Multi-objective Bayesian global optimization for continuous problems and applications

Yang, K.

Author: Yang, Kaifeng
Title: Multi-objective Bayesian global optimization for continuous problems and applications
Date: 2017-12-06

Chapter 6

EHVI Gradient Calculation

The Expected Hypervolume Improvement (EHVI) is a frequently used infill criterion in Multi-Objective Bayesian Global Optimization (MOBGO), due to its good ability to lead the exploration. Recently, the computational complexity of EHVI calculation is reduced to $O(n \log n)$ both in 2-D and 3-D cases. However, the maximization of EHVI, which is carried out in each iteration of the algorithm, still requires a significant amount of time. This chapter introduces a formula for the Expected Hypervolume Improvement Gradient (EHVIG) and proposes an efficient algorithm to calculate EHVIG. The new criterion (EHVIG) is utilized by two different strategies in this chapter. Firstly, it enables gradient ascent methods to be used in MOBGO. Moreover, since the EHVIG of an optimal solution should be a zero vector, it can be regarded as a stopping criterion in global optimization, e.g., an Evolution Strategy. Empirical experiments are performed on seven benchmark problems. The experimental results show that the second proposed strategy, using EHVIG as a stopping criterion, can outperform the normal MOBGO on the problems, where the optimal solutions are located in the interior of the search space. For the remaining test problems, EHVIG can still perform better when gradient projection is applied.

This chapter mainly discusses the computation of the 2-D EHVIG and how to apply EHVIG in MOBGO by two approaches: using EHVIG in gradient ascent algorithm for local search and using EHVIG as a stopping criterion in an evolutionary algorithm. The chapter is structured as follows: Section 6.1 describes the motivations of the EHVIG research; Section 6.2 introduces the definition of the EHVIG and proposes an efficient algorithm to calculate 2-D EHVIG, including a computational performance assessment between the proposed efficient exact calculation method and numerical calculation method in 2-D EHVIG case; Section 6.3 introduces the gradient method using EHVIG in MOBGO; Section 6.4 illustrates how to utilize EAs (CMA-ES in this chapter) assisted by the stopping criterion EHVIG in MOBGO; Section 6.5 shows the empirical experimental results.

6.1 Motivations

Compared to EAs, MOBGO still performs much slower using the infill criterion EHVI, because EHVI needs to be called many times in the process of searching for the optimal point based on the Kriging models. Since the calculation of the EHVI can be formulated in a closed form, it is possible to analyze its differentiability. It is easy to see, that all components of the EHVI expression are differentiable. However, a precise formula of the EHVIG has not been derived until now. Once the formula of EHVIG is derived, it could speed up the MOBGO in the process of searching for the optimal point by using the gradient ascent algorithm to maximize EHVI or using it as a stopping criterion in EAs.

6.2 Expected Hypervolume Improvement Gradient (EHVIG)

Considering the definition of the EHVI in Equation (2-1) and the efficient algorithm to calculate 2-D EHVI (minimization case) in A.4, the EHVI is differentiable with respect to the predictive mean and its corresponding standard deviation, which are again differentiable with respect to the input vector (or target point) in the search space. The EHVIG is the first order derivative of the EHVI with respect to a target point $\mathbf{x}$ under consideration in the search space. It is defined as:

Definition 6.1 (Expected Hypervolume Improvement Gradient) [1] *Given parameters of the multivariate predictive distribution* $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ *at a target point* $\mathbf{x}$ *in the search space, the Pareto-front approximation* $\mathcal{P}$, *and a reference point* $\mathbf{r}$, *the*

[1] The prediction of $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ depends on a Kriging model and a target point $\mathbf{x}$ in the search space. Explicitly, EHVIG is dependent on the target point $\mathbf{x}$.

expected hypervolume improvement gradient (EHVIG) at $\mathbf{x}$ *is defined as:*

$$EHVIG(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \mathcal{P}, \mathbf{r}) = \frac{\partial \left(\int_{\mathbb{R}^d} HVI(\mathcal{P}, \mathbf{y}) \cdot PDF_{\boldsymbol{\mu}, \boldsymbol{\sigma}}(\mathbf{y}) d\mathbf{y} \right)}{\partial \mathbf{x}} = \frac{\partial \left(EHVI(\boldsymbol{\mu}, \boldsymbol{\sigma}, \mathcal{P}, \mathbf{r}) \right)}{\partial \mathbf{x}} \tag{2-1}$$

