Part III

Dynamic Vector Evaluated Particle Swarm Optimisation

Chapter 9

Introduction to Dynamic Vector Evaluated Particle Swarm Optimisation Algorithm

''Goals allow you to control the direction of change in your favor."
–Brian Tracy

This chapter discusses the VEPSO algorithm that has been adapted to solve DMOOPs. The adapted VEPSO algorithm, dynamic VEPSO (DVEPSO), is discussed in Section 9.1. Section 9.2 discusses the tasks of the DVEPSO algorithm that are performed at the top-algorithm level, while Section 9.3 discusses the tasks of the sub-swarms that are performed at the lower-algorithm level. Experiments that were conducted to investigate the influence of various guide update approaches on the performance of DVEPSO are discussed in Section 9.4. Information is provided with regards to the benchmark functions, performance measures and the default configuration of the DVEPSO algorithm used for the experiments, as well as the statistical analysis that was conducted on the obtained data. Furthermore, the obtained results are analysed and discussed. A summary of this chapter is provided in Section 9.5.

9.1 Dynamic Vector Evaluated Particle Swarm Optimisation Algorithm

This section discusses the changes made to the SMOO VEPSO algorithm discussed in Section 7.2 in order to solve DMOOPs. The adapted algorithm, DVEPSO, is presented in Algorithm 9.

Algorithm 9 DVEPSO for DMOO

```
for number of iterations do
    check whether a change has occurred
    if change has occurred
        respond to change
        remove dominated solutions from archive
    perform PSO iteration
    if new solutions are non-dominated
        if space in archive
            add new solutions to archive
    else
        remove solutions from archive
        add new solutions to archive
    select sentry particles
```

Similar to VEPSO, the DVEPSO algorithm consists of two layers, namely a top layer that manages the sub-swarms and a lower layer that contains the sub-swarms. This is illustrated in Figure 9.1.

In order to track a changing POF an algorithm must be able to detect that a change in the environment has occurred and then respond to the change appropriately. Therefore, when solving DMOOPs, the sub-swarms in the lower layer check whether the environment has changed, in addition to optimising the assigned objective function. When VEPSO is used to solve static MOOPs, sharing of knowledge between the sub-swarms and the management of the archive (as discussed in Section 7.2) are managed at the top level. However, the top layer of DVEPSO also manages the way in which the sub-swarms

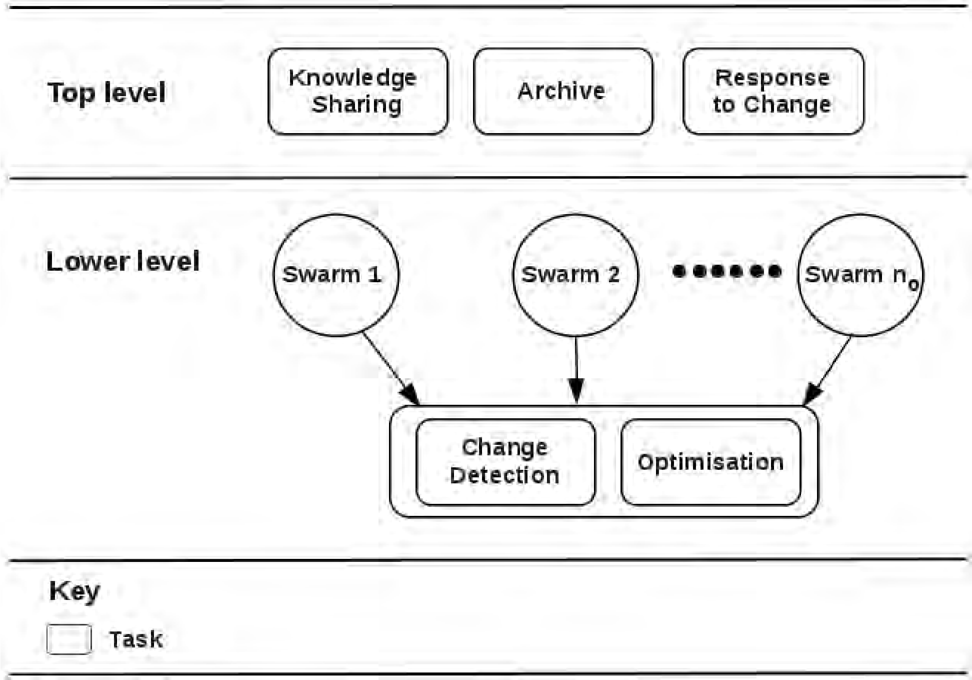


Figure 9.1: The two layers of the DVEPSO algorithm

respond to a change once the change has been detected.

9.2 Top-level Tasks

This section discusses a task that is performed at the top level of the DVEPSO algorithm, namely responding to a change in the environment. This task is performed in addition to the top-level tasks performed by VEPSO (refer to Section 7.2).

If a change has been detected by one or more of the sub-swarms, DVEPSO has to respond to the change to ensure tracking of the changing POF. When a change has been detected, one of the following responses are used:

- re-evaluate all particles in the sub-swarm, or
- re-initialise a percentage of the particles in the sub-swarm.

Re-evaluating the particles ensures that all previously obtained information is preserved. However, the particles already converged towards the POF, and therefore the diversity of the swarm has to be increased to increase exploration of a new environment. If re-evaluation is used, additional ways should be used to increase the swarm's diversity. However, re-initialisation introduces diversity by re-initialising a certain percentage of the swarm's particles. Re-initialisation preserves previously obtained information from the particles that are not re-initialised. However, it may occur that particles with optimal positions in the new environment are re-initialised and thereby the information is lost.

Greeff and Engelbrecht [72] proposed that the above listed responses can be applied to either all sub-swarms, or to only the sub-swarm(s) whose objective function has changed. Applying the response to all sub-swarms increases the diversity of all sub-swarms and thereby increases the exploration of the sub-swarms. If a sub-swarm's objective function did not change and re-initialisation is used, a percentage of previously obtained information is removed. However, the increasing diversity may lead to exploration of the search space that was not explored before.

After one of the above responses was applied, the following re-evaluations or updates are performed:

- The pbest of each particle is reset to the particle's current position. This ensures that the particle is not biased towards the previous optima. If the new optima is far away from the previous optima and the particle is biased towards the previous optima, it may become stuck at the previous optima or a local optima without finding the new optima.
- Once the particles' pbests are reset, a new gbest is determined. This ensures that the gbest does not attract the other particles towards a previous optimum that is not optimal anymore.

Furthermore, if a change in the environment occurs, the following approaches are proposed to manage the archive [78]:

- remove all solutions from the archive (referred to as ac), or
- re-evaluate the solutions in the archive against the current DMOOP. Then, all solutions that were previously non-dominated but became dominated after the change in the environment occurred, are:
  - removed from the archive (referred to as a_{re}). This approach does not use previously obtained knowledge in the new environment. When an environ-

ment change is severe, previously found solutions that are still non-dominated in the new environment, may cause new non-dominated solutions that are in close proximity of the previous solutions to be removed from the archive, even if they are more optimal than the previously found solution. This may occur when selecting which solutions to remove from a full archive are based on removing solutions from crowded regions of the approximated POF.

  - hill-climbing is applied to a dominated solution in an attempt to change these solutions back to non-dominated solutions. If hill-climbing is unsuccessful, the dominated solution is removed from the archive. However, if hill-climbing is successful, the dominated solution is removed from the archive and the new solution obtained through hill-climbing is added to the archive. This approach is referred to as a_{reh}. This approach re-uses previously obtained knowledge in the new environment and will only be useful if the environmental change is not severe.

- when a change in the environment occurs, a number of particles whose positions represent non-dominated solutions are randomly selected. The average change that the selected particles experience in each objective (or dimension), c_{avg_k}, is calculated. Then, if a selected particle's objective value differs by a threshold β_k (e.g. $\beta_k = c_{avg_k}/2.0$), the solutions in the archive that are within a specied radius c_r (e.g. distance to closest selected particle/2.0) from the selected particle, are deleted. This approach is referred to as a_r. If a_r is used to manage the archive, then before a_r is executed, either a_{re} or a_{reh} is performed. If a_{re} was fist performed, this approach is referred to as a_{ra}. Otherwise, if a_{reh} was first performed, this approach is referred to as a_{rah}. Applying a_r to the archive removes solutions from a certain region of the archive (that falls within the radius c_r of a selected particle) if the environment changes drastically for the decision variable values that produced the solutions of the specific region. This ensures that newly found solutions are added to the archive when the environment changes drastically, increasing the diversity of the archive.

9.3 Low-level Tasks

This section discusses the tasks of change detection and guide updates that are performed at the lower-level of DVEPSO by the sub-swarms. These tasks are performed in addition to the other low-level tasks performed by VEPSO (refer to Section 7.2).

9.3.1 Change Detection

In order to solve DMOOPs, DVEPSO must be able to detect a change that occurred in the environment. Change detection is done using sentry particles [22], where a specified number of particles are randomly selected and re-evaluated after the algorithm performed the specific iteration, but before the next iteration starts. If the sentry particle's fitness value differs after re-evaluation with more than a specified value, the swarm is notified that a change in the environment has occurred. If a change in the environment of a sub-swarm has occurred, the sub-swarm alerts the top-level of DVEPSO. The top-level then informs the sub-swarms which response to execute.

9.3.2 Guide Update Approaches

Similar to VEPSO, the search process of DVEPSO is driven through the local and global guides. VEPSO uses no Pareto-dominance information for the guide updates. However, for DVEPSO, guide update approaches that use Pareto-dominance information and therefore do dominance checking are also investigated. The following guide update approaches are proposed for DVEPSO [71]:

- The standard VEPSO guide update, where the particle's fitness is measured with regards to only the objective function that the specific swarm optimises. Only if an improvement in the fitness of the current guide can be obtained, is the guide updated. No Pareto-dominance information is used. With reference to a local guide, this approach is referred to as p_s and with reference to a global guide, g_s.
- The dominant approach, where each particle's fitness is measured with respect to all objectives of the DMOOP. If the particle's position dominates the current local guide, the particle's current position is selected as the new local guide. This

strategy is referred to as p_d. If this approach is used to update a global guide, it is referred to as g_d.

- The non-dominated approach, where a guide is updated if the new position is non-dominated with respect to the guide. When used as a local guide update, it is referred to as p_n, and g_n if used as a global guide update.
- The random approach, where a guide is updated if the new position is non-dominated with respect to the guide, by randomly selecting either the particle position or the corresponding guide. When used as a local guide update, it is referred to as p_r and g_r if used as a global guide update.

The effectiveness of these approaches when used by DVEPSO is unknown. Therefore, experiments were conducted to investigate the influence of these guide update approaches on the performance of DVEPSO. The next section discusses the experiments and the results that were obtained from the experiments.

9.4 Effectiveness of Guide Update Approaches

Various guide update approaches exist as discussed in Section 9.3. This section describes experiments that were conducted to investigate the influence of the various guide update approaches on the performance of DVEPSO. It should be noted that this section focuses on guide update approaches, and not on guide selection approaches. Guide selection approaches focus on the selection of solutions from the archive to guide the optimisation process to ensure a diverse set of solutions. The guides that are selected from the archive are then used as the local (personal best) and global guides (global best) of the PSO algorithm. The guide update approaches discussed in this section focus on methods that are used to update the swarm's local (personal best) and global (global best) guides using the solutions found by the particles.

Section 9.4.1 discusses the experimental setup and the benchmark functions and performance measures that were used to evaluate the performance of the various guide update approaches. The DVEPSO configuration used for the experiments, as well as the statistical analysis process that was performed on the obtained data, are also discussed. The results obtained from the experiments are discussed in Section 9.4.2.

9.4.1 Experimental Setup

All combinations of the local and global guide updates discussed in Section 9.3 were used in the experiments.

All experiments consisted of 30 independent runs and each run continued for 1000 iterations. For all benchmark functions, the severity of change (n_t) was set to 1, 10 and 20 and the frequency of change (τ_t) was set to either 10, 25 or 50. This selection of n_t and τ_t values enables the evaluation of DVEPSO in both a fast and slowly changing environment, and an evironment that changes either gradually or severely over time.

The PSO parameters were set to values that lead to convergent behaviour [63], namely $w = 0.72$ and $c_1 = c_2 = 1.49$. Convergent behaviour ensures that the particles converge towards the current POF. After a change in the environment, diversity is introduced into the swarm to ensure more exploration to find the new POF.

All code was implemented in the Computational Intelligence library (CIlib) [122]. All simulations were run on the Sun Hybrid System's Harpertown and Nehalem Systems of the Center for High Performance Computing [24]. The SUN Nehalem system has an Intel Nehalem processor of 2.93 GHz, 2304 CPU cores, 3465 Gb of Memory and produces 24 TFlops at peak performance [24]. The SUN Harpertown system has an Intel Xeon processor of 3.0 GHz, 384 CPU cores, 768 Gb of Memory and produces 3 TFlops at peak performance [24].

Benchmark Functions

Based on the analysis of DMOOPs in Chapter 3, fifteen benchmark functions were selected of various DMOOP Types to study the influence of guide update approaches on the performance of DVEPSO, namely a modified version of DIMP2 with a concave POF (referred to as DIMP2 in the rest of the thesis), FDA1$_{Zhou}$, FDA2, FDA2$_{Camara}$, FDA3 [58], FDA3$_{Camara}$, dMOP2, dMOP3, dMOP2$_{iso}$, dMOP2$_{dec}$, HE1, HE2, HE6, HE7 and HE9.

DIMP2 is a Type I problem where each decision variable has its own rate of change, except the variable x_1 that controls the spread of solutions. FDA1$_{Zhou}$ has non-linear dependencies between the decision variables and is a Type II problem. FDA2 and dMOP2 are Type II DMOOPs with a POF that changes from convex to concave. FDA2$_{Camara}$

also has a POF that changes from convex to concave over time, but is a Type III DMOOP. FDA3 and FDA3$_{Camara}$ are Type II DMOOPs with a convex POF where the density of solutions in the POF changes over time. dMOP3 is a Type I DMOOP with a convex POF where the spread of the POF solutions changes over time. HE1 and HE2 are both a Type III DMOOP with a discontinuous POF that consists of various disconnected continuous sub-regions. HE6, HE7 and HE9 are Type III DMOOPs where each decision variable has a different POS and the POSs are non-linear functions. dMOP2$_{iso}$ and dMOP2$_{dec}$ are similar to dMOP2, but with an isolated and deceptive POF respectively.

Even though the DMOOPs FDA2 and FDA3 are problematic (refer to Section 3.2.1), they were selected for the experiments to determine whether DVEPSO can still track the changing POF in spite of the issues with these DMOO functions.

Performance Measures

Chapter 4 discussed the analysis of DMOO performance measures. Based on this analysis, three performance measures were selected for this study, to determine the performance of DVEPSO for the different guide update approaches.

The first performance measure is the number of non-dominated solutions (NS) in the found POF. Even though this measure does not provide any information with regards to the quality of the solutions, it provides additional information when comparing the performance of various algorithms.

The second performance measure is the acc_{alt} measure (see Equation (4.25)), referred to in this chapter as acc. A low acc value indicates a good performance. The calculation of acc requires sampled solutions of the true POF, POF'. For these experiments, POF' solutions were created for each DMOOP by dividing the range of each variable into one thousand equally sized intervals. For each combination of decision variable values the objective function values were calculated using the equation of the true POF, POF, for the specific DMOOP. This process was followed for each n_t-τ_t combination. The HV was calculated according to [7], using the source code available at [61].

The effect of the changes in the environment on acc of the algorithm is quantified by the third measure, namely $stab$ (refer to Equation (4.21)), where a low $stab$ value indicates good performance.

Algorithm Configuration

The following default configuration of DVEPSO was used for the experiments:

- Each swarm has 20 particles and a random swarm topology is used.
- The non-dominated solutions found so far is stored in an archive with size set to 100. If the archive is full, a solution from a crowded region in the found POF is removed. The croweded region is determined by calculating the distance between each solution in the archive and its nearest solution in the archive, and selecting the solution(s) with the smallest distance value.
- Sentry particles is used for change detection (refer to lines 2 and 13 in Algorithm 9). If a change has been detected, 30% of the particles of the swarm(s) whose objective function changed is re-initialised (refer to line 4 in Algorithm 9). The non-dominated solutions in the archive are re-evaluated and the solutions that have become dominated are removed from the archive (refer to line 5 in Algorithm 9). Each particle's pbest is set to its current position and a new gbest is determined.

Statistical Analysis

This section discusses the statistical analysis procedure performed on the obtained data. For each function and for each n_t-τ_t combination, a Kruskal-Wallis test was performed over the obtained data to determine whether there is a statistical significant difference in performance. For each performance measure the obtained data is the mean of the performance measure values for each iteration just before a change occurred in the environment over 30 runs. If this test indicated that there was a difference, pairwise Mann-Whitney U tests were performed between the pairs of obtained data for all the guide update approaches.

For each pair of guide update approaches, if the pairwise Mann-Whitney U test indicated a statistically significant difference, a win was recorded for the winning algorithm and a loss for the losing algorithm.

All statistical tests were performed for a confidence level of 95%. The null hypothesis was that there is no statistical significant difference between the performance of the various guide update approaches. The alternative hypothesis was that there is a difference in mean performance.

9.4.2 Results

This section presents the results obtained by the various guide update approaches. The results are discussed considering the various n_t-τ_t combinations, with regards to three performance measures and with regards to DMOOP Types I to III. General observations are also highlighted. Tables 9.1 to 9.13 present the wins and losses. Only the tables highlighting interesting trends are discussed and therefore presented in this section. The other wins and losses tables are presented in Appendix D. Only statistical significant values are included in the tables. The p-values obtained for the various Mann-Whitney U tests, as well as the average performance measure values, are presented in Appendix D.

Results with regards to Performance Measures

Table 9.1 presents the wins and losses for each performance measure calculated over all DMOOPs and all n_t-τ_t combinations.

