary algorithms and other multiobjective metaheuristics and the efficiency of calculating this indicator is of crucial importance especially in the case of large sets or many dimensional objective spaces. We use a similar divide and conquer scheme as in the original Quick Hypervolume algorithm, but in our algorithm we split the problem into smaller sub-problems in a different way. Through both theoretical analysis and a computational study we show that our approach improves the computational complexity of the algorithm and practical running times.

1. Introduction

In this paper, we consider the problem of calculating the exact hypervolume of the space dominated by a set of d-dimensional points. This hypervolume is often used as the quality indicator in the multiobjective evolutionary algorithms (MOEAs) and other multiobjective metaheuristics (MOMHs), where the set of points corresponds to images in the objective space of the solutions generated by a MOMH. Multiple quality indicators have been proposed in the literature, however, the hypervolume indicator has the advantage of being compatible with the comparison of approximation sets based on the dominance relation (see Zitzler et al., 2003 for details) and is one of the most often used indicators. The hypervolume indicator may be used a posteriori to evaluate the final set of solutions generated by a MOMH e.g. for the purpose of a computational experiment comparing different algorithms or to tune parameters of a MOMH. Some authors proposed also indicator-based MOMHs that use the hypervolume to guide the work of the algorithms (Jiang et al., 2015; Zitzler and Künzli, 2004).

The exact calculation of the hypervolume may become, however, computationally demanding especially in the case of large sets in many dimensional objective spaces. Thus the exact calculation of the hypervolume obtained a significant interest from the research community (Beume et al., 2009; Chan, 2013; Lacour et al., 2017; Russo and Francisco, 2014; 2016; While and Bradstreet, 2012). According to the recent study of Lacour et al. (2017) the state-of-the-art algorithms for the exact calculation of the hypervolume are Quick Hypervolume (QHV) (Russo and Francisco, 2014, 2016), Hypervolume Box Decomposition Algorithm (HBDA) (Lacour et al., 2017) and Walking Fish Group algorithm (WFG) (While et al., 2012). From the theoretical point of view the currently most efficient algorithm in the case $d \geq 4$ in terms of the worst case complexity ($\mathcal{O}(n^{\frac{d}{3}}\text{polygon}(n))$ is by Chan (2013). To our knowledge, there is currently, however, no available implementation of this approach and no evidence of its practical efficiency (Lacour et al., 2017).

In this paper, we improve QHV algorithm by modifying the way the problem is split into smaller sub-problems. This modification although may seem relatively simple significantly improves the computational complexity of the algorithm and practical running times. Since our work is based on relatively recently published results we do not give in this paper an extended overview of the applications of the hypervolume indicator and the algorithms for the hypervolume calculation. Instead we refer an interested reader to Russo and Francisco (2014), Russo and Francisco (2016), Beume et al. (2009), While and Bradstreet (2012), Lacour et al. (2017) and While et al. (2012) for recent overviews of this area.

The paper is organized in the following way. In the next section, we define the problem of the hypervolume calculation. In Section 3, the improved Quick Hypervolume (QHV-II) algorithm is proposed. The computational complexity of QHV-II algorithm is analyzed and compared to QHV in Section 4. In Section 5, a computational study is presented. The paper finishes with conclusions and directions for further research.

E-mail addresses: andrzej.jaszkiewicz@put.poznan.pl, jaszkiewicz@cs.put.poznan.pl, andrzej.jaszkiewicz@cs.put.poznan.pl

http://dx.doi.org/10.1016/j.cor.2017.09.016
0305-0548/

2. Problem formulation

Consider a d-dimensional space $\mathbb{R}^d$ that will be interpreted as the space of d maximized objectives.

We say that a point $s^1 \in \mathbb{R}^d$ *dominates* a point $s^2 \in \mathbb{R}^d$ if, and only if, $s^1_j \geq s^2_j \ \forall j \in \{1, \ldots, d\} \wedge \exists j \in \{1, \ldots, d\} : s^1_j > s^2_j$. We denote this relation by $s^1 \succ s^2$.

We will consider *hypercuboids* in $\mathbb{R}^d$ parallel to the axes, defined by two extreme points $r_* \in \mathbb{R}^d$ and $r^* \in \mathbb{R}^d$ such that $H(r^*, r_*) = \{s \in \mathbb{R}^d \mid \forall j \in \{1, \ldots, d\}\ r_{*j} \leq s_j \leq r^*_j\}$.

Consider a finite set of points $S \subset H(r^*, r_*)$. The hypervolume of the space dominated by S within hypercuboid $H(r^*, r_*)$, denoted by $\mathcal{H}(S, H(r^*, r_*))$ is the Lebesgue measure of the set $\bigcup_{s \in S} H(s, r_*)$. The introduction of r^* may seem superfluous since it does not influence the hypervolume, however, such definition will facilitate further explanation of the algorithms which are based on the idea of splitting the original problem into sub-problems corresponding to smaller hypercuboids.