According to the definition of EHVIG in Equation (2-1) and the efficient algorithm to calculate EHVI in Equation (A-12), we can substitute the Equation (A-12) into Equation (2-1), say that the formula of EHVIG for 2-D case can be expressed as:

$$EHVIG(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\sigma}, \mathcal{P}, \mathbf{r}) = \frac{\partial (EHVI(\boldsymbol{\mu}, \boldsymbol{\sigma}, \mathcal{P}, \mathbf{r}))}{\partial \mathbf{x}}$$

$$= \frac{\partial \big(\sum_{i=1}^{n+1} (y_1^{(i-1)} - y_1^{(i)}) \cdot \Phi(\frac{y_1^{(i)} - \mu_1}{\sigma_1}) \cdot \Psi(y_2^{(i)}, y_2^{(i)}, \mu_2, \sigma_2) \big)}{\partial \mathbf{x}} + \tag{2-2}$$

$$\frac{\partial \Big(\sum_{i=1}^{n+1} \big(\Psi(y_1^{(i-1)}, y_1^{(i-1)}, \mu_1, \sigma_1) - \Psi(y_1^{(i-1)}, y_1^{(i)}, \mu_1, \sigma_1) \big) \cdot \Psi(y_2^{(i)}, y_2^{(i)}, \mu_2, \sigma_2) \Big)}{\partial \mathbf{x}} \tag{2-3}$$

For the Terms (2-2) and (2-3), the prerequisite of calculating these two Terms is to calculate the gradient of the Ψ function and of the $\Phi(\frac{y-\mu}{\sigma})$ function. The final expressions for $\frac{\partial \Psi(a,b,\mu,\sigma)}{\partial \mathbf{x}}$ and $\frac{\partial \Phi(\frac{y-\mu}{\sigma})}{\partial \mathbf{x}}$ are shown in Equation (2-4) and Equation (2-5), respectively. For detailed proofs, please refer to the Appendix (A.3) in this dissertation.

$$\frac{\partial \Psi(a, b, \mu, \sigma)}{\partial \mathbf{x}} = \big(\frac{b-a}{\sigma} \cdot \phi(\frac{b-\mu}{\sigma}) - \Phi(\frac{b-\mu}{\sigma}) \big) \cdot \frac{\partial \mu}{\partial \mathbf{x}} + \phi(\frac{b-\mu}{\sigma}) \cdot \big(1 + \frac{(b-\mu)(b-a)}{\sigma^2} \big) \cdot \frac{\partial \sigma}{\partial \mathbf{x}} \tag{2-4}$$

$$\frac{\partial \Phi(\frac{y-\mu}{\sigma})}{\partial \mathbf{x}} = \phi(\frac{y-\mu}{\sigma}) \cdot (\frac{\mu - y}{\sigma^2} \cdot \frac{\partial \sigma}{\partial \mathbf{x}} - \frac{1}{\sigma} \cdot \frac{\partial \mu}{\partial \mathbf{x}}) \tag{2-5}$$

By substituting Equations (2-4) and (2-5) into Term (2-2) with applying the chain rule, Term (2-2) can be expressed by:

$$\frac{\partial\Big(\sum_{i=1}^{n+1}(y_1^{(i-1)}-y_1^{(i)})\cdot\Phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)\Big)}{\partial\mathbf{x}}$$

$$=\sum_{i=1}^{n+1}(y_1^{(i-1)}-y_1^{(i)})\cdot\frac{\partial\Big(\Phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)\Big)}{\partial\mathbf{x}}$$

$$=\sum_{i=1}^{n+1}(y_1^{(i-1)}-y_1^{(i)})\cdot$$
$$\Big(\phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\cdot(\frac{\mu_1-y_1^{(i)}}{\sigma_1^2}\cdot\frac{\partial\sigma_1}{\partial\mathbf{x}}-\frac{1}{\sigma_1}\cdot\frac{\partial\mu_1}{\partial\mathbf{x}})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)+$$
$$\big(0-\Phi(\frac{(y_2^{(i)}-\mu_2)}{\sigma_2}\cdot\frac{\partial\mu_2}{\partial\mathbf{x}})+\phi(\frac{y_2^{(i)}-\mu_2}{\sigma_2})\cdot(1+0)\cdot\frac{\partial\sigma_2}{\partial\mathbf{x}}\big)\cdot\Phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\Big)$$