Table 9.1: Overall Wins and Losses for Various Performance Measures

PM	**Results**	**pbest-gbest combination**															
		s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
acc	Wins	222	219	229	287	150	146	164	149	165	161	156	162	171	138	144	188
acc	Losses	347	345	313	225	169	165	109	128	149	120	123	123	152	151	137	95
acc	Diff	-125	-126	-84	62	-19	-19	55	21	16	41	33	39	19	-13	7	93
acc	Rank	15	16	14	2	12	12	3	7	9	4	6	5	8	11	10	**1**
stab	Wins	297	246	300	267	61	50	39	62	72	32	59	28	64	35	60	34
stab	Losses	111	110	69	96	88	132	126	113	84	113	102	126	84	136	89	127
stab	Diff	186	136	231	171	-27	-82	-87	-51	-12	-81	-43	-98	-20	-101	-29	-93
stab	Rank	2	4	**1**	3	7	12	13	10	5	11	9	15	6	16	8	14
NS	Wins	267	396	348	449	67	226	243	132	81	236	141	241	62	233	135	251
NS	Losses	303	205	212	171	384	148	131	190	380	133	165	158	377	141	182	141
NS	Diff	-36	191	136	278	-317	78	112	-58	-299	103	-24	83	-315	92	-47	110
NS	Rank	11	2	3	**1**	16	9	4	13	14	6	10	8	15	7	12	5

With regards to *acc*, the following observations are made:

- The best and second best performance were obtained by p_d-g_r and p_s-g_r respectively.
- All p_s combinations, except p_s-g_r, performed poorly and p_s-g_n obtained the worst

rank. With regards to the g_s combinations, p_s-g_s and p_n-g_s performed poorly. However, p_d-g_s and p_r-g_s performed reasonably well.

- For the p_n combinations, p_n-g_d and p_n-g_r performed well, but p_n-g_s and p_n-g_n performed poorly. All g_n combinations performed poorly, except p_r-g_n that performed well.
- All the p_d combinations performed average, except p_d-g_r that obtained the best performance. For the g_d combinations, p_n-g_d and p_r-g_d performed well. However, p_s-g_d and p_d-g_d performed badly.
- All p_r combinations performed reasonably well and all g_r combinations performed really well.
- With the exception of p_r-g_r, using the same update approach for both pbest and gbest lead to a poor performance.

The following observations are made with regards to *stab*:

- The best performance was obtained by p_s-g_d and the worst by p_d-g_n.
- In contrast to their performance with regards to *acc*, all p_s combinations performed really well with regards to *stab*. Furthermore, all g_s combinations performed well.
- Except p_n-g_s that performed well, all p_n combinations performed average or poorly. The g_n combinations obtained a mixed performance with regards to *stab*. A good performance was obtained by p_s-g_n, an average performance by p_r-g_n and a poor performance by p_n-g_n and p_d-g_n.
- For the p_d combinations, p_d-g_s and p_d-g_d performed well. However, p_d-g_n and p_d-g_r performed really bad. All g_d combinations performed well, except p_n-g_d that performed poorly.
- In contrast with the p_r combinations' performance with regards to *acc*, p_r-g_s performed well with regards to *stab*, p_r-g_n and p_r-g_d performed average and p_r-g_r performed poorly. All g_r combinations performed rather poorly, except p_s-g_r that performed well.
- Using the same update approach for both pbest and gbest produced a really good performance for p_s-g_s, an average performance for p_d-g_d and a poor performance for p_n-g_n and p_r-g_r.

With regards to NS, the following observations are made:

- The best performance was obtained by p_s-g_r and p_n-g_s performed the worst.
- All p_s combinations produced good results, except p_s-g_s. However, all g_s combinations performed poorly.
- Two p_n combinations performed poorly, namely p_n-g_s and p_n-g_r. However, p_n-g_r performed well and p_n-g_n performed average. All g_n combinations performed average, except p_s-g_n that performed well.
- For the p_d combinations, p_d-g_n and p_d-g_r performed well, while p_d-g_s and p_d-g_d performed poorly. Mixed results were also obtained by g_d. Good performance was achieved with p_s-g_d and p_n-g_d. However, average and poor performance were obtained by p_r-g_d and p_d-g_d respectively.
- All p_r combinations performed average, except p_r-g_s that performed poorly. All g_r combinations performed well or average, except p_n-g_r that performed badly.
- Using the same update approach for both pbest and gbest lead to either average (p_n-g_n and p_r-g_r) or poor performance (p_s-g_s and p_d-g_d).

The guide update approach of the original VEPSO algorithm, p_s-g_s, obtained the second lowest rank with regards to *acc*, the second best rank with regards to *stab* and the eleventh rank with regards to NS. Therefore, the guide update approaches that use Pareto-dominance information outperformed this approach with regards to all performance measures.

Another approach to measure the performance of a DMOO algorithm, is to analyse the performance of the algorithm in various types of environments, such as a fast or slow changing environment and a gradually or severely changing environment. Therefore, the next section discusses the overall performance of the guide update approaches, measured over all performance measures and all n_t-τ_t combinations.

Results with regards to Various Frequencies and Severities of Change

The wins and losses calculated over all performance measures and DMOOPs for the various n_t-τ_t combinations are presented in Table 9.2.

For a fast changing environment ($n_t = 10$ and $\tau_t = 10$) the following observations are made:

- The best performance was obtained by p_s-g_r and p_n-g_s performed the worst.
- All p_s combinations performed really well. Two g_s combinations performed well, namely p_s-g_s and p_r-g_s. The other two g_s combinations performed poorly.
- A poor performance was obtained by all p_n combinations, except p_n-g_d that performed well. In contrast, a good performance was obtained by all g_n combinations, except p_n-g_n.
- All p_r combinations performed average, except p_r-g_s that performed poorly. However, all g_r combinations performed well, except p_n-g_r.
- An good or average performance was obtained by all p_d combinations, except p_d-g_s that obtained a poor performance. In addition, a good or average performance was obtained by all g_d combinations.

Table 9.2: Overall Wins and Losses for Various Frequencies and Severities of Change

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**ResultsResults**	**pbest-gbest combination**															
			s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	25	Diff	81	123	145	98	-102	-14	-17	-20	-86	-22	-5	-17	-98	-34	-25	-7
10	25	Rank	4	2	**1**	3	16	7	8	10	14	11	5	8	15	13	12	6
10	50	Wins	126	172	151	219	55	124	136	72	68	133	72	130	58	116	75	148
10	50	Losses	174	141	135	127	159	92	71	86	148	75	74	87	156	97	81	65
10	50	Diff	-48	31	16	92	-104	32	65	-14	-80	58	-2	43	-98	19	-6	83
10	50	Rank	13	7	9	**1**	16	6	3	12	14	4	10	5	15	8	11	2
1	10	Wins	174	169	180	200	82	85	101	76	100	87	85	80	89	96	75	95
1	10	Losses	197	177	176	106	117	105	71	109	116	71	83	82	100	93	96	75
1	10	Diff	-23	-8	4	94	-35	-20	30	-33	-16	16	2	-2	-11	3	-21	20
1	10	Rank	14	9	5	**1**	16	12	2	15	11	4	7	8	10	6	13	3
20	10	Wins	125	144	144	177	54	81	67	62	57	73	61	70	50	68	61	65
20	10	Losses	142	100	106	76	101	65	74	67	98	59	81	71	105	76	69	69
20	10	Diff	-17	44	38	101	-47	16	-7	-5	-41	14	-20	-1	-55	-8	-8	-4
20	10	Rank	12	2	3	**1**	15	4	9	8	14	5	13	6	16	10	10	7

The following observations are made for a slower changing environment, i.e. with $\tau_t = 25$ and $\tau_t = 50$:

- The best performance for $\tau_t = 25$ and $\tau_t = 50$ were obtained by p_s-g_d and p_s-g_r respectively. For both $\tau_t = 25$ and $\tau_t = 50$, the worst performance was obtained by p_n-g_s.
- All p_s combinations performed poorly, except p_s-g_s that performed well for $\tau_t = 25$. For both $\tau_t = 25$ and $\tau_t = 50$, all g_s combinations performed poorly, except p_s-g_s

that performed well for $\tau_t = 25$.

- For $\tau_t = 25$ all p_n combinations performed average, except p_n-g_s that performed poorly. However, with $\tau_t = 50$, p_n-g_n and p_n-g_d performed well, while the other two p_n combinations performed poorly. For $\tau_t = 50$ all g_n combinations performed well. However, for $\tau_t = 25$, p_s-g_n and p_n-g_n performed well, while p_d-g_n and p_r-g_n performed poorly.
- In both environments, all p_r combinations performed well or average, except p_r-g_s that performed poorly. All g_r combinations performed well for both $\tau_t = 25$ and $\tau_t = 50$, except p_n-g_r that performed average for $\tau_t = 25$ and poorly for $\tau_t = 50$.
- All p_d combinations performed poorly for $\tau_t = 25$, except p_d-g_r that performed well. However, for $\tau_t = 50$, p_d-g_r and p_d-g_n obtained a good performance, while the other two p_d combinations performed poorly. For the g_d combinations, all combinations performed well for $\tau_t = 25$, except p_d-g_d. For $\tau_t = 50$, all g_d combinations obtained an average performance.

For a severely changing environment ($n_t = 1$), the following observations are made:

- The best rank was obtained by p_s-g_r and the worst rank by p_n-g_s.
- All p_s combinations performed well, except p_s-g_s that performed poorly. In contrast, all g_s combinations performed rather poorly.
- Only one p_n combination, namely p_n-g_d performed well, while the other p_n combinations performed poorly. On the other hand, all g_n combinations performed well, except p_n-g_n that performed poorly.
- An average performance was obtained by p_r-g_s, and the other p_r combinations performed well. With the exception of p_d-g_r that performed really bad, all g_r combinations obtained a good rank.
- A good rank was obtained by p_d-g_n and p_d-g_r, an average rank by p_d-g_s and a poor rank by p_d-g_d. All g_d combinations performed well, except p_d-g_d that performed badly.

The following observations are made for a gradually changing environment ($n_t = 20$):

- The best performance was obtained by p_s-g_r, while p_d-g_s performed the worst.
- A really good performance was obtained by all p_s combinations, except p_s-g_s that

performed poorly. In contrast, a very bad rank was obtained by all g_s combinations.

- All p_n combinations performed well, except p_n that obtained a very poor performance. A good performance was also obtained by all g_n combinations.
- Two p_r combinations performed well, namely p_r-g_n and p_r-g_r. However, the other two p_r combinations performed badly. All g_r combinations performed well.
- All p_d combinations obtained an average performance, except p_d-g_s that obtained a very bad performance. A good or average performance was obtained by all g_d combinations, except p_r-g_d that performed poorly.

The original VEPSO algorithm's guide update approach, p_s-g_s, obtained the third and fourth highest rank for $n_t = 10$ and $\tau_t = 10$, and $n_t = 10$ and $\tau_t = 25$ respectively. However, p_s-g_s obtained rank thirteen, fourteen and twelve for $n_t = 10$ and $\tau_t = 50$, $n_t = 1$ and $\tau_t = 10$, and $n_t = 20$ and $\tau_t = 10$ respectively. Therefore, p_s-g_s struggles in slower changing environments, as well as environments that change either gradually or more severely.

Results for Various Dynamic Multi-objective Optimisation Problem Types

The DMOOPs against which DVEPSO was tested against, are of various DMOOP Types. With the different DMOOP Types, the POS or POF or both change over time. This section discusses the performance of the various guide update approaches with regards to the DMOOP Types I, II and III.

Type I DMOOPs

The wins and losses of the guide update approaches for Type I DMOOPs with regards to the performance measures over all n_t-τ_t combinations are presented in Table 9.3. The Type I DMOOPs are DIMP2 and dMOP3.

The following observations are made with regards to *acc*:

- The best performance with regards to *acc* was obtained by p_r-g_s and the worst performance by p_s-g_s.
- All p_s combinations performed really poor. Two g_s combinations performed well, namely p_r-g_s and p_d-g_s. However, the other two g_s combinations obtained a poor rank.

- Two p_n combinations, p_n-g_d and p_n-g_r, performed well, while the other two p_n combinations performed poorly. A similar trend was observed with the g_d combinations, where p_d-g_n and p_r-g_n performed well and the other two g_n combinations obtained a poor performance.
- For the p_r combinations, p_r-g_s and p_r-g_n performed really well and p_r-g_d and p_r-g_r performed average.
- All p_d combinations performed really well. In contrast, all g_d combinations obtained an average performance, except p_s-g_d that performed poorly.

Table 9.3: Overall Wins and Losses solving Type I DMOOPs for Various Performance Measures

PM	**Results**	**pbest-gbest combination**															
		s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
acc	Wins	0	0	1	0	26	27	33	30	34	31	31	28	29	28	32	32
acc	Losses	82	76	70	72	12	16	8	2	0	1	9	4	2	2	6	0
acc	Diff	-82	-76	-69	-72	14	11	25	28	34	30	22	24	27	26	26	32
acc	Rank	16	15	13	14	11	12	8	4	**1**	3	10	9	5	6	6	2
stab	Wins	1	0	0	0	10	8	13	10	16	8	9	9	8	11	10	14
stab	Losses	32	36	19	15	1	5	2	1	0	0	9	2	0	1	4	0
stab	Diff	-31	-36	-19	-15	9	3	11	9	16	8	0	7	8	10	6	14
stab	Rank	15	16	14	13	5	11	3	5	**1**	7	12	9	7	4	10	2
NS	Wins	0	11	0	0	13	14	12	12	12	12	12	13	12	12	12	12
NS	Losses	36	36	36	36	1	1	3	1	1	1	2	0	1	1	1	2
NS	Diff	-36	-25	-36	-36	12	13	9	11	11	11	10	13	11	11	11	10
NS	Rank	14	13	14	14	3	**1**	12	4	4	4	10	**1**	4	4	4	10
all	Wins	1	11	1	0	49	49	58	52	62	51	52	50	49	51	54	58
all	Losses	150	148	125	123	14	22	13	4	1	2	20	6	3	4	11	2
all	Diff	-149	-137	-124	-123	35	27	45	48	61	49	32	44	46	47	43	56
all	Rank	16	15	14	13	10	12	7	4	**1**	3	11	8	6	5	9	2

With regards to *stab*, the following observations are made:

- Similar to *acc*, the best rank was obtained by p_r-g_s. The worst rank was obtained by p_s-g_n.
- Similar to their performance with regards to *acc*, all p_s combinations performed poorly. With regards to the g_s combinations, all g_s combinations obtained a good performance, except p_s-g_s.
- All p_n combinations performed well, except p_n-g_n that performed badly. Similar to

acc, p_d-g_n and p_r-g_n performed well, while the other two g_n combinations obtained a poor performance.

- A good performance was obtained by all p_r combinations, except p_r-g_d that performed poorly. Similarly, all g_r combinations performed well, except p_s-g_r that obtained a poor rank.
- All p_d combinations performed well, except p_d-g_d that performed average. The g_d combinations obtained mixed results. A good performance was obtained by p_n-g_d, an average performance by p_d-g_d and a poor performance by p_r-g_d and p_s-g_d.

The following observations are made with regards to NS:

- The best performance was obtained by p_r-g_r.
- Once again, all p_s combinations performed poorly. However, all g_s combinations performed well, except p_s-g_s that performed badly.
- Similar to acc and $stab$, all p_n combinations obtained a good performance, except p_n-g_n that performed poorly. The same trend was observed for g_d combinations, with all performing well, except p_s-g_n that performed badly.
- All p_r combinations performed well, with the exception of p_r-g_d that performed average. Two g_r combinations, p_r-g_r and p_n-g_r, performed well, p_d-g_r performed average and p_s-g_r performed poorly.
- With the exception of p_d-g_r that performed average, all p_r combinations performed well. In contrast, p_d-g_d performed well, p_r-g_d performed average and the other two g_d combinations performed badly.

Table 9.4 presents the wins and losses measured over all performance measures for the various n_t-τ_t combinations for Type I DMOOPs.
The following observations are made with regards to the obtained results:

- All p_s combinations performed poorly for all n_t-τ_t combinations, except p_s-g_s that performed well for $n_t = 20$ and $\tau_t = 10$. For the g_s combinations, three combinations performed well with one performing poorly for $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 25$, and $n_t = 20$ and $\tau_t = 10$. For $n_t = 10$ and $\tau_t = 50$, and $n_t = 1$ and $\tau_t = 10$, two g_s combinations performed well, and two combinations performed badly.

Table 9.4: Overall Wins and Losses solving Type I DMOOPs for Various Frequencies and Severities of Change

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**Results**	**pbest-gbest combination**															
			s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	Wins	0	0	0	0	22	22	21	20	24	23	22	22	22	23	27	33
10	10	Losses	63	80	68	63	0	0	2	3	0	0	0	0	0	0	2	0
10	10	Diff	-63	-80	-68	-63	22	22	19	17	24	23	22	22	22	23	25	33
10	10	Rank	13	16	15	13	6	6	11	12	3	4	6	6	6	4	2	**1**
10	25	Wins	0	0	1	0	9	6	6	11	7	5	5	7	5	4	7	7
10	25	Losses	30	14	12	22	0	0	0	0	0	0	0	0	0	2	0	0
10	25	Diff	-30	-14	-11	-22	9	6	6	11	7	5	5	7	5	2	7	7
10	25	Rank	16	14	13	15	2	7	7	**1**	3	9	9	3	9	12	3	3
10	50	Wins	0	11	0	0	6	5	7	6	6	6	11	8	7	7	6	6
10	50	Losses	21	24	13	13	2	7	1	1	1	1	2	2	1	1	1	1
10	50	Diff	-21	-13	-13	-13	4	-2	6	5	5	5	9	6	6	6	5	5
10	50	Rank	16	13	13	13	11	12	2	6	6	6	**1**	2	2	2	6	6
1	10	Wins	0	0	0	0	9	10	24	11	24	12	14	10	15	11	10	11
1	10	Losses	36	30	25	25	12	15	2	0	0	1	1	4	0	1	8	1
1	10	Diff	-36	-30	-25	-25	-3	-5	22	11	24	11	13	6	15	10	2	10
1	10	Rank	16	15	13	13	11	12	2	5	**1**	5	4	9	3	7	10	7
20	10	Wins	1	0	0	0	3	6	0	4	1	5	0	3	0	6	4	1
20	10	Losses	0	0	7	0	0	0	8	0	0	0	17	0	2	0	0	0
20	10	Diff	1	0	-7	0	3	6	-8	4	1	5	-17	3	-2	6	4	1
20	10	Rank	8	11	14	11	6	**1**	15	4	8	3	16	6	13	**1**	4	8
all	all	Wins	1	11	1	0	49	49	58	52	62	51	52	50	49	51	54	58
all	all	Losses	150	148	125	123	14	22	13	4	1	2	20	6	3	4	11	2
all	all	Diff	-149	-137	-124	-123	35	27	45	48	61	49	32	44	46	47	43	56
all	all	Rank	16	15	14	13	10	12	7	4	**1**	3	11	8	6	5	9	2

- Two p_n combinations performed well and two poorly for $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 50$, and $n_t = 1$ and $\tau_t = 10$. All p_n combinations performed well for $n_t = 10$ and $\tau_t = 25$. For $n_t = 20$ and $\tau_t = 10$, three p_n combinations performed well and only one performed badly. The g_n combinations also obtained mixed results. For $n_t = 10$ and $\tau_t = 10$, and $n_t = 20$ and $\tau_t = 10$, three combinations performed well and one performed badly. On the other hand, for the other n_t-τ_t combinations two g_n combinations performed well and two poorly.
- All p_r combinations performed well for all n_t-τ_t combinations, except p_r-g_d that performed poorly for $n_t = 20$ and $\tau_t = 10$. For all n_t-τ_t combinations, three g_r combinations obtained a good performance and one combination obtained a poor performance, except for $n_t = 10$ and $\tau_t = 10$ where two combinations performed

badly.