3. Quick Hypervolume II algorithm

In this section we propose a modification of the QHV algorithm proposed by Russo and Francisco (2014); 2016). We call this modified algorithm QHV-II. Both QHV and QHV-II are based on the following observations:

1. $\forall s' \in S \quad \mathcal{H}(S, H(r^*, r_*)) = \mathcal{H}(s', r_*) + \mathcal{H}\Big(\big(\bigcup_{s \in S \setminus \{s'\}} H(s, r_*)\big) \setminus H(s', r_*)\Big)$, i.e. the hypervolume of the space dominated by S is equal to the hypervolume of the hypercuboid defined by a single point $s' \in S$ and r_*, i.e. $\mathcal{H}(s', r_*)$, plus the hypervolume of the area dominated by the remaining points, i.e. $S \setminus \{s'\}$ excluding the area of hypercuboid $H(s', r_*)$.
2. The region $H(r^*, r_*) \setminus H(s', r_*)$ may be defined as a union of non-overlapping hypercuboids $\{H_1, \ldots, H_L\}$.
3. Consider a point $s^1 \notin H(r^*, r_*) \wedge s^1 \succ r_*$. The hypervolume of the space dominated by s^1 within $H(r^*, r_*)$ is equal to the hypervolume of the space dominated by the projection of s^1 onto $H(r^*, r_*)$. The projection means that the coordinates of the projected point larger than the corresponding coordinates of r^* are replaced by the corresponding coordinates of r^*.

The above observations immediately suggest the possibility of calculating the hypervolume in a recursive manner with Algorithm 1. The algorithm selects a pivot point, calculates the hypervolume of the area dominated by the pivot point, and then splits the problem of calculating the remaining hypervolume into a number of sub-problems. If the number of points is sufficiently small it uses simple geometric properties to calculate the hypervolume.

Algorithm 1 General QHV.

```
Parameters ↓: H(r*, r_*), S ⊂ H(r*, r_*)

if S contains one or two points then
  Calculate 𝓗(S, H(r*, r_*)) using simple geometric properties
  HyperVolume ← 𝓗(S, H(r*, r_*))
else
  Select a pivot point s' ∈ S
  HyperVolume ← 𝓗(s', r_*)
  Split H(r*, r_*) \ H(s', r_*) into a set of non-overlapping hyper-
  cuboids {H_1, ..., H_L}.
  for all H_l ∈ {H_1, ..., H_L} do
    Construct set S_l containing the points dominating r^l_* and if
    necessary projected onto H_l
    HyperVolume ← HyperVolume + QHV(H_l, S_l)
return HyperVolume
```

Russo and Francisco (2014, 2016) propose to split the region $H(r^*, r_*) \setminus H(s', r_*)$ into $2^d - 2$ hypercuboids corresponding to each possible combination of the comparisons on each objective, where a coordinate may be $<$ or $\geq$ than the corresponding coordinate of the pivot point s', with the exception of the two combinations corresponding to the areas dominated and dominating s'. Each such hypercuboid may be defined by a binary vector where 0 at jth position means that $s_j < s'_j$ and 1 at jth position means that $s_j \geq s'_j$. We will call such hypercuboids *basic hypercuboids*.

We propose a different splitting scheme. We split the region $H(r^*, r_*) \setminus H(s', r_*)$ into d hypercuboids defined in the following way:

- H_1 is defined by the condition $s_1 \geq s'_1$
- ...
- H_j is defined by the conditions $s_l < s'_l \quad \forall l = 1, \ldots, j-1 \wedge s_j \geq s'_j$
- ...

In other words, the hypercuboids are defined not by binary vectors but by the following schemata of binary vectors:

- $v_1 = 1 * \cdots *$
- ...
- $v_i = 0 \ldots 01 * \cdots *$, with 1 at jth position
- ...
- $v_d = 0 \ldots 01$

where * means any symbol either 0 or 1. Each of these hypercuboids is obviously an union of a number of the basic hypercuboids.

The difference between the splitting schemes in QHV and QHV-II is graphically illustrated in Fig. 1 for the 3-objective case. In this case, there are 6 basic hypercuboids. The colors describe the hypercuboids corresponding to the different sub-problems. Please note, that in the case of QHV-II the hypercuboid defined by the condition $s_1 \geq s'_1$ contains also the region dominating s' but this region does not contain any points. In addition, the arrows indicate the directions of projections of the points onto the hypercuboids.

Alike proposed in Russo and Francisco (2014, 2016) as the pivot point we select the point $s' \in S$ with the maximum $\mathcal{H}(s', r_*)$.

Please note, that the points projected onto a hypercuboid H_l may become dominated since some coordinates are replaced with lower values. Russo and Francisco (2014, 2016) propose to explicitly remove the dominated points e.g. with the algorithm proposed in Bentley (1980). We do not use this step in QHV-II since we did not find it practically beneficial in the preliminary computational experiments. Note, however, that alike (Russo and Francisco, 2014; 2016) we implemented only a naive method for the removal of the dominated points by comparing all pairs of points. This could perhaps be improved by using more advanced methods. Please note, however, that the pivot point s' selected in the way described above is guaranteed to be non-dominated within S. Furthermore, while assigning points to sets S_l each point is compared to s' and the points dominated by s' may be removed. It does not guarantee an immediate removal of all dominated points but finally all dominated points will be removed by the algorithm because each of the dominated points will be dominated by one of the selected pivots or eliminated while using the simple geometric properties when the number of points is sufficiently small.

Please also note, that as suggested in Russo and Francisco (2014, 2016) the projected points do not need to be constructed explicitly, but their coordinates may be calculated on demand to reduce the memory requirements.