$$=\sum_{i=1}^{n+1}(y_1^{(i-1)}-y_1^{(i)})\cdot$$
$$\Big(\phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\cdot(\frac{\mu_1-y_1^{(i)}}{\sigma_1^2}\cdot\frac{\partial\sigma_1}{\partial\mathbf{x}}-\frac{1}{\sigma_1}\cdot\frac{\partial\mu_1}{\partial\mathbf{x}})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)+$$
$$\big(\phi(\frac{y_2^{(i)}-\mu_2}{\sigma_2})\cdot\frac{\partial\sigma_2}{\partial\mathbf{x}}-\Phi(\frac{y_2^{(i)}-\mu_2}{\sigma_2})\cdot\frac{\partial\mu_2}{\partial\mathbf{x}}\big)\cdot\Phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\Big) \tag{2-6}$$

Similar to the derivation of Term (2-2), Term (2-3) can be expressed by:

$$\frac{\partial\Bigg(\sum_{i=1}^{n+1}\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)-\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)\Bigg)}{\partial\mathbf{x}}$$

$$=\sum_{i=1}^{n+1}\Bigg(\frac{\partial\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)-\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)}{\partial\mathbf{x}}\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)+$$
$$\frac{\partial\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)}{\partial\mathbf{x}}\cdot\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)-\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)\Bigg)$$

$$=\sum_{i=1}^{n+1}\Bigg(\frac{\partial\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)\big)}{\partial\mathbf{x}}\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)-$$
$$\frac{\partial\big(\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)}{\partial\mathbf{x}}\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)+$$
$$\frac{\partial\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)}{\partial\mathbf{x}}\cdot\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)-\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)\Bigg)$$

$$
\begin{aligned}
=&\sum_{i=1}^{n+1}\Big((\phi(\frac{y_1^{(i-1)}-\mu_1}{\sigma_1})\cdot\frac{\partial\sigma_1}{\partial\mathbf{x}}-\Phi(\frac{y_1^{(i-1)}-\mu_1}{\sigma_1})\cdot\frac{\partial\mu_1}{\partial\mathbf{x}})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)-\\
&([\frac{y_1^{(i)}-y_1^{(i-1)}}{\sigma_1}\cdot\phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})-\Phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})]\cdot\frac{\partial\mu_1}{\partial\mathbf{x}}+[\phi(\frac{y_1^{(i)}-\mu_1}{\sigma_1})\cdot\\
&(1+\frac{(y_1^{(i)}-\mu_1)(y_1^{(i)}-y_1^{(i-1)})}{\sigma_1^2})]\cdot\frac{\partial\sigma_1}{\partial\mathbf{x}})\cdot\Psi(y_2^{(i)},y_2^{(i)},\mu_2,\sigma_2)+\\
&(\phi(\frac{y_2^{(i)}-\mu_2}{\sigma_2})\cdot\frac{\partial\sigma_2}{\partial\mathbf{x}}-\Phi(\frac{y_2^{(i)}-\mu_2}{\sigma_2})\cdot\frac{\partial\mu_2}{\partial\mathbf{x}})\cdot\\
&\big(\Psi(y_1^{(i-1)},y_1^{(i-1)},\mu_1,\sigma_1)-\Psi(y_1^{(i-1)},y_1^{(i)},\mu_1,\sigma_1)\big)\Big)
\end{aligned}
\tag{2-7}
$$

Then, the EHVIG is the sum of Terms (2-6) and (2-7). In these two Terms, $\frac{\partial\mu_i}{\partial\mathbf{x}}$ and $\frac{\partial\sigma_i}{\partial\mathbf{x}}$ $(i = 1, 2)$ are the first order derivatives of the Kriging predictive means and standard deviations at a target point $\mathbf{x}$, respectively. These parameters can be precisely calculated by means of a Kriging model. For the details of the formulas and how to calculate these parameters, please refer to [103].

Performance Assessment The performance assessment of the EHVIG will be illustrated by a single numerical experiment. The bi-criteria optimization problem is: $y_1(\mathbf{x}) = ||\mathbf{x}-\mathbf{1}|| \to \min$, $y_2(\mathbf{x}) = ||\mathbf{x}+\mathbf{1}|| \to \min$, $\mathbf{x} \in [-1, 6]\times[-1, 6] \subset \mathbb{R}^2$ [2]. Figure 6.1 shows the landscape of EHVIG, in which the evaluated points are marked by blue circles. The EHVIG calculated by the exact method is indicated by the black arrow in the left figure. The EHVIG calculated by the numerical method is indicated by the red arrows in the right figure. The landscapes of EHVIG in both figures are very similar, however, there exist some slight differences between them, while very small and caused by numerical errors.