- For the p_d combinations, all performed well for $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 50$, and $n_t = 1$ and $\tau_t = 10$. For $n_t = 10$ and $\tau_t = 25$, and $n_t = 20$ and $\tau_t = 10$, all p_d combinations, except p_s-g_d, obtained a good performance. Three of the g_d combinations obtained a good performance and one combination obtained a poor performance for $n_t = 10$ and $\tau_t = 25$, and $n_t = 10$ and $\tau_t = 50$. For $n_t = 10$ and $\tau_t = 10$, and $n_t = 1$ and $\tau_t = 10$, two combinations performed well and two performed poorly. Furthermore, for $n_t = 20$ and $\tau_t = 10$, only one g_d combination obtained a good performance and the other three obtained a bad performance.

Type II DMOOPs

The wins and losses for Type II DMOOPs with regards to the performance measures over all n_t-τ_t combinations are presented in Table 9.5. The Type II DMOOPs are FDA1$_{Zhou}$, FDA2, FDA3, FDA3$_{Camara}$, dMOP2, dMOP2$_{iso}$ and dMOP2$_{dec}$.

Table 9.5: Overall Wins and Losses solving Type II DMOOPs for Various Performance Measures

PM	**Results**	**pbest-gbest combination**															
		s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
acc	Wins	62	95	87	144	47	76	73	47	56	82	53	81	51	70	45	86
acc	Losses	181	144	155	73	64	48	30	68	58	32	58	35	64	48	62	35
acc	Diff	-119	-49	-68	71	-17	28	43	-21	-2	50	-5	46	-13	22	-17	51
acc	Rank	16	14	15	**1**	11	6	5	13	8	3	9	4	10	7	11	2
stab	Wins	100	109	111	92	46	33	15	46	50	16	46	13	49	16	45	16
stab	Losses	52	37	34	68	30	71	69	43	27	59	36	70	28	80	28	71
stab	Diff	48	72	77	24	16	-38	-54	3	23	-43	10	-57	21	-64	17	-55
stab	Rank	3	2	**1**	4	8	11	13	10	5	12	9	15	6	16	7	14
NS	Wins	52	83	67	126	32	109	141	43	45	107	47	139	25	107	49	146
NS	Losses	147	110	125	81	132	45	28	73	127	44	72	38	135	56	75	30
NS	Diff	-95	-27	-58	45	-100	64	113	-30	-82	63	-25	101	-110	51	-26	116
NS	Rank	14	10	12	7	15	4	2	11	13	5	8	3	16	6	9	**1**
all	Wins	214	287	265	362	125	218	229	136	151	205	146	233	125	193	139	248
all	Losses	380	291	314	222	226	164	127	184	212	135	166	143	227	184	165	136
all	Diff	-166	-4	-49	140	-101	54	102	-48	-61	70	-20	90	-102	9	-26	112
all	Rank	16	8	12	**1**	14	6	3	11	13	5	9	4	15	7	10	2

The following are observed with regards to *acc*:

- The best performance was achieved by p_s-g_r and the worst by p_s-g_s.

- All p_s combinations performed poorly, except p_s-g_r that performed really well. Two g_s combinations, p_r-g_s and p_d-g_s performed average and the other two g_s combinations obtained a poor performance.
- Two p_n combinations, p_n-g_s and p_n-g_r performed well and the other two performed badly. All g_n combinations performed well, except p_s-g_n that performed poorly.
- A good performance was obtained by all p_r combinations. With the exception of p_n-g_r that performed poorly, all g_r combinations obtained a good performance.
- The p_d combinations obtained mixed results. A good performance was obtained by p_d-g_n and p_d-g_r, an average performance by p_d-g_s and a poor performance by p_d-g_d. Two g_s combinations obtained a good performance, namely p_n-g_d and p_r-g_d. The other two g_d combinations performed badly.

The following observations are made with regards to *stab*:

- The best rank was obtained by p_s-g_d and the worst by p_d-g_n.
- In contrast to *acc*, all p_s combinations performed really well with regards to *stab*, obtaining the top four ranks. Furthermore, all g_s combinations obtained a good performance.
- Two p_n combinations, p_n-g_s and p_n-g_r obtained an average performance. The other two p_n combinations performed poorly. All g_n combinations performed badly with the exception of p_s-g_n that performed very good.
- For the p_r combinations, p_r-g_s and p_r-g_d performed well, but the other two combinations performed badly. The g_r combinations obtained mixed results, with p_s-g_r performing well, p_n-g_r performing average and the other two g_r combinations performing poorly.
- Two p_d combinations obtained a good performance, namely p_d-g_r and p_d-g_n. Furthermore, an average performance was obtained by p_d-g_s and a poor performance by p_d-g_d. In contrast, all g_d combinations performed well, except p_s-g_d that performed badly.

The following are observed with regards to NS:

- The best performance was achieved by p_d-g_r and the worst by p_d-g_s.
- Two p_s combinations, p_s-g_r and p_s-g_n, performed well. The other two p_s combina-

tions performed poorly. A bad performance was also obtained by all g_s combinations.

- For the p_n combinations, p_n-g_n and p_n-g_d performed well, but the other two p_n combinations performed badly. All g_n combinations performed well or average.
- All p_r combinations, except p_r-g_s, performed well. Similarly, all g_r combinations obtained a good performance, except p_n-g_r that performed poorly.
- For the p_d combinations, all performed well, except p_d-g_s that performed badly. Similarly, all g_d combinations obtained a good rank, except p_s-g_d that obtained a poor rank.

Table 9.6 presents the wins and losses measured over all performance measures for the various n_t-τ_t combinations for Type II DMOOPs.

Table 9.6: Overall Wins and Losses solving Type II DMOOPs for Various Frequencies and Severities of Change

$\mathbf{n_t}$	τ_t	**Results**	**pbest-gbest combination**															
			s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	Wins	58	64	67	89	31	35	46	32	28	38	26	46	36	40	32	48
10	10	Losses	85	76	67	41	45	48	30	38	38	39	40	32	38	33	39	27
10	10	Diff	-27	-12	0	48	-14	-13	16	-6	-10	-1	-14	14	-2	7	-7	21
10	10	Rank	16	12	6	**1**	14	13	3	9	11	7	14	4	8	5	10	2
10	25	Wins	72	75	81	75	14	39	29	22	23	30	41	34	21	20	22	44
10	25	Losses	44	39	30	41	62	33	32	38	58	33	33	34	61	34	36	34
10	25	Diff	28	36	51	34	-48	6	-3	-16	-35	-3	8	0	-40	-14	-14	10
10	25	Rank	4	2	**1**	3	16	7	9	13	14	9	6	8	15	11	11	5
10	50	Wins	25	54	39	83	34	76	90	37	45	78	38	90	25	77	45	96
10	50	Losses	103	75	93	82	77	41	33	40	71	32	41	41	82	54	37	30
10	50	Diff	-78	-21	-54	1	-43	35	57	-3	-26	46	-3	49	-57	23	8	66
10	50	Rank	16	11	14	8	13	5	2	9	12	4	9	3	15	6	7	**1**
1	10	Wins	42	65	57	67	24	35	32	26	27	31	26	30	22	34	24	32
1	10	Losses	78	64	77	38	21	27	18	46	28	16	31	21	22	33	31	23
1	10	Diff	-36	1	-20	29	3	8	14	-20	-1	15	-5	9	0	1	-7	9
1	10	Rank	16	8	14	**1**	7	6	3	14	11	2	12	4	10	8	13	4
20	10	Wins	17	29	21	48	22	33	32	19	28	28	15	33	21	22	16	28
20	10	Losses	70	37	47	20	21	15	14	22	17	15	21	15	24	30	22	22
20	10	Diff	-53	-8	-26	28	1	18	18	-3	11	13	-6	18	-3	-8	-6	6
20	10	Rank	16	13	15	**1**	8	2	2	9	6	5	11	2	9	13	11	7
all	all	Wins	214	287	265	362	125	218	229	136	151	205	146	233	125	193	139	248
all	all	Losses	380	291	314	222	226	164	127	184	212	135	166	143	227	184	165	136
all	all	Diff	-166	-4	-49	140	-101	54	102	-48	-61	70	-20	90	-102	9	-26	112
all	all	Rank	16	8	12	**1**	14	6	3	11	13	5	9	4	15	7	10	2

The following observations are made with regards to the obtained results:

- All p_s combinations performed well for $n_t = 10$ and $\tau_t = 25$. For $n_t = 10$ and $\tau_t = 10$, and $n_t = 1$ and $\tau_t = 10$, two p_s combinations performed well and two performed poorly. For the other two n_t-τ_t combinations, only one p_s combination performed well and the other p_s combinations performed badly. For $n_t = 20$ and $\tau_t = 10$, all g_s combinations performed well, except one that performed really bad. For $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 25$, and $n_t = 1$ and $\tau_t = 10$, three p_s combinations performed poorly and only one performed well. Furthermore, for $n_t = 20$ and $\tau_t = 10$, all g_s combinations obtained a poor performance.
- For $n_t = 20$ and $\tau_t = 10$, all p_n combinations obtained a good performance. Three p_n combinations performed well and one performed poorly for $n_t = 10$ and $\tau_t = 50$, and $n_t = 1$ and $\tau_t = 10$. However, for $n_t = 10$ and $\tau_t = 10$, and $n_t = 10$ and $\tau_t = 25$, two p_n combinations obtained a good performance and the other two a poor performance. All g_n combinations performed well for $n_t = 1$ and $\tau_t = 10$. For $n_t = 10$ and $\tau_t = 25$, and $n_t = 10$ and $\tau_t = 50$, all g_n combinations obtained a good performance, except one that performed badly. For the other two n_t-τ_t combinations, two g_n combinations obtained good ranks and the other two obtained poor ranks.
- All p_r combinations, except one, obtained a good performance for $n_t = 10$ and $\tau_t = 25$, $n_t = 10$ and $\tau_t = 50$, and $n_t = 20$ and $\tau_t = 10$. For $n_t = 10$ and $\tau_t = 10$, and $n_t = 1$ and $\tau_t = 10$, two p_r combinations performed well and two performed badly. For the g_r combinations, all performed well for $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 50$, and $n_t = 20$ and $\tau_t = 10$. Furthermore, for $n_t = 10$ and $\tau_t = 25$, and $n_t = 1$ and $\tau_t = 10$, only one g_r combination performed poorly and all the other obtained a good performance.
- For $n_t = 10$ and $\tau_t = 10$, all p_d combinations performed well or average. Three p_d combinations performed well with only one p_d combination performing badly for $n_t = 10$ and $\tau_t = 25$. For $n_t = 20$ and $\tau_t = 10$, two p_d combinations obtained a good performance and two obtained a poor performance. Only one p_d combination performed well for $n_t = 10$ and $\tau_t = 25$, with three p_d combinations obtaining a poor performance. For $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 25$, and $n_t =$

10 and $\tau_t = 50$, all g_d combinations performed well, except one that performed poorly. However, three g_d combinations obtained a poor performance and only one performed well for $n_t = 1$ and $\tau_t = 10$, and $n_t = 20$ and $\tau_t = 10$.

Type III DMOOPs

The wins and losses of the guide update approaches for Type III DMOOPs with regards to the performance measures over all n_t-τ_t combinations are presented in Table 9.7. The Type III DMOOPs are FDA2$_{Camara}$, HE1, HE2, HE6, HE7 and HE9.

Table 9.7: Overall Wins and Losses solving Type III DMOOPs for Various Performance Measures

PM	**Results**	**pbest-gbest combination**															
		s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
acc	Wins	160	124	141	143	77	43	58	72	75	48	72	53	91	40	67	70
acc	Losses	84	125	88	80	93	101	71	58	91	87	56	84	86	101	69	60
acc	Diff	76	-1	53	63	-16	-58	-13	14	-16	-39	16	-31	5	-61	-2	10
acc	Rank	**1**	8	3	2	11	15	10	5	11	14	4	13	7	16	9	6
stab	Wins	196	137	189	175	5	9	11	6	6	8	4	6	7	8	5	4
stab	Losses	27	37	16	13	57	56	55	69	57	54	57	54	56	55	57	56
stab	Diff	169	100	173	162	-52	-47	-44	-63	-51	-46	-53	-48	-49	-47	-52	-52
stab	Rank	2	4	**1**	3	12	7	5	16	11	6	15	9	10	7	12	12
NS	Wins	215	302	281	323	22	103	90	77	24	117	82	89	25	114	74	93
NS	Losses	128	66	61	57	258	105	100	123	259	91	101	121	249	90	113	109
NS	Diff	87	236	220	266	-236	-2	-10	-46	-235	26	-19	-32	-224	24	-39	-16
NS	Rank	4	2	3	**1**	16	7	8	13	15	5	10	11	14	6	12	9
all	Wins	571	563	611	641	104	155	159	155	105	173	158	148	123	162	146	167
all	Losses	239	228	165	150	408	262	226	250	407	232	214	259	391	246	239	225
all	Diff	332	335	446	491	-304	-107	-67	-95	-302	-59	-56	-111	-268	-84	-93	-58
all	Rank	4	3	2	**1**	16	12	8	11	15	7	5	13	14	9	10	6

The following observations are made with regards to *acc*:

- The best performance was obtained by p_s-g_s and the worst performance by p_d-g_n.
- All p_s combinations performed well. Two g_s combinations, p_s-g_s and p_d-g_s, obtained a good performance, while the other two g_s combinations performed poorly.
- The p_n combinations obtained mixed results, with p_n-g_r performing well, p_n-g_d performing average and the other two p_n combinations performing badly. All g_n combinations obtained a poor performance, except p_s-g_n that performed well.
- All p_r combinations obtained a poor performance, except p_r-g_d that obtained a

good performance. In contrast, all g_r combinations performed well, except p_r-g_r that performed poorly.

- A good performance was obtained by all p_d combinations, except p_d-g_n that performed badly. Furthermore, all g_d combinations performed well or average.

With regards to *stab*, the following observations are made:

- A p_s combination, p_s-g_d, obtained the best performance. The worst performance was obtained by p_n-g_r.
- All p_s combinations performed very well, obtaining the top four ranks. However, only one g_s combination, p_s-g_s performed well, while the others performed poorly.
- Two p_n combinations obtained a good or average performance, namely p_n-g_r and p_n-g_d. The other two p_n combinations perfromed badly. In contrast, all g_n combinations obtained a good rank.
- For the p_r combinations, p_r-g_d and p_r-g_r performed well, while the other two p_r combinations performed poorly. Similarly, p_s-g_r and p_r-g_r obtained a good performance, while the other g_r combinations obtained a poor performance.
- Only one p_d combination, p_d-g_n, performed well. All other p_d combinations performed badly. For the g_d combinations, p_s-g_d and p_n-g_d obtained a good performance, but the other g_d combinations obtained a poor performance.

The following observations are made with regards to NS:

- The best performance was obtained by p_s-g_r and the worst performance by p_n-g_s.
- Similar to *stab*, all p_s combinations performed well, obtaining the top four ranks. However, similar to *stab*, for g_s only p_s-g_s obtained a good performance and the other g_s combinations performed poorly.
- Similar to *stab*, p_n-g_r and p_n-g_d performed well, while the other p_n combinations performed badly. Furthermore, also similar to *stab*, all g_n combinations obtained a good performance.
- Two p_r combinations, p_r-g_n and p_r-g_d, obtained a good or average performance. The other two p_r combinations obtained a poor performance. For g_r, p_s-g_r and p_d-g_r performed well, while the other g_r combinations performed badly.
- For p_d, p_d-g_n and p_d-g_r performed well, while p_d-g_s and p_d-g_d obtained a poor

performance. Two g_d combinations, p_s-g_d and p_n-g_d obtained a good performance. Furthermore, p_r-g_d obtained an average performance and p_d-g_d performed poorly.

Table 9.8 presents the wins and losses measured over all performance measures for the various n_t-τ_t combinations for Type III DMOOPs.

Table 9.8: Overall Wins and Losses solving Type III DMOOPs for Various Frequencies and Severities of Change

$\mathbf{n_t}$	τ_t	**Results**	**pbest-gbest combination**															
			s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	Wins	121	115	132	136	14	24	33	34	12	31	35	30	16	30	30	26
10	10	Losses	35	48	20	31	85	58	46	49	86	44	40	53	78	50	47	49
10	10	Diff	86	67	112	105	-71	-34	-13	-15	-74	-13	-5	-23	-62	-20	-17	-23
10	10	Rank	3	4	**1**	2	15	13	6	8	16	6	5	11	14	10	9	11
10	25	Wins	110	122	121	107	9	18	20	26	11	21	21	24	12	21	22	20
10	25	Losses	27	21	16	21	72	44	40	41	69	45	39	48	75	43	40	44
10	25	Diff	83	101	105	86	-63	-26	-20	-15	-58	-24	-18	-24	-63	-22	-18	-24
10	25	Rank	4	2	**1**	3	15	13	8	5	14	10	6	10	15	9	6	10
10	50	Wins	101	107	112	136	15	43	39	29	17	49	23	32	26	32	24	46
10	50	Losses	58	49	39	35	87	47	37	52	83	45	41	45	81	48	50	34
10	50	Diff	43	58	73	101	-72	-4	2	-23	-66	4	-18	-13	-55	-16	-26	12
10	50	Rank	4	3	2	**1**	16	8	7	12	15	6	11	9	14	10	13	5
1	10	Wins	132	104	123	133	49	40	45	39	49	44	45	40	52	51	41	52
1	10	Losses	83	83	74	43	84	63	51	63	88	54	51	57	78	59	57	51
1	10	Diff	49	21	49	90	-35	-23	-6	-24	-39	-10	-6	-17	-26	-8	-16	1
1	10	Rank	2	4	2	**1**	15	12	6	13	16	9	6	11	14	8	10	5
20	10	Wins	107	115	123	129	17	30	22	27	16	28	34	22	17	28	29	23
20	10	Losses	36	27	16	20	80	50	52	45	81	44	43	56	79	46	45	47
20	10	Diff	71	88	107	109	-63	-20	-30	-18	-65	-16	-9	-34	-62	-18	-16	-24
20	10	Rank	4	3	2	**1**	15	10	12	8	16	6	5	13	14	8	6	11
all	all	Wins	571	563	611	641	104	155	159	155	105	173	158	148	123	162	146	167
all	all	Losses	239	228	165	150	408	262	226	250	407	232	214	259	391	246	239	225
all	all	Diff	332	335	446	491	-304	-107	-67	-95	-302	-59	-56	-111	-268	-84	-93	-58
all	all	Rank	4	3	2	**1**	16	12	8	11	15	7	5	13	14	9	10	6

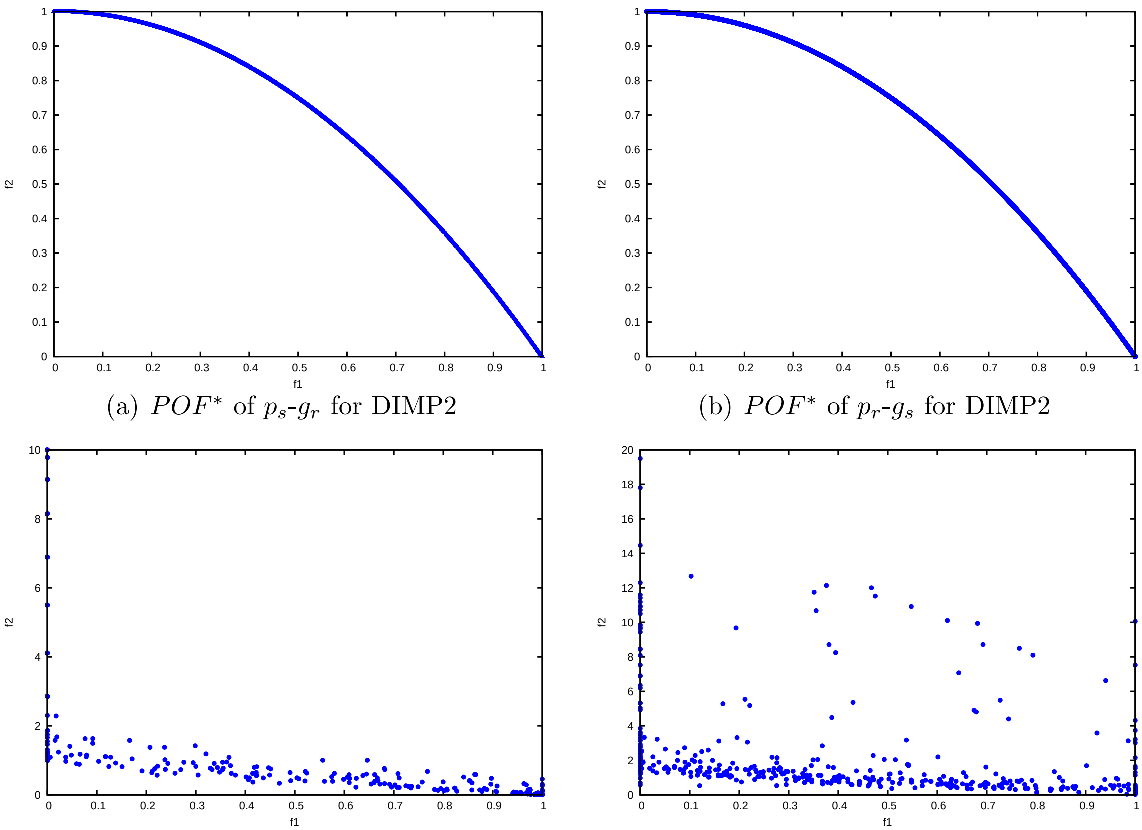
The following are observed with regards to the obtained results:

- All p_s combinations performed really well, obtaining the top four ranks for all n_t-τ_t combinations. For g_s mixed results were obtained. For $n_t = 1$ and $\tau_t = 10$, two g_s combinations obtained a good performance, while the other two performed badly. For the other n_t-τ_t combinations, only one g_s combination performed well, while the other three g_s combinations obtained a poor performance.

- For $n_t = 1$ and $\tau_t = 10$, only one p_n combination obtained a good performance, while the other p_n combinations performed poorly. Two p_n combinations performed well for the other n_t-τ_t combinations, while the other two p_n combinations obtained a bad rank. For $n_t = 10$ and $\tau_t = 50$, and $n_t = 1$ and $\tau_t = 10$, all g_n combinations obtained a good or average performance. Three g_n combinations performed well and one performed poorly for all the other n_t-τ_t combinations.
- Three p_r combinations obtained a good performance and one, p_r-g_s, a poor performance for $n_t = 10$ and $\tau_t = 25$, and $n_t = 1$ and $\tau_t = 10$. For the other n_t-τ_t combinations, two p_r combinations produced good ranks and two produced poor ranks. All g_r combinations performed good or average for $n_t = 10$ and $\tau_t = 25$. For $n_t = 10$ and $\tau_t = 10$, two g_r combinations obtained a good performance and two obtained a poor performance. Three g_r combinations obtained good ranks and one obtained a poor rank for all the other n_t-τ_t combinations.
- A similar trend to p_r was observed for p_d. For $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $\tau_t = 25$, and $n_t = 1$ and $\tau_t = 10$, all g_d combinations performed good or average. Three g_d combinations obtained a good rank and one obtained a poor rank for $n_t = 20$ and $\tau_t = 10$. Furthermore, for $n_t = 10$ and $\tau_t = 50$, two g_d combinations performed well, and two performed badly.

General Observations with regards to DMOOP Types

It is interesting to note the difference in performance obtained by the p_s combinations for the three types of DMOOPs. It should be noted that although the p_s combinations ranked poorly for the Type I DMOOPs and three p_s combinations ranked poorly for the Type II DMOOPs, it does not indicate that they didn't successfully track the changing POF. It only indicates that the other guide-update approaches' performance measure values were statistically better, resulting in more wins. This is indicated in Figure 9.2. Figure 9.2 illustrates the approximated POF of the best performing p_s combination (p_s-g_r) and the best performing guide update approach for the Type I DMOOPs (p_r-g_s). When solving DIMP2, both guide update approaches successfully found the POF. However, over the various runs, p_s-g_r did find more outlier solutions that were further away from the true POF than p_r-g_s. Therefore, p_r-g_s obtained a better rank. For dMOP3, both guide update approaches found solutions close to the true POF. Even though p_r-g_s found more outlier solutions than p_s-g_r in some of the runs, p_r-g_s found much more solutions with a better spread than p_s-g_r. Therefore, p_r-g_s obtained better perfomance measure values.



(a) POF^* of p_s-g_r for DIMP2
(b) POF^* of p_r-g_s for DIMP2
(c) POF^* of p_s-g_r for dMOP3
(d) POF^* of p_r-g_s for dMOP3

Figure 9.2: POF^* of p_s-g_r on the right and p_r-g_s on the left for $n_t = 10$ and $\tau_t = 10$

The next section discusses the overall performance of the various guide update approaches. This overall perfromance is measured over all performance measures and all n_t-τ_t combinations.

Overall Performance

The overall wins and losses obtained by the various guide update approaches are presented in Table 9.9.

With regards to the overall performance of the guide update approaches, the following observations are made:

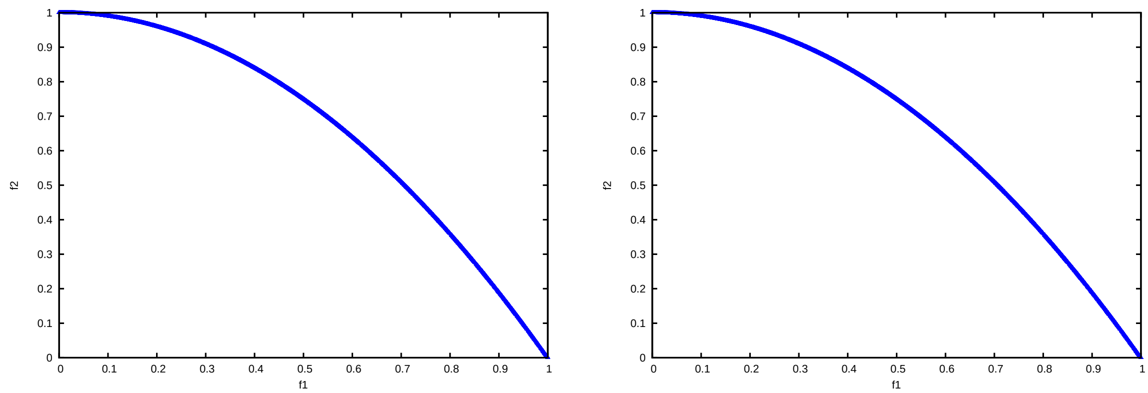
- The best overall rank was obtained by p_s-g_r and the worst by p_n-g_s.
- All p_s combinations obtained a good rank. However, all g_s combinations, except p_s-g_s, ranked the worst.
- Two p_n combinations, namely p_n-g_d and p_n-g_n, obtained a good and average rank respectively. The other two p_n combinations obtained a bad rank. On the other hand, an average or good performance was obtained by all g_n combinations.
- Two p_d combinations performed well, namely p_d-g_n and p_d-g_r. The other two p_d combinations performed poorly. Similarly, two g_d combinations, namely p_s-g_d and p_n-g_d, performed well, while the other two performed poorly.
- For the p_r combinations, p_r-g_n and p_r-g_r performed average and the other two p_r combinations obtained a poor rank. In contrast, all g_r combinations, except p_n-g_r, performed well.

Table 9.9: Overall Wins and Losses by the various guide update approaches

Results	**pbest-gbest combination**															
	s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
Wins	786	861	877	1003	278	422	446	343	318	429	356	431	297	406	339	473
Losses	761	660	594	492	641	445	366	431	613	366	390	407	613	428	408	363
Diff	25	201	283	511	-363	-23	80	-88	-295	63	-34	24	-316	-22	-69	110
Rank	7	3	2	**1**	16	10	5	13	14	6	11	8	15	9	12	4

The best performing guide update approach, p_s-g_r uses no Pareto-dominance information for the pbest update. This enables the swarm to focus on optimising its specific objective, without taking the other objectives into account. However, for the gbest update Pareto-dominance information is taken into account. When a pbest is non-dominated with regards to the gbest, either the pbest or the current gbest is randomly selected as the new gbest. Therefore, Pareto-dominance is not required for a gbest update.

POFs found by DVEPSO using the p_s-g_r guide update approach during a single run for DIMP2 are illustrated in Figure 9.3. In Figure 9.3, POFs found for $n_t = 10$ and $\tau_t = 10$ (fast changing environment) are shown on the left and for $n_t = 1$ and $\tau_t = 10$ (severely changing environment) are shown on the right. The figures indicate that DVEPSO successfully tracked the changing POF of DIMP2 in both a fast changing and severely changing environment.

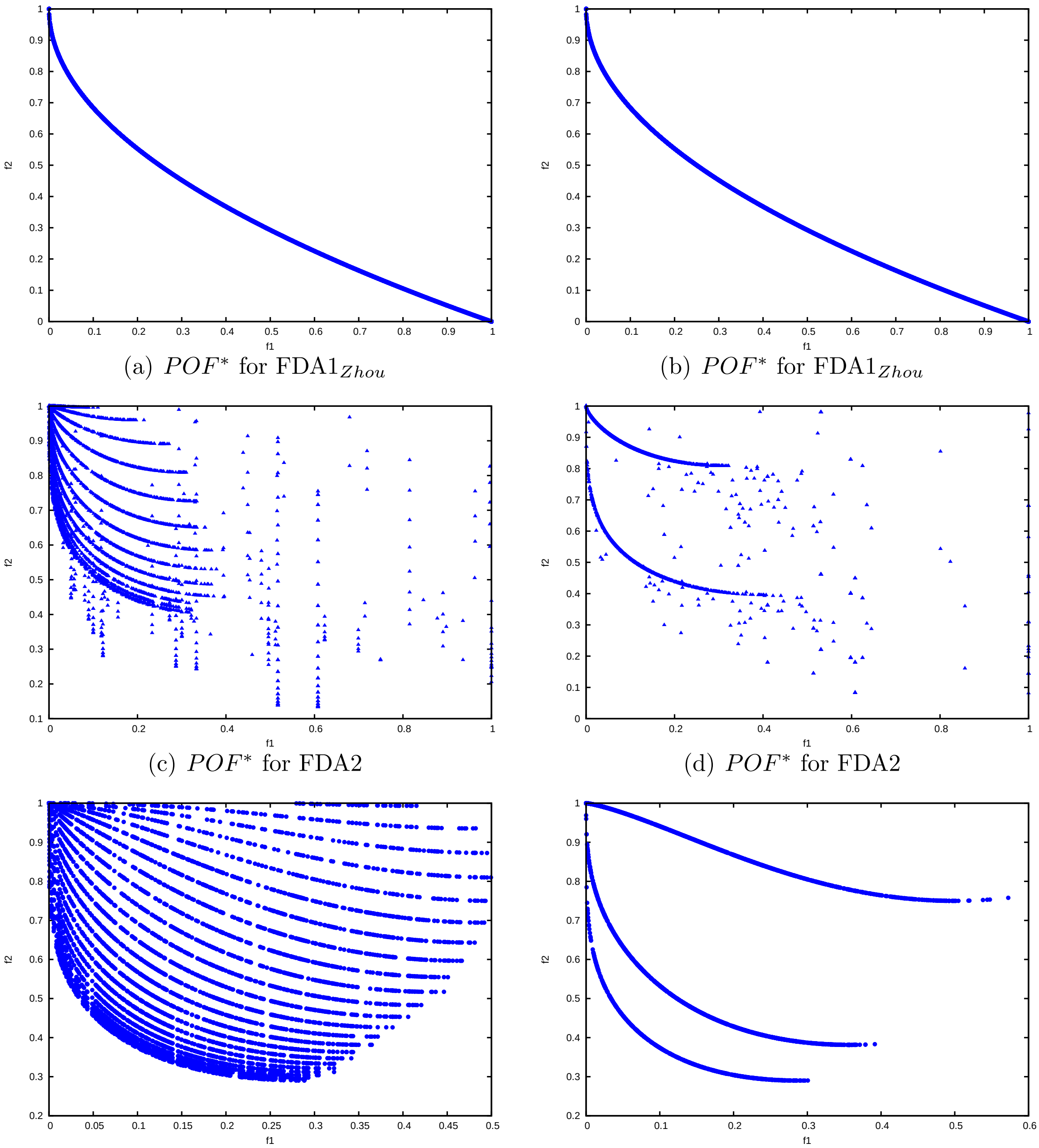


(a) (b)

Figure 9.3: POF^* for DIMP2 of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right

Figures 9.4 and 9.5 illustrate POFs found by DVEPSO using the p_s-g_r guide update approach during a single run for the FDA DMOOPs. Figures 9.4(a) and 9.4(b) indicate that DVEPSO successfully tracked the changing POF over time for FDA1$_{Zhou}$ in both a fast changing and severely changing environment. For FDA2, DVEPSO struggled to track the changing POF for every change in the environment, but did find a POF^* close to POF for many time steps, even though the spread of solutions were not that good. This is illustrated in Figures 9.4(c) and 9.4(d). However, for FDA2$_{Camara}$, DVEPSO successfully tracked the changing POF^* over time with a good spread of solutions as indicated in Figures 9.4(e) and 9.4(f). From Figure 9.5 it can be seen that DVEPSO successfully tracked the changing POF, finding a good spread of solutions for both $FDA3$ and FDA3$_{Camara}$ with a fast changing environment. However, with a severely changing environment, DVEPSO did not find as good a spread of solutions as in the case with a fast changing environment.

POFs found by DVEPSO using the p_s-g_r guide update approach during a single run for the dMOP DMOOPs are illustrated in Figures 9.6 and 9.7. When solving dMOP2, DVEPSO successfully tracked the changing POF over time for both a fast changing,



(a) POF^* for FDA1$_{Zhou}$

(b) POF^* for FDA1$_{Zhou}$

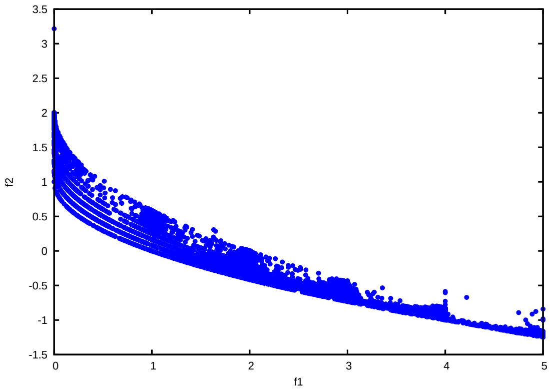
(c) POF^* for FDA2

(d) POF^* for FDA2

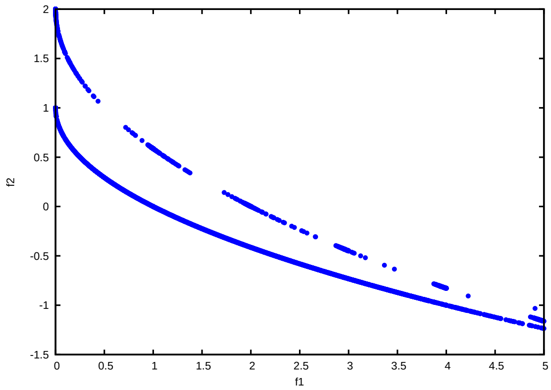
(e) POF^* for FDA2$_{Camara}$

(f) POF^* for FDA2$_{Camara}$

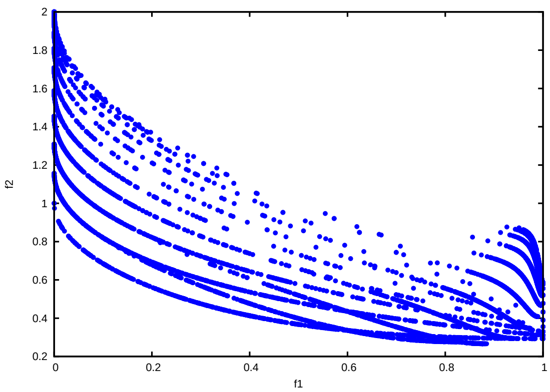
Figure 9.4: POF^* for FDA1 and FDA2 functions of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right



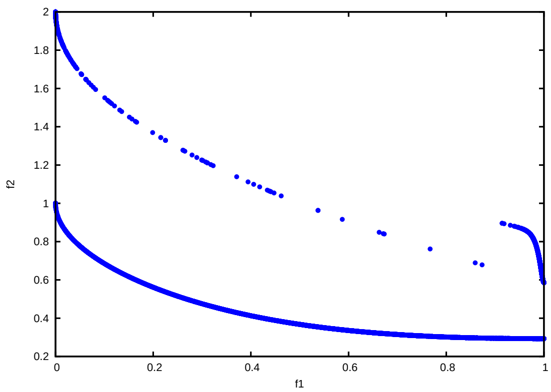
(a) POF^* for FDA3



(b) POF^* for FDA3



(c) POF^* for FDA3$_{Camara}$



(d) POF^* for FDA3$_{Camara}$

Figure 9.5: POF^* for FDA3 functions of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right

and severely changing environment, as illustrated in Figure 9.6. However, DVEPSO also found many outlier solutions. Figure 9.6 indicates the outlier solutions found by DVEPSO while solving the dMOP2 functions. The POF^* found by DVEPSO for the dMOP2 functions without the outliers are shown in Figure 9.7. When dMOP2 had a deceptive POF, DVEPSO found outlier solutions that were very far away from POF as can be seen in Figure 9.6(h). These outlier solutions caused large reference vectors being used to calculate the HV values and therefore very large acc values were reported (refer to Appendix D).