6.3 Gradient Ascent Algorithm

Previously, the optimizer *opt* in Algorithm 3 was chosen as CMA-ES [104], which is a state-of-the-art heuristic global optimization algorithm. Since the formula of 2-D EHVIG is derived in this chapter, a gradient ascent algorithm can replace CMA-ES to speed up the process of finding an optimal point x^*.

Many gradient ascent algorithms (GAAs) exist. The conjugate gradient algorithm is one of the most efficient algorithms among them. However, it cannot solve the

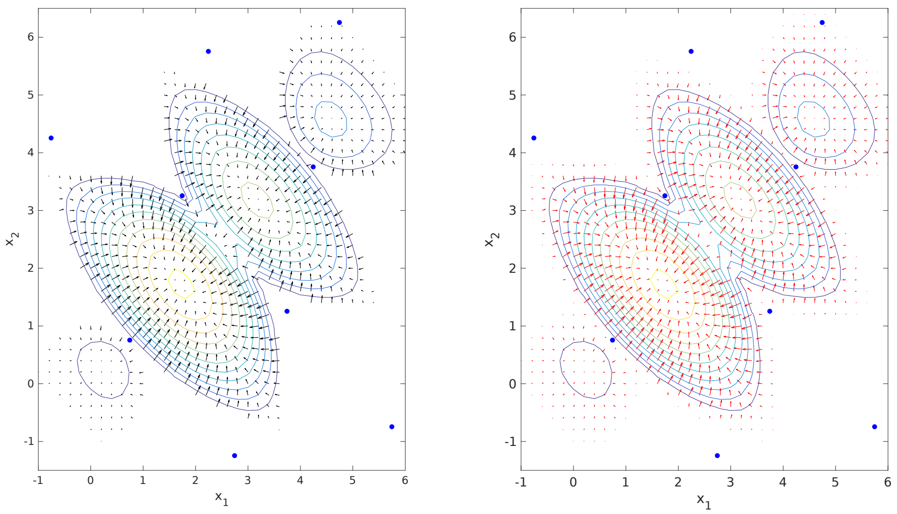

Figure 6.1: The landscape of EHVIG. Left: computed using exact calculation algorithm, Right: computed using numerical calculation method.

constrained problems, and this is the reason why we exclude it in this chapter. For the other GAAs, the general formula of computing the next solution is:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + s \cdot \nabla F(\mathbf{x}^{(t)}) \tag{3-8}$$

where $\mathbf{x}^{(t)}$ is the current solution, $\mathbf{x}^{(t+1)}$ is the updated solution, s is the stepsize, and $\nabla F(\cdot)$ is the gradient of the objective functions or of the infill criterion. In this chapter, ∇F is EHVIG.

Another important aspect is that the starting point is very important to the performance of GAAs. In order to improve the probability of finding the globally optimal point, CMA-ES was used to initialize the starting points in this chapter. The structure of gradient ascent based search algorithm is shown in Algorithm 10.

6.4 Stopping Criterion – EHVIG

Traditionally, when EAs are searching for the optimal point $\mathbf{x}^*$, convergence velocity and some other statistical criteria are used to determine whether the EAs should stop/restart or not. These criteria can balance the quality of the performance and efficiency of the execution time to some degree, but not optimally.

Algorithm 10: Gradient Ascent Based Search Algorithm

Input: Kriging Models $M_1, \cdots, M_d$, Pareto-front approximation $\mathcal{P}$, reference point $\mathbf{r}$, number of clusters N_c

Output: Optimal solution $\mathbf{x}^*$

1: Initialize λ points using CMA-ES with 15 iterations;
2: Cluster λ points into N_c clusters $G_1, \cdots, G_{N_c}$;
3: **for** $i = 1$ *to* $i <= N_c$ **do**
4: Update starting point $\mathbf{x}^s$, $\mathbf{x}^s = mean(G_i)$;
5: Calculate the optimal point $\mathbf{x}^{*i}$ using simple gradient ascent algorithm and the starting point $\mathbf{x}^s$;
6: Calculate the corresponding EHVI value $EHVI^i$
7: Find the optimal point $\mathbf{x}^*$ among $\mathbf{x}^{*1}, \cdots, \mathbf{x}^{*N_c}$;
8: Return $\mathbf{x}^*$;

Because all these criteria are blind to whether an individual is already the optimal or not.