When solving dMOP3 with a fast changing environment, DVEPSO found a good spread of solutions in the area of POF. However, with a severely changing environment, DVEPSO found a reasonable spread of solutions reasonably close to POF. This is illustrated in Figures 9.6(a) and 9.6(b).

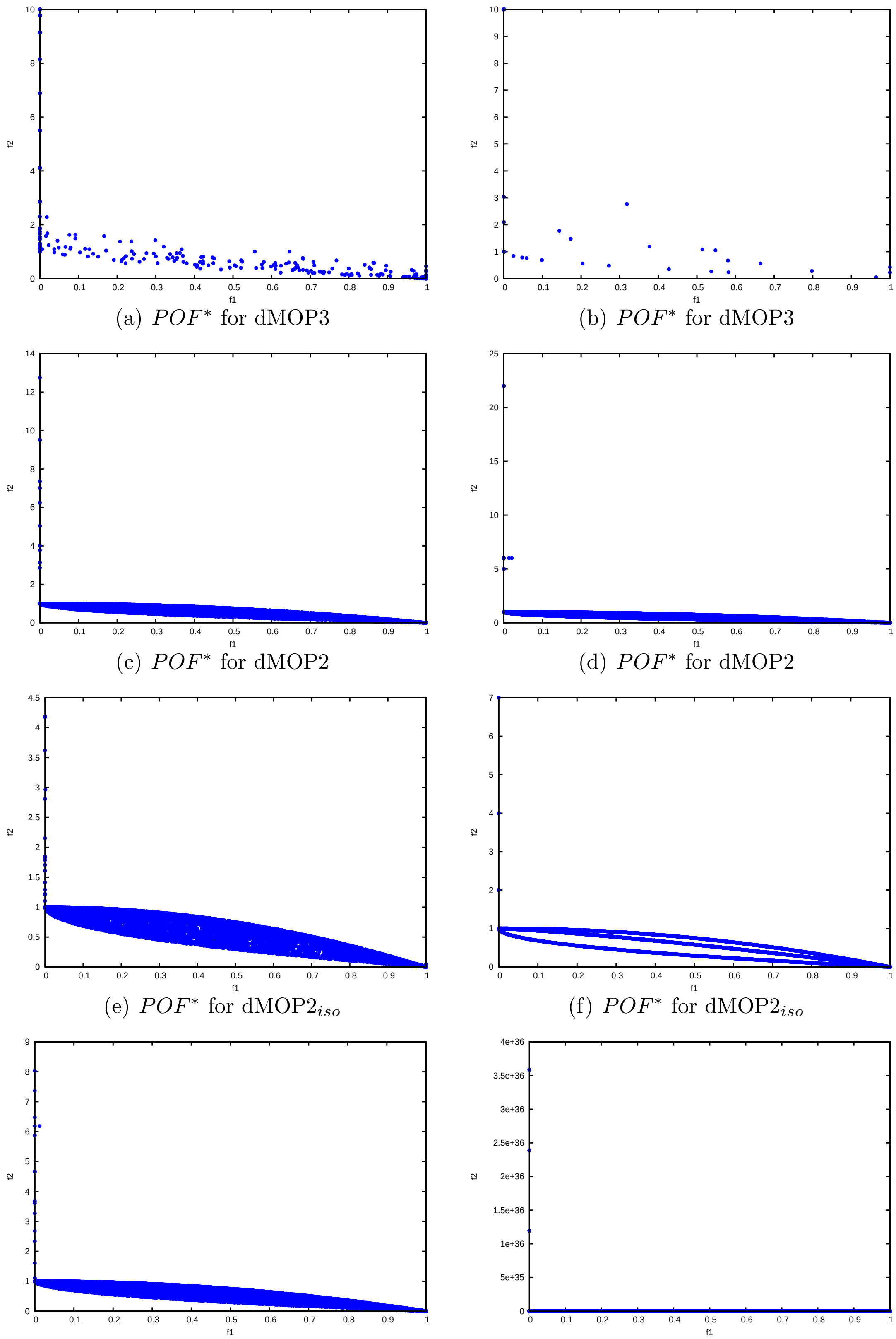
POFs found by DVEPSO using the p_s-g_r guide update approach during a single run for the HE DMOOPs are illustrated in Figures 9.8 and 9.9. Figure 9.8 indicates that DVEPSO struggled to converge to the discontinuous POFs of HE1 and HE2. When solving HE6 and HE7, DVEPSO found a POF^* that was close to POF. However, DVEPSO also found many solutions further away from POF as can be seen in Figure 9.9. When solving HE9, DVEPSO only found a few solutions and struggled to converge to the POF, as illustrated in Figures 9.9(e) and 9.9(f).

The original VEPSO algorithm's guide update approach, p_s-g_s, obtained the seventh overall rank. The other three p_s combinations obtained the top three overall ranks. Therefore, the results indicate that using Pareto-dominance information to update the guides, enhances the performance of DVEPSO.

The next section discusses general observations with regards to the performance of the various guide update approaches solving the various DMOOPs.

General Observations

It is interesting to note that p_s-g_r, which ranked the best of all guide update approaches, performed much better solving FDA2 than the modified FDA2 function, FDA2$_{Camara}$. The FDA2 DMOOP was adapted because the POF of the original DMOOP changes from convex to concave for only specific decision variable values (refer to Section 3.2.1). However, it should be noted that the results only indicate that relative to the other guide update approaches, p_s-g_r, performed better for FDA2 than for FDA2$_{Camara}$. Table 9.10 presents the wins and losses of the various guide update approaches for FDA2. When solving FDA2, p_s-g_r obtained the best performance with regards to acc over all n_t-τ_t, the tenth rank with regards to $stab$ and the seventh rank with regards to NS. This lead to an overall rank of four.



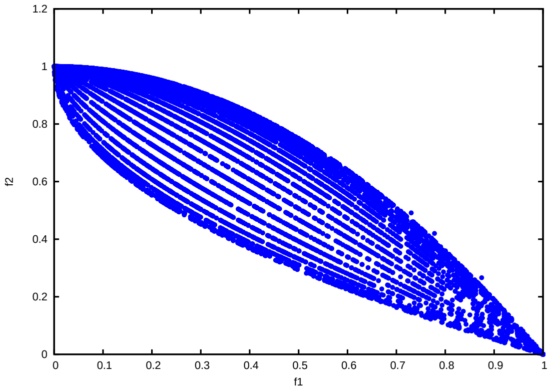
(a) POF^* for dMOP3 (b) POF^* for dMOP3

(c) POF^* for dMOP2 (d) POF^* for dMOP2

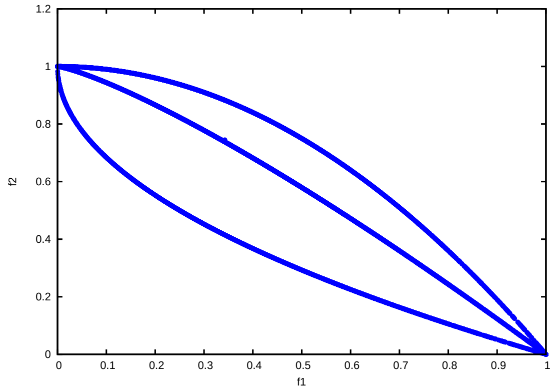
(e) POF^* for dMOP2$_{iso}$ (f) POF^* for dMOP2$_{iso}$

(g) POF^* for dMOP2$_{dec}$ (h) POF^* for dMOP2$_{dec}$

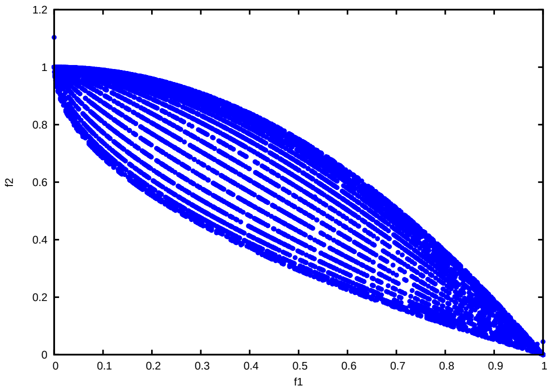
Figure 9.6: POF^* for dMOP functions of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right



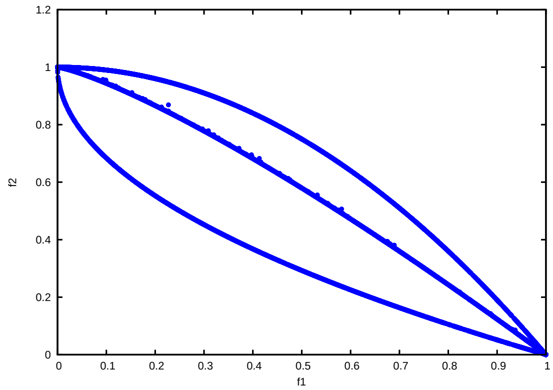
(a) Zoomed in area of POF^* for dMOP2



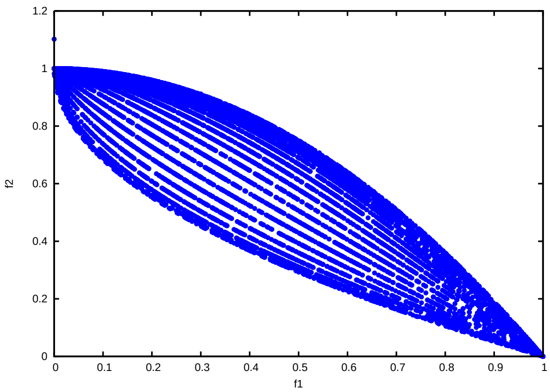
(b) Zoomed in area of POF^* for dMOP2



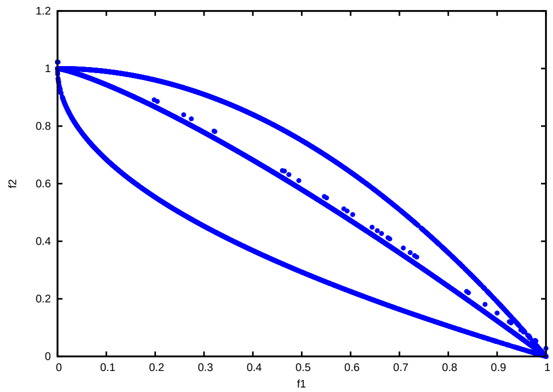
(c) Zoomed in area of POF^* for dMOP2$_{iso}$



(d) Zoomed in area of POF^* for dMOP2$_{iso}$

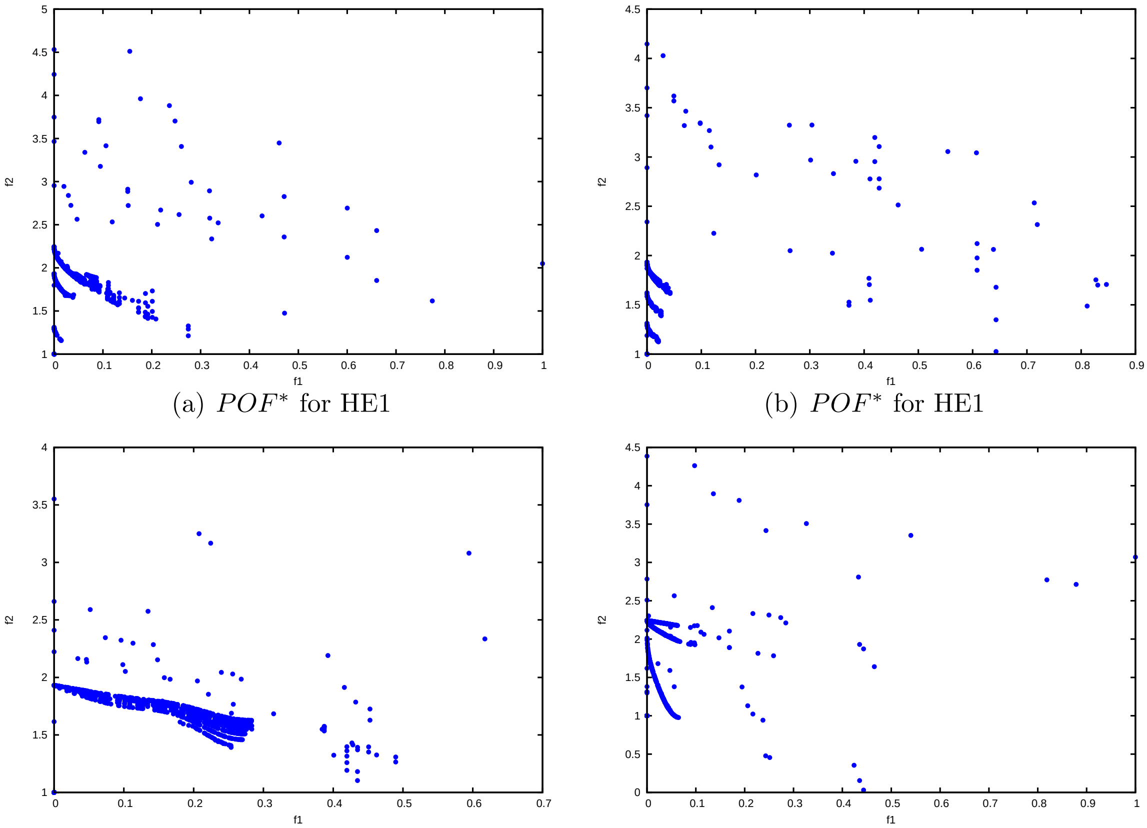


(e) Zoomed in area of POF^* for dMOP2$_{dec}$



(f) Zoomed in area of POF^* for dMOP2$_{dec}$

Figure 9.7: Zoomed in areas of POF^* for dMOP functions of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right



(a) POF^* for HE1 (b) POF^* for HE1

(c) POF^* for HE2 (d) POF^* for HE2

Figure 9.8: POF^* for HE1 and HE2 of DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right

Table 9.10: Wins and Losses of FDA2

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	0	0	1	12	4	9	9	3	3	9	3	12	3	10	3	12
10	10	*acc*	Losses	14	13	13	0	7	3	1	7	7	3	8	0	7	3	7	0
10	10	*acc*	Diff	-14	-13	-12	12	-3	6	8	-4	-4	6	-5	12	-4	7	-4	12
10	10	*acc*	Rank	16	15	14	**1**	8	6	4	9	9	6	13	**1**	9	5	9	**1**
10	25	*acc*	Wins	1	1	0	14	3	8	8	3	4	9	4	8	3	5	3	9
10	25	*acc*	Losses	13	13	15	0	9	1	1	6	6	1	2	1	7	2	6	0
10	25	*acc*	Diff	-12	-12	-15	14	-6	7	7	-3	-2	8	2	7	-4	3	-3	9
10	25	*acc*	Rank	14	14	16	**1**	13	4	4	10	9	3	8	4	12	7	10	2
10	50	*acc*	Wins	1	1	0	10	2	10	13	2	2	10	2	13	2	9	2	14
10	50	*acc*	Losses	13	7	15	3	7	3	0	7	7	3	7	1	7	6	7	0

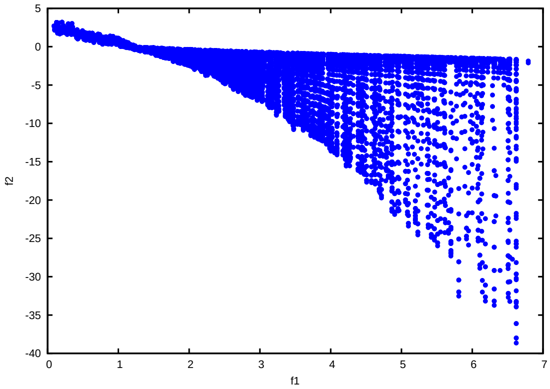
Continued on next page

n_t	τ_t	PM	Results	pbest-gbest combination															
				s-s	s-n	s-d	s-r	n-s	n-n	n-d	n-r	r-s	r-n	r-d	r-r	d-s	d-n	d-d	d-r
10	50	*acc*	Diff	-12	-6	-15	7	-5	7	13	-5	-5	7	-5	12	-5	3	-5	14
10	50	*acc*	Rank	15	14	16	4	8	4	2	8	8	4	8	3	8	7	8	**1**
1	10	*acc*	Wins	2	0	1	15	6	10	5	5	5	12	4	5	5	12	3	5
1	10	*acc*	Losses	13	15	14	0	3	1	3	5	4	1	11	4	4	1	12	4
1	10	*acc*	Diff	-11	-15	-13	15	3	9	2	0	1	11	-7	1	1	11	-9	1
1	10	*acc*	Rank	14	16	15	**1**	5	4	6	11	7	2	12	7	7	2	13	7
20	10	*acc*	Wins	0	2	0	12	4	10	9	4	4	9	4	12	3	9	4	10
20	10	*acc*	Losses	14	13	14	0	7	2	0	7	7	2	7	0	12	3	7	1
20	10	*acc*	Diff	-14	-11	-14	12	-3	8	9	-3	-3	7	-3	12	-9	6	-3	9
20	10	*acc*	Rank	15	14	15	**1**	8	5	3	8	8	6	8	**1**	13	7	8	3
all	all	*acc*	Wins	4	4	2	63	19	47	44	17	18	49	17	50	16	45	15	50
all	all	*acc*	Losses	67	61	71	3	33	10	5	32	31	10	35	6	37	15	39	5
all	all	*acc*	Diff	-63	-57	-69	60	-14	37	39	-15	-13	39	-18	44	-21	30	-24	45
all	all	*acc*	Rank	15	14	16	**1**	9	6	4	10	8	4	11	3	12	7	13	2
10	10	*stab*	Wins	14	7	13	6	6	0	0	6	6	0	6	0	6	0	6	0
10	10	*stab*	Losses	0	1	0	2	2	10	10	2	2	10	2	10	3	10	2	10
10	10	*stab*	Diff	14	6	13	4	4	-10	-10	4	4	-10	4	-10	3	-10	4	-10
10	10	*stab*	Rank	**1**	3	2	4	4	11	11	4	4	11	4	11	10	11	4	11
10	25	*stab*	Wins	7	7	9	0	7	0	0	7	6	0	7	0	7	0	7	0
10	25	*stab*	Losses	0	0	0	9	0	9	9	0	1	8	0	9	0	9	1	9
10	25	*stab*	Diff	7	7	9	-9	7	-9	-9	7	5	-8	7	-9	7	-9	6	-9
10	25	*stab*	Rank	2	2	**1**	11	2	11	11	2	9	10	2	11	2	11	8	11
10	50	*stab*	Wins	7	7	7	0	7	0	0	7	7	0	7	0	7	4	7	0
10	50	*stab*	Losses	0	0	0	9	0	9	10	0	0	10	0	10	0	9	0	10
10	50	*stab*	Diff	7	7	7	-9	7	-9	-10	7	7	-10	7	-10	7	-5	7	-10
10	50	*stab*	Rank	**1**	**1**	**1**	11	**1**	11	13	**1**	**1**	13	**1**	13	**1**	10	**1**	13
1	10	*stab*	Wins	2	4	2	2	2	0	0	2	1	2	1	0	2	0	4	0
1	10	*stab*	Losses	1	0	1	0	0	10	1	0	0	0	0	0	0	10	0	1
1	10	*stab*	Diff	1	4	1	2	2	-10	-1	2	1	2	1	0	2	-10	4	-1
1	10	*stab*	Rank	8	**1**	8	3	3	15	13	3	8	3	8	12	3	15	**1**	13
20	10	*stab*	Wins	14	8	13	3	7	1	0	7	6	1	4	0	7	0	7	0
20	10	*stab*	Losses	0	1	0	7	2	8	12	2	2	8	3	10	2	9	2	10
20	10	*stab*	Diff	14	7	13	-4	5	-7	-12	5	4	-7	1	-10	5	-9	5	-10
20	10	*stab*	Rank	**1**	3	2	10	4	11	16	4	8	11	9	14	4	13	4	14
all	all	*stab*	Wins	44	33	44	11	29	1	0	29	26	3	25	0	29	4	31	0
all	all	*stab*	Losses	1	2	1	27	4	46	42	4	5	36	5	39	5	47	5	40
all	all	*stab*	Diff	43	31	43	-16	25	-45	-42	25	21	-33	20	-39	24	-43	26	-40
all	all	*stab*	Rank	**1**	3	**1**	10	5	16	14	5	8	11	9	12	7	15	4	13
10	10	NS	Wins	0	1	1	3	3	10	13	3	3	10	4	13	3	10	3	13
10	10	NS	Losses	14	14	13	7	6	3	0	6	6	3	6	0	6	3	6	0
10	10	NS	Diff	-14	-13	-12	-4	-3	7	13	-3	-3	7	-2	13	-3	7	-3	13
10	10	NS	Rank	16	15	14	13	8	4	**1**	8	8	4	7	**1**	8	4	8	**1**
10	25	NS	Wins	1	0	0	7	0	2	9	0	0	1	0	9	0	1	0	9
10	25	NS	Losses	5	4	7	0	5	0	0	4	3	0	3	0	4	0	4	0
10	25	NS	Diff	-4	-4	-7	7	-5	2	9	-4	-3	1	-3	9	-4	1	-4	9
10	25	NS	Rank	10	10	16	4	15	5	**1**	10	8	6	8	**1**	10	6	10	**1**
10	50	NS	Wins	0	0	0	10	2	10	13	0	4	10	0	13	0	9	2	14
10	50	NS	Losses	8	7	10	3	7	3	0	7	7	3	10	1	8	6	7	0
10	50	NS	Diff	-8	-7	-10	7	-5	7	13	-7	-3	7	-10	12	-8	3	-5	14
10	50	NS	Rank	13	11	15	4	9	4	2	11	8	4	15	3	13	7	9	**1**

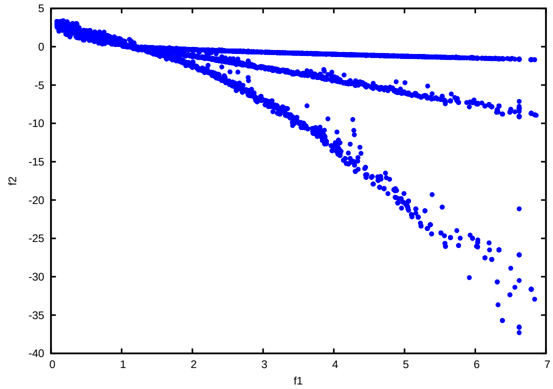
Continued on next page

$\mathbf{n_t}$	τ_t	PM	Results	pbest-gbest combination															
				s-s	s-n	s-d	s-r	n-s	n-n	n-d	n-r	r-s	r-n	r-d	r-r	d-s	d-n	d-d	d-r
1	10	NS	Wins	1	0	1	2	4	10	11	4	4	5	4	10	4	9	4	11
1	10	NS	Losses	13	15	12	12	5	0	0	5	4	2	5	0	5	0	6	0
1	10	NS	Diff	-12	-15	-11	-10	-1	10	11	-1	0	3	-1	10	-1	9	-2	11
1	10	NS	Rank	15	16	14	13	8	3	**1**	8	7	6	8	3	8	5	12	**1**
20	10	NS	Wins	0	0	0	6	1	8	13	4	3	8	3	13	2	10	2	13
20	10	NS	Losses	13	10	12	3	7	4	0	6	4	3	6	0	7	3	8	0
20	10	NS	Diff	-13	-10	-12	3	-6	4	13	-2	-1	5	-3	13	-5	7	-6	13
20	10	NS	Rank	16	14	15	7	12	6	**1**	9	8	5	10	**1**	11	4	12	**1**
all	all	NS	Wins	2	1	2	28	10	40	59	11	14	34	11	58	9	39	11	60
all	all	NS	Losses	53	50	54	25	30	10	0	28	24	11	30	1	30	12	31	0
all	all	NS	Diff	-51	-49	-52	3	-20	30	59	-17	-10	23	-19	57	-21	27	-20	60
all	all	NS	Rank	15	14	16	7	11	4	2	9	8	6	10	3	13	5	11	**1**
10	10	all	Wins	14	8	15	21	13	19	22	12	12	19	13	25	12	20	12	25
10	10	all	Losses	28	28	26	9	15	16	11	15	15	16	16	10	16	16	15	10
10	10	all	Diff	-14	-20	-11	12	-2	3	11	-3	-3	3	-3	15	-4	4	-3	15
10	10	all	Rank	15	16	14	3	8	6	4	9	9	6	9	**1**	13	5	9	**1**
10	25	all	Wins	9	8	9	21	10	10	17	10	10	10	11	17	10	6	10	18
10	25	all	Losses	18	17	22	9	14	10	10	10	10	9	5	10	11	11	11	9
10	25	all	Diff	-9	-9	-13	12	-4	0	7	0	0	1	6	7	-1	-5	-1	9
10	25	all	Rank	14	14	16	**1**	12	7	3	7	7	6	5	3	10	13	10	2
10	50	all	Wins	8	8	7	20	11	20	26	9	13	20	9	26	9	22	11	28
10	50	all	Losses	21	14	25	15	14	15	10	14	14	16	17	12	15	21	14	10
10	50	all	Diff	-13	-6	-18	5	-3	5	16	-5	-1	4	-8	14	-6	1	-3	18
10	50	all	Rank	15	12	16	4	9	4	2	11	8	6	14	3	12	7	9	**1**
1	10	all	Wins	5	4	4	19	12	20	16	11	10	19	9	15	11	21	11	16
1	10	all	Losses	27	30	27	12	8	11	4	10	8	3	16	4	9	11	18	5
1	10	all	Diff	-22	-26	-23	7	4	9	12	1	2	16	-7	11	2	10	-7	11
1	10	all	Rank	14	16	15	7	8	6	2	11	9	**1**	12	3	9	5	12	3
20	10	all	Wins	14	10	13	21	12	19	22	15	13	18	11	25	12	19	13	23
20	10	all	Losses	27	24	26	10	16	14	12	15	13	13	16	10	21	15	17	11
20	10	all	Diff	-13	-14	-13	11	-4	5	10	0	0	5	-5	15	-9	4	-4	12
20	10	all	Rank	14	16	14	3	10	5	4	8	8	5	12	**1**	13	7	10	2
all	all	all	Wins	50	38	48	102	58	88	103	57	58	86	53	108	54	88	57	110
all	all	all	Losses	121	113	126	55	67	66	47	64	60	57	70	46	72	74	75	45
all	all	all	Diff	-71	-75	-78	47	-9	22	56	-7	-2	29	-17	62	-18	14	-18	65
all	all	all	Rank	14	15	16	4	10	6	3	9	8	5	11	2	12	7	12	**1**

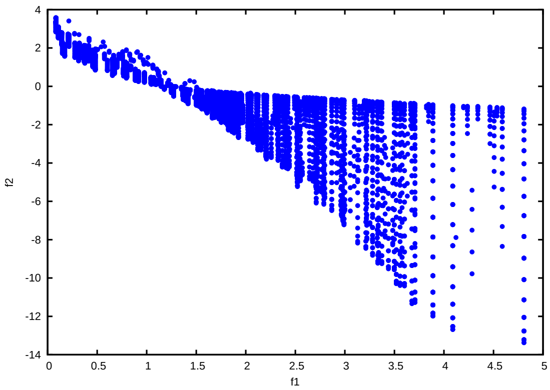
The wins and losses of the various guide update approaches for FDA2$_{Camara}$ are presented in Table 9.11. When solving FDA2$_{Camara}$, there was no statistical significant difference between the performance of the various guide update approaches with regards to *acc* for $n_t = 10$ and $\tau_t = 10$, $n_t = 10$ and $n_t = 25$, $n_t = 10$ and $\tau_t = 50$, and $n_t = 20$ and $\tau_t = 10$. With regards to *stab*, there was no statistical significant difference for $n_t = 10$



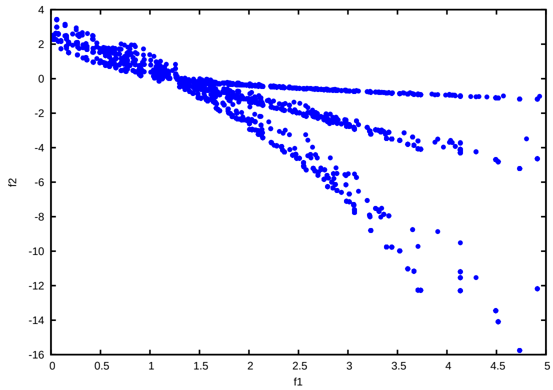
(a) POF^* for HE6



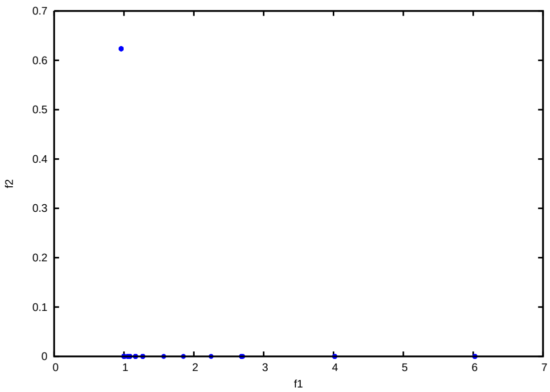
(b) POF^* for HE6



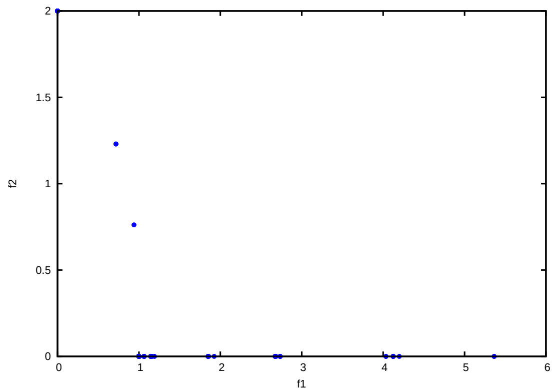
(c) POF^* for HE7



(d) POF^* for HE7



(e) POF^* for HE9



(f) POF^* for HE9

Figure 9.9: POF^* of HE6, HE7 and HE9 DVEPSO using p_s-g_r for $n_t = 10$ and $\tau_t = 10$ on the left and for $n_t = 1$ and $\tau_t = 10$ on the right

and $\tau_t = 10$, and $n_t = 10$ and $n_t = 25$. For $n_t = 10$ and $\tau_t = 25$, and $n_t = 20$ and $\tau_t = 10$, there was no statistical significant difference between the performance of the various guide update approaches with regards to NS. Over all performance measures for $n_t = 10$ and $\tau_t = 25$, there was no statistical significant difference between the performance of the various guide update approaches. Therefore, even though p_s-g_r performed quite poor for FDA2$_{Camara}$, there was no statistical significant difference between the performance of the various guide update approaches when solving FDA2$_{Camara}$. A similar trend was observed for FDA3 and FDA3$_{Camara}$.

Table 9.11: Wins and Losses of FDA2$_{Camara}$

$\mathbf{n_t}$	$\boldsymbol{\tau_t}$	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
10	10	*acc*	Losses	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*acc*	Diff	0	-13	0	1	1	1	1	1	1	1	1	1	1	1	1	1
10	10	*acc*	Rank	14	16	14	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
1	10	*acc*	Wins	0	0	0	2	3	3	3	3	3	3	3	3	3	3	3	3
1	10	*acc*	Losses	13	13	12	0	0	0	0	0	0	0	0	0	0	0	0	0
1	10	*acc*	Diff	-13	-13	-12	2	3	3	3	3	3	3	3	3	3	3	3	3
1	10	*acc*	Rank	15	15	14	13	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
all	all	*acc*	Wins	0	0	0	3	4	4	4	4	4	4	4	4	4	4	4	4
all	all	*acc*	Losses	13	26	12	0	0	0	0	0	0	0	0	0	0	0	0	0
all	all	*acc*	Diff	-13	-26	-12	3	4	4	4	4	4	4	4	4	4	4	4	4
all	all	*acc*	Rank	15	16	14	13	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
10	10	*stab*	Wins	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*stab*	Losses	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*stab*	Diff	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*stab*	Rank	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
10	25	*stab*	Wins	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	25	*stab*	Losses	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	25	*stab*	Diff	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	25	*stab*	Rank	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
10	50	*stab*	Wins	0	0	1	0	1	4	4	3	3	1	0	0	3	1	0	0
10	50	*stab*	Losses	9	5	0	5	0	0	0	0	0	0	2	0	0	0	0	0
10	50	*stab*	Diff	-9	-5	1	-5	1	4	4	3	3	1	-2	0	3	1	0	0
10	50	*stab*	Rank	16	14	6	14	6	**1**	**1**	3	3	6	13	10	3	6	10	10
1	10	*stab*	Wins	0	0	0	2	3	3	3	3	3	3	3	3	3	3	3	3
1	10	*stab*	Losses	13	13	12	0	0	0	0	0	0	0	0	0	0	0	0	0
1	10	*stab*	Diff	-13	-13	-12	2	3	3	3	3	3	3	3	3	3	3	3	3
1	10	*stab*	Rank	15	15	14	13	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
20	10	*stab*	Wins	0	0	0	0	0	1	0	0	0	2	0	0	0	2	1	0
20	10	*stab*	Losses	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	10	*stab*	Diff	-4	-2	0	0	0	1	0	0	0	2	0	0	0	2	1	0
20	10	*stab*	Rank	16	15	5	5	5	3	5	5	5	**1**	5	5	5	**1**	3	5
all	all	*stab*	Wins	0	0	1	2	4	8	7	6	6	6	3	3	6	6	4	3
all	all	*stab*	Losses	26	20	12	5	0	0	0	0	0	0	2	0	0	0	0	0