Considering the gradient of the optimal point in the search space should be a zero vector and EHVIG can be exactly calculated, EHVIG can be used as a stopping/restart criterion in EAs when they are searching for the optimal point with the EHVI as the infill criterion. Theoretically speaking, the EHVI should be maximized during the procedure, therefore, this strategy should also be required to check the negative value of the second derivative of EHVI at this point. However, this is omitted due to the complexities. The structure of CMA-ES assisted by EHVIG is shown in Algorithm 11.

6.5 Experimental Results

Experimental Settings The benchmarks were well-known test problems: BK1 [78], SSFYY1 [79], ZDT1, ZDT2, ZDT3 [80] and the generalized Schaffer problem [81] with different parameter settings for γ. All these benchmarks were employed by using different searching strategies in MOBGO, as shown in Table 6.1. Each trail was repeated for 10 times. All the experiments were finished on the same computer: Intel(R) i7-3770 CPU @ 3.40GHz, RAM 16GB. The operating system was Ubuntu 16.04 LTS (64 bit) and platform was MATLAB 8.4.0.150421

(R2014b), 64 bit.

Algorithm 11: CMA-ES Assisted by EHVIG

Input: Kriging Models $M_1, \cdots, M_d$, Pareto-front approximation $\mathcal{P}$, reference point $\mathbf{r}$, restart number N_r

Output: Optimal solution $\mathbf{x}^*$

```
for i = 1 to i <= N_r do
    Initialize parameters in CMA-ES;
    flag = 1 ;
    for flag ≥ ε do
        Get offspring by normal CMA-ES with default parameters ;
        Select the best individual x*i from the offspring ;
        Predict the mean value μ* and the standard deviation σ* at x*i ;
        flag = Sum(|EHVIG(x*i, μ*, σ*, P, r)|) ;
Find the optimal point x* among x*1, ··· , x*N_r ;
Return x*;
```

Table 6.1: CMA-ES Parameter Settings.

	ϵ	N_r	Stopping Criterion	Max Iter.	Gradient Decent	N_c
Alg. 1	/	3	Default	2000	No	/
Alg. 2	10^{-5}	3	EHVIG	2000	No	/
Alg. 3	/	0	Default	15	Yes	4
Alg. 4	/	0	Default	15	No	/
Alg. 5	10^{-5}	3	EHVIG projection	2000	No	/

Table 6.2 shows the final experimental results. The final performance on each algorithm is evaluated by HV and execution time. The highest value of HV on each test problem is indicated in bold, and the smallest value of the standard deviation of HV is also shown in bold. For the execution time, both the least execution time and smallest standard deviation of time, among Alg. 1, Alg. 2 and Alg. 3 are indicated in bold.

Here, Alg. 4 (original CMA-ES with no restart mechanism and with a max iteration of 15) is a control group for Alg. 3 to test whether the GAA works as predicted or not. Since there is no new mechanism added to Alg. 4 and max iteration is too small, the performance of Alg. 4 is indeed worse than the other three algorithms. Hence, there is no need to compare the execution time of Alg. 4 with others.

From Table 6.2, it can be seen that Alg. 3, using GAA for searching an optimal point and CMA-ES for the initialization of the starting points, can improve the final performance a little bit, compared to Alg. 4. However, it can not outperform the original CMA-ES (Alg. 1). The reason is related to the starting points in the GAA, that is: GAA is very sensitive to the starting point and the starting points generated by CMA-ES with 15 iterations are located at the local optimal area.

Compared to original CMA-ES (Alg. 1), Alg. 2 (CMA-ES using EHVIG as the stopping criterion) outperforms Alg. 1 on BK1, SSFYY1, GSP, and GSP12. Among these four test problems, the execution time of Alg. 2 is much faster than Alg. 1 in the cases of the SSFYY1 and GSP problems. When applying EHVIG as a stopping criterion in Alg. 2, algorithm CMA-ES can terminate the loop earlier when the EHVIG of one individual is a zero vector, and therefore some execution time can be saved. In other words, while original CMA-ES does not know whether a current individual is already the optimal solution or not, EHVIG can be used as a criterion to check this individual. For the BK1 and GSP12 problems, Alg. 2 needs more time, but the performance of Alg. 2 is better than Alg. 1.