Continued on next page

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
all	all	*stab*	Diff	-26	-20	-11	-3	4	8	7	6	6	6	1	3	6	6	4	3
all	all	*stab*	Rank	16	15	14	13	8	**1**	2	3	3	3	12	10	3	3	8	10
10	10	*NS*	Wins	0	0	0	0	2	1	2	2	2	2	0	1	2	1	2	2
10	10	*NS*	Losses	0	8	0	11	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*NS*	Diff	0	-8	0	-11	2	1	2	2	2	2	0	1	2	1	2	2
10	10	*NS*	Rank	12	15	12	16	**1**	9	**1**	**1**	**1**	**1**	12	9	**1**	9	**1**	**1**
10	50	*NS*	Wins	0	0	0	10	2	10	13	0	4	10	0	13	0	9	2	14
10	50	*NS*	Losses	8	7	10	3	7	3	0	7	7	3	10	1	8	6	7	0
10	50	*NS*	Diff	-8	-7	-10	7	-5	7	13	-7	-3	7	-10	12	-8	3	-5	14
10	50	*NS*	Rank	13	11	15	4	9	4	2	11	8	4	15	3	13	7	9	**1**
1	10	*NS*	Wins	0	2	0	0	4	4	4	4	4	4	4	4	4	4	4	4
1	10	*NS*	Losses	12	12	13	13	0	0	0	0	0	0	0	0	0	0	0	0
1	10	*NS*	Diff	-12	-10	-13	-13	4	4	4	4	4	4	4	4	4	4	4	4
1	10	*NS*	Rank	14	13	15	15	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
all	all	*NS*	Wins	0	2	0	10	8	15	19	6	10	16	4	18	6	14	8	20
all	all	*NS*	Losses	20	27	23	27	7	3	0	7	7	3	10	1	8	6	7	0
all	all	*NS*	Diff	-20	-25	-23	-17	1	12	19	-1	3	13	-6	17	-2	8	1	20
all	all	*NS*	Rank	14	16	15	13	8	5	2	10	7	4	12	3	11	6	8	**1**
10	10	all	Wins	0	0	0	1	3	2	3	3	3	3	1	2	3	2	3	3
10	10	all	Losses	0	21	0	11	0	0	0	0	0	0	0	0	0	0	0	0
10	10	all	Diff	0	-21	0	-10	3	2	3	3	3	3	1	2	3	2	3	3
10	10	all	Rank	13	16	13	15	**1**	9	**1**	**1**	**1**	**1**	12	9	**1**	9	**1**	**1**
10	50	all	Wins	0	0	1	10	3	14	17	3	7	11	0	13	3	10	2	14
10	50	all	Losses	17	12	10	8	7	3	0	7	7	3	12	1	8	6	7	0
10	50	all	Diff	-17	-12	-9	2	-4	11	17	-4	0	8	-12	12	-5	4	-5	14
10	50	all	Rank	16	14	13	7	9	4	**1**	9	8	5	14	3	11	6	11	2
1	10	all	Wins	0	2	0	4	10	10	10	10	10	10	10	10	10	10	10	10
1	10	all	Losses	38	38	37	13	0	0	0	0	0	0	0	0	0	0	0	0
1	10	all	Diff	-38	-36	-37	-9	10	10	10	10	10	10	10	10	10	10	10	10
1	10	all	Rank	16	14	15	13	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
20	10	all	Wins	0	0	0	0	0	1	0	0	0	2	0	0	0	2	1	0
20	10	all	Losses	4	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	10	all	Diff	-4	-2	0	0	0	1	0	0	0	2	0	0	0	2	1	0
20	10	all	Rank	16	15	5	5	5	3	5	5	5	**1**	5	5	5	**1**	3	5
all	all	all	Wins	0	2	1	15	16	27	30	16	20	26	11	25	16	24	16	27
all	all	all	Losses	59	73	47	32	7	3	0	7	7	3	12	1	8	6	7	0
all	all	all	Diff	-59	-71	-46	-17	9	24	30	9	13	23	-1	24	8	18	9	27
all	all	all	Rank	15	16	14	13	8	3	**1**	8	7	5	12	3	11	6	8	2

The average performance measure values at each iteration just before a change in the environment occurred obtained by DVEPSO using either p_s-g_s or p_s-g_r guide update approaches, are illustrated in Figures 9.10 to 9.12. In Figures 9.10 to 9.12 the values obtained by p_s-g_s and p_s-g_r are illustrated with a magenta triangle and blue circle respectively. The wins and losses of Table 9.11 are calculated based on these performance measure values. Similar figures for the other DMOOPs can be found in Appendix D.

Figure 9.10 shows that p_s-g_r outperformed p_s-g_s with regards to *acc* for all n_t-τ_t combinations. This is confirmed in Table 9.11 where p_s-g_r obtained the highest rank of all guide update approaches for the wins and losses with regards to *acc*.

Figure 9.11 indicates that p_s-g_s outperformed p_s-g_r with regards to *stab* for all n_t-τ_t combinations. Table 9.11 confirms this observation, since p_s-g_s obtained the highest rank of all guide update approaches for the wins and losses with regards to *stab*.

Figure 9.12 shows that p_s-g_r outperformed p_s-g_s with regards to NS for all n_t-τ_t combinations. This is confirmed in Table 9.11 where p_s-g_r obtained a higher rank than p_s-g_r for the wins and losses with regards to NS.

When solving DMOOPs with discontinuous POFs, the p_s combinations outperformed the other guide update approaches. Table 9.12 presents the wins and losses for HE1 of the various guide update approaches. A similar trend was observed for HE2.

Table 9.12: Wins and Losses of HE1

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	15	12	14	13	6	1	3	9	3	6	8	1	8	0	8	2
10	10	*acc*	Losses	0	3	1	2	8	13	10	4	10	8	5	12	4	15	4	10
10	10	*acc*	Diff	15	9	13	11	-2	-12	-7	5	-7	-2	3	-11	4	-15	4	-8
10	10	*acc*	Rank	**1**	4	2	3	9	15	11	5	11	9	8	14	6	16	6	13
10	25	*acc*	Wins	14	12	14	13	4	1	3	7	6	0	7	11	2	2	2	7
10	25	*acc*	Losses	0	3	0	2	8	14	9	5	5	15	5	4	9	11	10	5
10	25	*acc*	Diff	14	9	14	11	-4	-13	-6	2	1	-15	2	7	-7	-9	-8	2
10	25	*acc*	Rank	**1**	4	**1**	3	10	15	11	6	9	16	6	5	12	14	13	6
10	50	*acc*	Wins	14	12	13	14	5	0	0	4	5	1	1	0	11	0	3	6
10	50	*acc*	Losses	0	3	2	0	5	12	10	5	5	8	6	9	4	10	5	5
10	50	*acc*	Diff	14	9	11	14	0	-12	-10	-1	0	-7	-5	-9	7	-10	-2	1
10	50	*acc*	Rank	**1**	4	3	**1**	7	16	14	9	7	12	11	13	5	14	10	6
1	10	*acc*	Wins	14	12	15	13	7	0	6	4	7	0	5	3	5	0	4	6
1	10	*acc*	Losses	1	3	0	2	4	13	4	8	4	13	6	12	4	13	10	4
1	10	*acc*	Diff	13	9	15	11	3	-13	2	-4	3	-13	-1	-9	1	-13	-6	2
1	10	*acc*	Rank	2	4	**1**	3	5	14	7	11	5	14	10	13	9	14	12	7
20	10	*acc*	Wins	14	12	14	13	7	3	4	9	4	0	9	1	7	0	9	3
20	10	*acc*	Losses	0	3	0	2	7	11	9	4	9	13	4	13	7	14	4	9
20	10	*acc*	Diff	14	9	14	11	0	-8	-5	5	-5	-13	5	-12	0	-14	5	-6
20	10	*acc*	Rank	**1**	4	**1**	3	8	13	10	5	10	15	5	14	8	16	5	12
all	all	*acc*	Wins	71	60	70	66	29	5	16	33	25	7	30	16	33	2	26	24
all	all	*acc*	Losses	1	15	3	8	32	63	42	26	33	57	26	50	28	63	33	33
all	all	*acc*	Diff	70	45	67	58	-3	-58	-26	7	-8	-50	4	-34	5	-61	-7	-9
all	all	*acc*	Rank	**1**	4	2	3	8	15	12	5	10	14	7	13	6	16	9	11

Continued on next page

n_t	τ_t	PM	Results	pbest-gbest combination															
				s-s	s-n	s-d	s-r	n-s	n-n	n-d	n-r	r-s	r-n	r-d	r-r	d-s	d-n	d-d	d-r
10	10	*stab*	Wins	12	12	12	13	0	1	0	0	0	0	1	0	0	1	0	0
10	10	*stab*	Losses	0	1	0	0	4	4	4	7	4	4	4	4	4	4	4	4
10	10	*stab*	Diff	12	11	12	13	-4	-3	-4	-7	-4	-4	-3	-4	-4	-3	-4	-4
10	10	*stab*	Rank	2	4	2	**1**	8	5	8	16	8	8	5	8	8	5	8	8
10	25	*stab*	Wins	12	12	12	12	1	0	1	0	0	1	0	0	0	1	0	0
10	25	*stab*	Losses	0	0	0	0	4	4	4	8	4	4	4	4	4	4	4	4
10	25	*stab*	Diff	12	12	12	12	-3	-4	-3	-8	-4	-3	-4	-4	-4	-3	-4	-4
10	25	*stab*	Rank	**1**	**1**	**1**	**1**	5	9	5	16	9	5	9	9	9	5	9	9
10	50	*stab*	Wins	12	12	12	12	0	0	0	0	0	0	0	0	0	0	0	0
10	50	*stab*	Losses	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4
10	50	*stab*	Diff	12	12	12	12	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4
10	50	*stab*	Rank	**1**	**1**	**1**	**1**	5	5	5	5	5	5	5	5	5	5	5	5
1	10	*stab*	Wins	15	12	13	13	0	0	3	0	0	1	0	3	1	0	1	1
1	10	*stab*	Losses	0	3	1	1	6	4	4	10	6	4	4	4	4	4	4	4
1	10	*stab*	Diff	15	9	12	12	-6	-4	-1	-10	-6	-3	-4	-1	-3	-4	-3	-3
1	10	*stab*	Rank	**1**	4	2	2	14	11	5	16	14	7	11	5	7	11	7	7
20	10	*stab*	Wins	12	12	12	12	0	0	0	0	0	0	0	0	0	0	0	0
20	10	*stab*	Losses	0	0	0	0	4	4	4	4	4	4	4	4	4	4	4	4
20	10	*stab*	Diff	12	12	12	12	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4
20	10	*stab*	Rank	**1**	**1**	**1**	**1**	5	5	5	5	5	5	5	5	5	5	5	5
all	all	*stab*	Wins	63	60	61	62	1	1	4	0	0	2	1	3	1	2	1	1
all	all	*stab*	Losses	0	4	1	1	22	20	20	33	22	20	20	20	20	20	20	20
all	all	*stab*	Diff	63	56	60	61	-21	-19	-16	-33	-22	-18	-19	-17	-19	-18	-19	-19
all	all	*stab*	Rank	**1**	4	3	2	14	9	5	16	15	7	9	6	9	7	9	9
10	10	*NS*	Wins	15	13	14	12	1	1	8	0	1	3	4	9	0	8	0	0
10	10	*NS*	Losses	0	2	1	3	8	7	4	12	7	7	5	4	7	4	9	9
10	10	*NS*	Diff	15	11	13	9	-7	-6	4	-12	-6	-4	-1	5	-7	4	-9	-9
10	10	*NS*	Rank	**1**	3	2	4	12	10	6	16	10	9	8	5	12	6	14	14
10	25	*NS*	Wins	13	15	13	12	0	2	2	2	0	6	0	0	5	4	3	0
10	25	*NS*	Losses	1	0	2	2	6	4	4	6	6	4	8	11	4	4	4	11
10	25	*NS*	Diff	12	15	11	10	-6	-2	-2	-4	-6	2	-8	-11	1	0	-1	-11
10	25	*NS*	Rank	2	**1**	3	4	12	9	9	11	12	5	14	15	6	7	8	15
10	50	*NS*	Wins	14	12	12	15	0	5	0	0	0	4	0	0	0	1	0	0
10	50	*NS*	Losses	1	2	2	0	4	4	6	4	7	4	4	5	6	4	4	6
10	50	*NS*	Diff	13	10	10	15	-4	1	-6	-4	-7	0	-4	-5	-6	-3	-4	-6
10	50	*NS*	Rank	2	3	3	**1**	8	5	13	8	16	6	8	12	13	7	8	13
1	10	*NS*	Wins	13	13	15	12	1	9	4	3	0	8	3	4	1	9	3	1
1	10	*NS*	Losses	1	1	0	3	12	4	7	7	15	4	7	6	9	4	7	12
1	10	*NS*	Diff	12	12	15	9	-11	5	-3	-4	-15	4	-4	-2	-8	5	-4	-11
1	10	*NS*	Rank	2	2	**1**	4	14	5	9	10	16	7	10	8	13	5	10	14
20	10	*NS*	Wins	15	13	14	12	3	6	0	0	8	7	2	3	3	8	0	1
20	10	*NS*	Losses	0	2	1	3	8	4	13	12	4	4	8	7	6	4	11	8
20	10	*NS*	Diff	15	11	13	9	-5	2	-13	-12	4	3	-6	-4	-3	4	-11	-7
20	10	*NS*	Rank	**1**	3	2	4	11	8	16	15	5	7	12	10	9	5	14	13
all	all	*NS*	Wins	70	66	68	63	5	23	14	5	9	28	9	16	9	30	6	2
all	all	*NS*	Losses	3	7	6	11	38	23	34	41	39	23	32	33	32	20	35	46
all	all	*NS*	Diff	67	59	62	52	-33	0	-20	-36	-30	5	-23	-17	-23	10	-29	-44
all	all	*NS*	Rank	**1**	3	2	4	14	7	9	15	13	6	10	8	10	5	12	16
10	10	all	Wins	42	37	40	38	7	3	11	9	4	9	13	10	8	9	8	2
10	10	all	Losses	0	6	2	5	20	24	18	23	21	19	14	20	15	23	17	23

Continued on next page

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	all	Diff	42	31	38	33	-13	-21	-7	-14	-17	-10	-1	-10	-7	-14	-9	-21
10	10	all	Rank	**1**	4	2	3	11	15	6	12	14	9	5	9	6	12	8	15
10	25	all	Wins	39	39	39	37	5	3	6	9	6	7	7	11	7	7	5	7
10	25	all	Losses	1	3	2	4	18	22	17	19	15	23	17	19	17	19	18	20
10	25	all	Diff	38	36	37	33	-13	-19	-11	-10	-9	-16	-10	-8	-10	-12	-13	-13
10	25	all	Rank	**1**	3	2	4	12	16	10	7	6	15	7	5	7	11	12	12
10	50	all	Wins	40	36	37	41	5	5	0	4	5	5	1	0	11	1	3	6
10	50	all	Losses	1	5	4	0	13	20	20	13	16	16	14	18	14	18	13	15
10	50	all	Diff	39	31	33	41	-8	-15	-20	-9	-11	-11	-13	-18	-3	-17	-10	-9
10	50	all	Rank	2	4	3	**1**	6	13	16	7	10	10	12	15	5	14	9	7
1	10	all	Wins	42	37	43	38	8	9	13	7	7	9	8	10	7	9	8	8
1	10	all	Losses	2	7	1	6	22	21	15	25	25	21	17	22	17	21	21	20
1	10	all	Diff	40	30	42	32	-14	-12	-2	-18	-18	-12	-9	-12	-10	-12	-13	-12
1	10	all	Rank	2	4	**1**	3	14	8	5	15	15	8	6	8	7	8	13	8
20	10	all	Wins	41	37	40	37	10	9	4	9	12	7	11	4	10	8	9	4
20	10	all	Losses	0	5	1	5	19	19	26	20	17	21	16	24	17	22	19	21
20	10	all	Diff	41	32	39	32	-9	-10	-22	-11	-5	-14	-5	-20	-7	-14	-10	-17
20	10	all	Rank	**1**	3	2	3	8	9	16	11	5	12	5	15	7	12	9	14
all	all	all	Wins	204	186	199	191	35	29	34	38	34	37	40	35	43	34	33	27
all	all	all	Losses	4	26	10	20	92	106	96	100	94	100	78	103	80	103	88	99
all	all	all	Diff	200	160	189	171	-57	-77	-62	-62	-60	-63	-38	-68	-37	-69	-55	-72
all	all	all	Rank	**1**	4	2	3	8	16	10	10	9	12	6	13	5	14	7	15

When solving DMOOPs where each decision variable has its own POS and the POS is a non-linear function, the p_s combinations outperformed the other guide update approaches. This was observed for HE6, HE7 and HE9. The only exception is with p_s-g_s, that performed poorly for HE7. The wins and losses obtained by the various guide update approaches for HE7 is presented in Table 9.13.

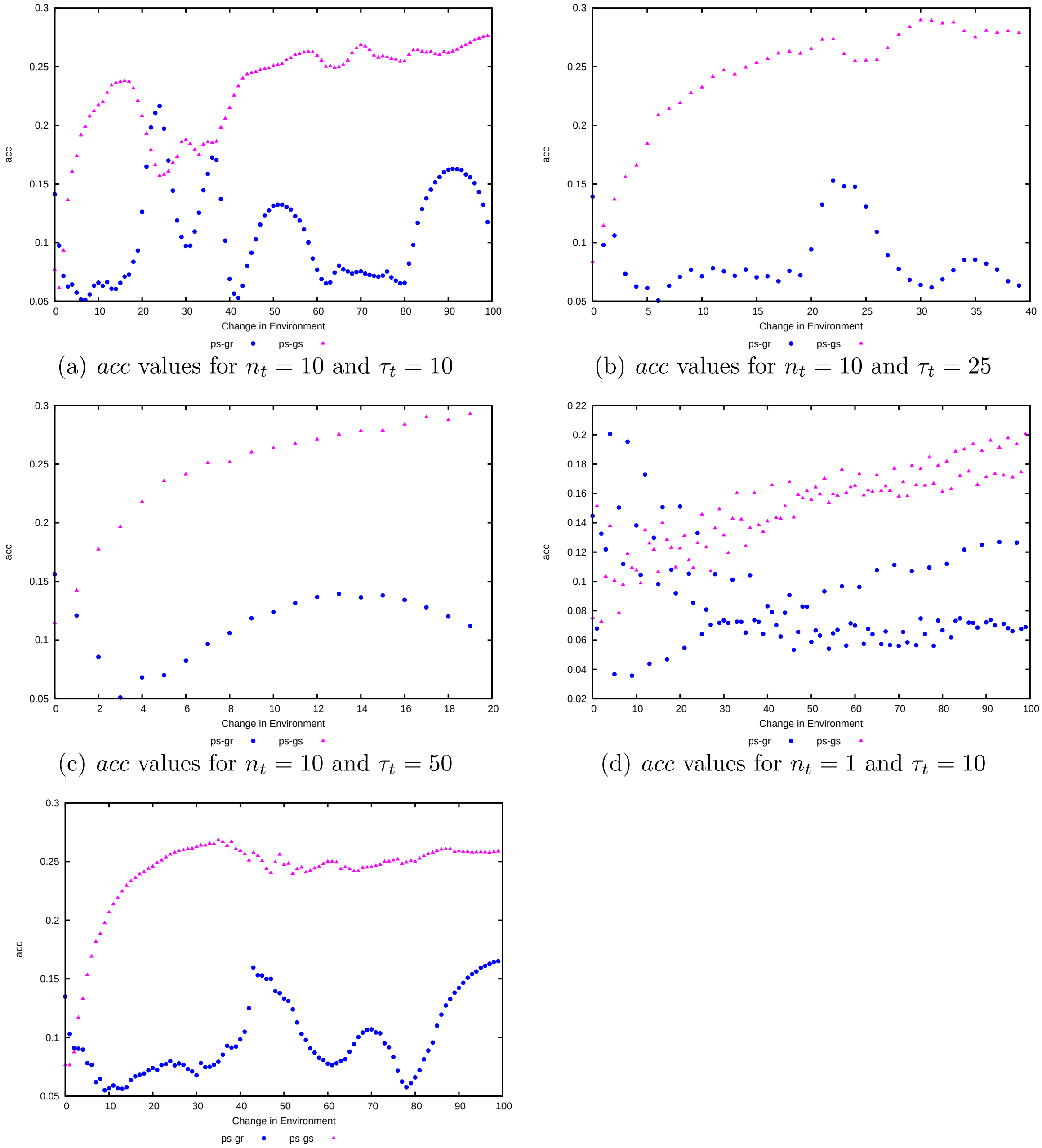
Table 9.13: Wins and Losses of HE7

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
1	10	*acc*	Wins	10	0	0	0	12	0	0	0	12	0	0	0	12	0	0	0
1	10	*acc*	Losses	0	4	4	4	0	4	4	3	0	4	4	4	0	4	4	3
1	10	*acc*	Diff	10	-4	-4	-4	12	-4	-4	-3	12	-4	-4	-4	12	-4	-4	-3
1	10	*acc*	Rank	4	7	7	7	**1**	7	7	5	**1**	7	7	7	**1**	7	7	5
all	all	*acc*	Wins	10	0	0	0	12	0	0	0	12	0	0	0	12	0	0	0
all	all	*acc*	Losses	0	4	4	4	0	4	4	3	0	4	4	4	0	4	4	3
all	all	*acc*	Diff	10	-4	-4	-4	12	-4	-4	-3	12	-4	-4	-4	12	-4	-4	-3
all	all	*acc*	Rank	4	7	7	7	**1**	7	7	5	**1**	7	7	7	**1**	7	7	5
10	10	*NS*	Wins	3	13	13	13	0	4	4	4	0	4	4	5	0	4	4	4
10	10	*NS*	Losses	12	0	0	0	13	4	3	3	13	3	3	3	13	3	3	3
10	10	*NS*	Diff	-9	13	13	13	-13	0	1	1	-13	1	1	2	-13	1	1	1
10	10	*NS*	Rank	13	**1**	**1**	**1**	14	12	5	5	14	5	5	4	14	5	5	5
10	25	*NS*	Wins	1	7	5	7	0	4	4	4	0	4	4	3	0	3	4	3
10	25	*NS*	Losses	9	0	0	0	12	0	0	0	12	0	0	3	13	2	0	2