On ZDT series problems, however, the performance of Alg. 2 is worse than Alg. 1. An explanation of this phenomenon is that the optimal solutions for ZDT series problems are located on the boundary of the search space. According to the definition of the gradient, EHVIG would be infeasible at these boundaries, and thus EHVIG would mislead CMA-ES to search the optimal solution. A remedy to improve the performance of Alg. 2 is applying the projection of EHVIG to check whether an individual is optimal or not on the boundaries, instead of EHVIG. Here, the projection of EHVIG is the orthogonal projection of EHVIG onto the active constraint boundary. Since we are only dealing with box constraints, all the components of the gradient that correspond to active boundaries in the same dimension are set to zero. In Table 6.2, compared to Alg. 2 in ZDT series problems, Alg. 5 is assisted by the projection of EHVIG and can reach Pareto front approximations closer to the true ones with less execution time. For ZDT1 and ZDT2 problems, the average HV values of Alg. 5 are even better than Alg. 1 with less execution time.

Table 6.2: Experimental Results.

Benchmark	Ref.			Alg. 1	Alg. 2	Alg. 3	Alg. 4	Alg. 5
BK1	(60,60)	Time(mins)	mean	**6.2817**	13.4433	8.0933	0.4350	/
			std.	**0.6480**	1.0280	0.8803	0.0166	/
		HV	mean	3175.7582	**3175.9683**	3166.4668	3133.8960	/
			std.	0.3620	**0.2940**	3.6840	6.0266	/
SSFYY1	(5,5)	Time(mins)	mean	13.1067	**4.7667**	7.2550	0.4233	/
			std.	5.4001	**0.3306**	0.3705	0.0117	/
		HV	mean	20.7096	**20.7098**	20.5474	20.0187	/
			std.	0.0069	**0.0035**	0.0361	0.1284	/
ZDT1	(11,11)	Time(mins)	mean	82.9317	76.9400	**15.0667**	6.6133	34.8383
			std.	38.5988	12.1167	**8.2437**	4.2966	14.7293
		HV	mean	120.6491	120.6488	120.6268	120.6275	**120.6498**
			std.	0.0055	**0.0052**	0.0069	0.0066	0.0063
ZDT2	(11,11)	Time(mins)	mean	40.3233	39.6800	**6.8889**	2.0983	33.8407
			std.	7.1394	6.1038	**0.1332**	0.1628	2.3391
		HV	mean	120.3025	120.2965	120.1151	119.2155	**120.3159**
			std.	0.0130	**0.0067**	0.3474	2.9890	0.0127
ZDT3	(11,11)	Time(mins)	mean	53.6267	45.9850	**8.5450**	2.8550	13.3067
			std.	8.5955	8.8638	**0.5120**	0.4217	9.0423
		HV	mean	**128.7486**	128.4772	127.7556	127.4168	128.6857
			std.	**0.0079**	0.7747	1.2385	1.2383	0.1029
GSP	(5,5)	Time(mins)	mean	46.4850	**7.5017**	13.3167	0.5283	/
			std.	40.2517	**0.3572**	0.7771	0.0112	/
		HV	mean	24.9066	**24.9066**	24.9055	24.9050	/
			std.	0.0001	**0.0000**	0.0001	0.0001	/
GSP12	(5,5)	Time(mins)	mean	20.3167	20.6650	**13.7200**	4.6867	/
			std.	**0.4215**	0.7123	0.4407	0.1403	/
		HV	mean	24.3914	**24.3930**	24.3883	24.3848	/
			std.	0.0034	0.0019	**0.0016**	0.0013	/

6.6 Summary

This chapter introduced an efficient algorithm to exactly calculate the 2-D EHVIG and applied EHVIG in *Multi-Objective Bayesian Global Optimization* using two different strategies in the process of searching for the optimal solution: using EHVIG as a stopping criterion in the original CMA-ES and using gradient ascent algorithm (CMA-ES used here to initialize the starting points).

The empirical experimental results show that the gradient ascent based algorithm is much faster than original CMA-ES, but it has an obvious drawback: it gets easily stuck at a locally optimal point. Another strategy, taking EHVIG as the stopping criterion in CMA-ES, can improve the quality of the final Pareto front and reduce some execution time, compared to original CMA-ES on the problem whose optimal points are not at the boundaries in the search space. This strategy does not work on ZDT series problems because EHVIG cannot be calculated at the boundaries in the search space. However, a useful remedy to this strategy is the projection of EHVIG.