Continued on next page

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	25	NS	Diff	-8	7	5	7	-12	4	4	4	-12	4	4	0	-13	1	4	1
10	25	NS	Rank	13	**1**	3	**1**	14	4	4	4	14	4	4	12	16	10	4	10
10	50	NS	Wins	3	8	5	9	0	3	3	3	0	4	4	3	0	3	3	3
10	50	NS	Losses	5	0	0	0	13	0	0	3	13	0	0	2	13	2	2	1
10	50	NS	Diff	-2	8	5	9	-13	3	3	0	-13	4	4	1	-13	1	1	2
10	50	NS	Rank	13	2	3	**1**	14	6	6	12	14	4	4	9	14	9	9	8
1	10	NS	Wins	3	13	13	13	0	4	4	4	0	4	4	4	0	4	4	4
1	10	NS	Losses	12	0	0	0	13	3	3	3	13	3	3	3	13	3	3	3
1	10	NS	Diff	-9	13	13	13	-13	1	1	1	-13	1	1	1	-13	1	1	1
1	10	NS	Rank	13	**1**	**1**	**1**	14	4	4	4	14	4	4	4	14	4	4	4
20	10	NS	Wins	3	13	13	13	0	4	4	4	0	4	4	4	1	4	4	4
20	10	NS	Losses	12	0	0	0	14	3	3	3	13	3	3	3	13	3	3	3
20	10	NS	Diff	-9	13	13	13	-14	1	1	1	-13	1	1	1	-12	1	1	1
20	10	NS	Rank	13	**1**	**1**	**1**	16	4	4	4	15	4	4	4	14	4	4	4
all	all	NS	Wins	13	54	49	55	0	19	19	19	0	20	20	19	1	18	19	18
all	all	NS	Losses	50	0	0	0	65	10	9	12	64	9	9	14	65	13	11	12
all	all	NS	Diff	-37	54	49	55	-65	9	10	7	-64	11	11	5	-64	5	8	6
all	all	NS	Rank	13	2	3	**1**	16	7	6	9	14	4	4	11	14	11	8	10
10	10	all	Wins	3	13	13	13	0	4	4	4	0	4	4	5	0	4	4	4
10	10	all	Losses	12	0	0	0	13	4	3	3	13	3	3	3	13	3	3	3
10	10	all	Diff	-9	13	13	13	-13	0	1	1	-13	1	1	2	-13	1	1	1
10	10	all	Rank	13	**1**	**1**	**1**	14	12	5	5	14	5	5	4	14	5	5	5
10	25	all	Wins	1	7	5	7	0	4	4	4	0	4	4	3	0	3	4	3
10	25	all	Losses	9	0	0	0	12	0	0	0	12	0	0	3	13	2	0	2
10	25	all	Diff	-8	7	5	7	-12	4	4	4	-12	4	4	0	-13	1	4	1
10	25	all	Rank	13	**1**	3	**1**	14	4	4	4	14	4	4	12	16	10	4	10
10	50	all	Wins	3	8	5	9	0	3	3	3	0	4	4	3	0	3	3	3
10	50	all	Losses	5	0	0	0	13	0	0	3	13	0	0	2	13	2	2	1
10	50	all	Diff	-2	8	5	9	-13	3	3	0	-13	4	4	1	-13	1	1	2
10	50	all	Rank	13	2	3	**1**	14	6	6	12	14	4	4	9	14	9	9	8
1	10	all	Wins	13	13	13	13	12	4	4	4	12	4	4	4	12	4	4	4
1	10	all	Losses	12	4	4	4	13	7	7	6	13	7	7	7	13	7	7	6
1	10	all	Diff	1	9	9	9	-1	-3	-3	-2	-1	-3	-3	-3	-1	-3	-3	-2
1	10	all	Rank	4	**1**	**1**	**1**	5	10	10	8	5	10	10	10	5	10	10	8
20	10	all	Wins	3	13	13	13	0	4	4	4	0	4	4	4	1	4	4	4
20	10	all	Losses	12	0	0	0	14	3	3	3	13	3	3	3	13	3	3	3
20	10	all	Diff	-9	13	13	13	-14	1	1	1	-13	1	1	1	-12	1	1	1
20	10	all	Rank	13	**1**	**1**	**1**	16	4	4	4	15	4	4	4	14	4	4	4
all	all	all	Wins	23	54	49	55	12	19	19	19	12	20	20	19	13	18	19	18
all	all	all	Losses	50	4	4	4	65	14	13	15	64	13	13	18	65	17	15	15
all	all	all	Diff	-27	50	45	51	-53	5	6	4	-52	7	7	1	-52	1	4	3
all	all	all	Rank	13	2	3	**1**	16	7	6	8	14	4	4	11	14	11	8	10

For dMOP2 it was interesting to observe the difference in performance of the p_s combinations when solving dMOP2, dMOP2 with an isolated POF, dMOP2$_{iso}$, and



(a) acc values for $n_t = 10$ and $\tau_t = 10$

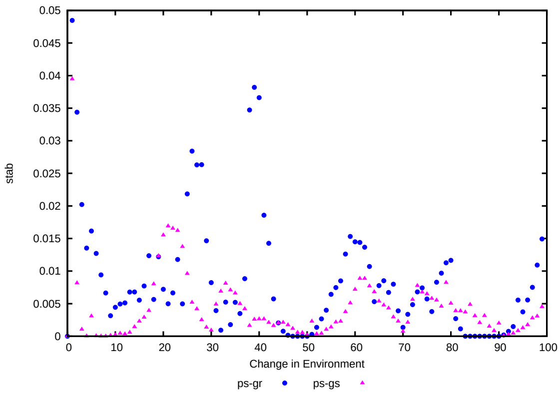
(b) acc values for $n_t = 10$ and $\tau_t = 25$

(c) acc values for $n_t = 10$ and $\tau_t = 50$

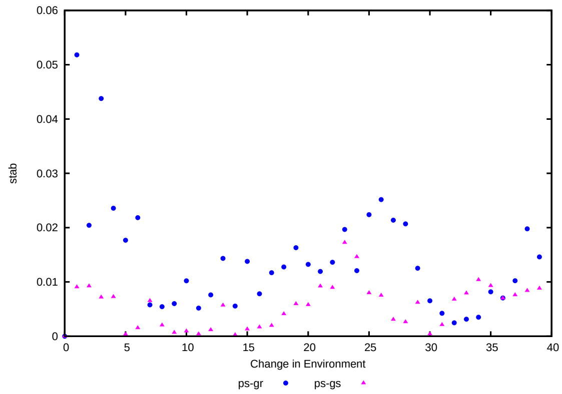
(d) acc values for $n_t = 1$ and $\tau_t = 10$

(e) acc values for $n_t = 20$ and $\tau_t = 10$

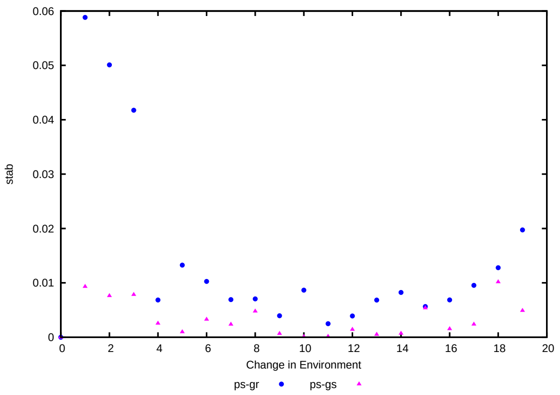
Figure 9.10: Average values of acc obtained by DVEPSO using either p_s-g_s or p_s-g_r solving FDA2



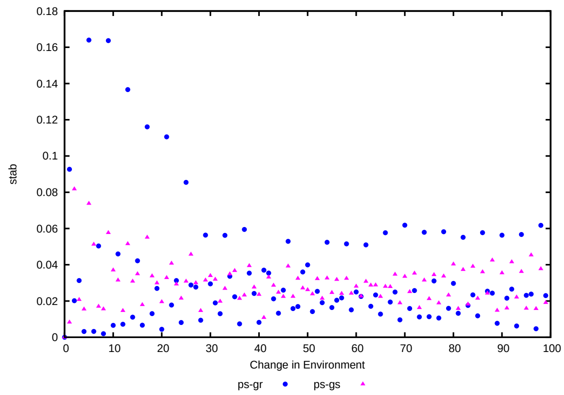
(a) $stab$ values for $n_t = 10$ and $\tau_t = 10$



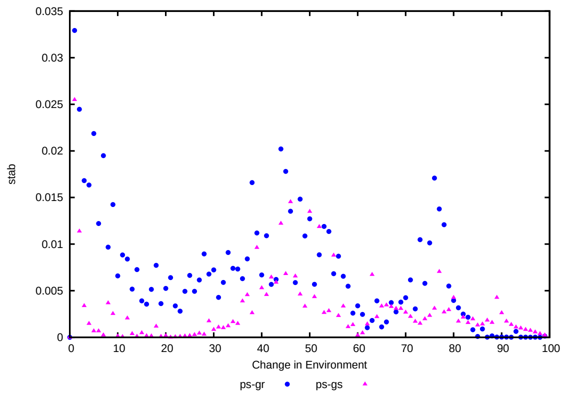
(b) $stab$ values for $n_t = 10$ and $\tau_t = 25$



(c) $stab$ values for $n_t = 10$ and $\tau_t = 50$

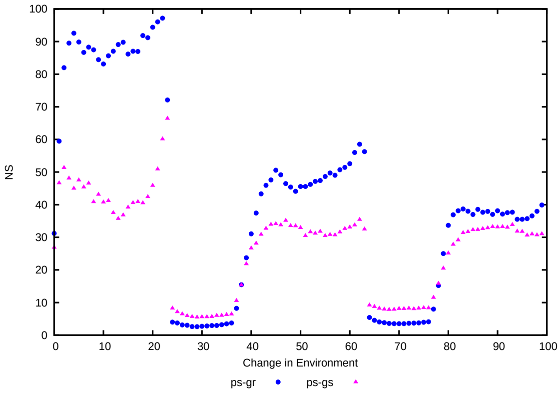


(d) $stab$ values for $nfda2_t = 1$ and $\tau_t = 10$

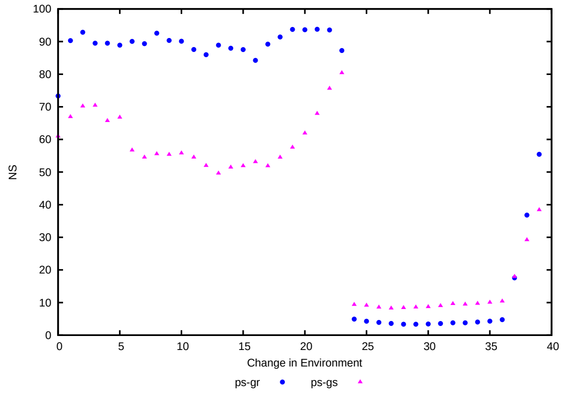


(e) $stab$ values for $n_t = 20$ and $\tau_t = 10$

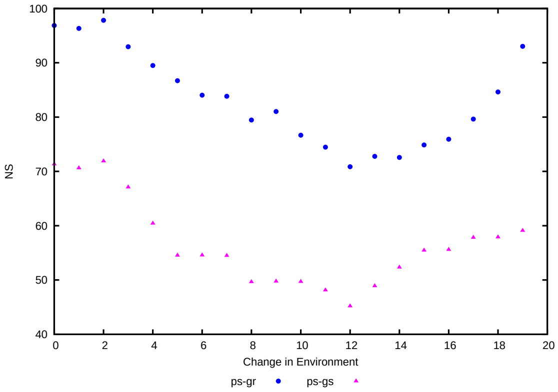
Figure 9.11: Average values of $stab$ obtained by DVEPSO using either p_s-g_s or p_s-g_r solving FDA2



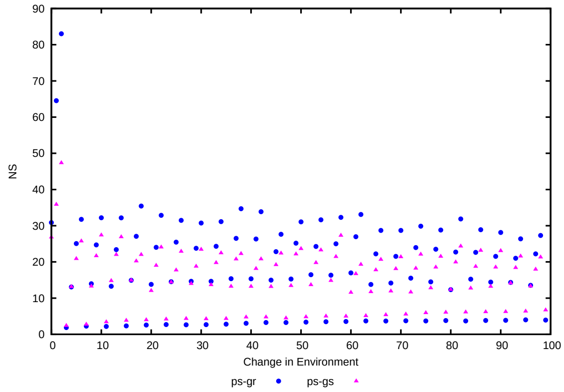
(a) NS values for $n_t = 10$ and $\tau_t = 10$



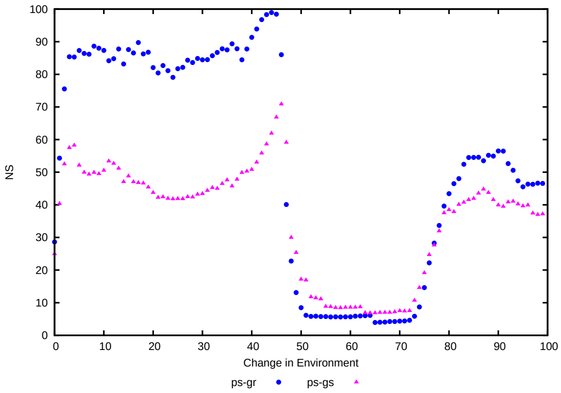
(b) NS values for $n_t = 10$ and $\tau_t = 25$



(c) NS values for $n_t = 10$ and $\tau_t = 50$



(d) NS values for $n_t = 1$ and $\tau_t = 10$



(e) NS values for $n_t = 20$ and $\tau_t = 10$

Figure 9.12: Average values of NS obtained by DVEPSO using either p_s-g_s or p_s-g_r solving FDA2

dMOP2 with a deceptive POF, dMOP2$_{dec}$. When solving dMOP2, all p_s combinations performed average, except p_s-g_s that performed the worst. Solving dMOP2$_{iso}$, p_s-g_r and p_s-g_n performed well, while the other two p_s combinations performed poorly. Once again, p_s-g_s obtained the worst performance when solving dMOP2$_{iso}$. However, for dMOP2$_{dec}$, all p_s combinations performed really well, obtaining the top four overall ranks. Tables 9.14 to 9.16 present the wins and losses for dMOP2, dMOP2$_{iso}$ and dMOP2$_{dec}$ respectively.

Table 9.14: Wins and Losses of dMOP2

$\mathbf{n_t}$	$\tau_\mathbf{t}$	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	7	1	1	1	4	0	4	3	4	3	3	3	3	3	4	3
10	10	*acc*	Losses	0	12	12	12	1	7	1	0	1	0	0	0	0	0	1	0
10	10	*acc*	Diff	7	-11	-11	-11	3	-7	3	3	3	3	3	3	3	3	3	3
10	10	*acc*	Rank	**1**	14	14	14	2	13	2	2	2	2	2	2	2	2	2	2
10	25	*acc*	Wins	2	3	2	0	1	1	0	0	1	6	3	0	2	1	1	6
10	25	*acc*	Losses	2	3	4	12	0	2	0	3	0	0	1	2	0	0	0	0
10	25	*acc*	Diff	0	0	-2	-12	1	-1	0	-3	1	6	2	-2	2	1	1	6
10	25	*acc*	Rank	9	9	13	16	5	12	9	15	5	**1**	3	13	3	5	5	**1**
10	50	*acc*	Wins	0	2	0	1	4	4	4	5	5	4	5	5	5	5	4	4
10	50	*acc*	Losses	13	12	14	12	0	6	0	0	0	0	0	0	0	0	0	0
10	50	*acc*	Diff	-13	-10	-14	-11	4	-2	4	5	5	4	5	5	5	5	4	4
10	50	*acc*	Rank	15	13	16	14	7	12	7	**1**	**1**	7	**1**	**1**	**1**	**1**	7	7
1	10	*acc*	Wins	5	5	5	4	0	2	3	0	3	0	1	2	0	0	1	3
1	10	*acc*	Losses	2	0	10	0	2	3	0	6	3	1	0	2	1	2	0	2
1	10	*acc*	Diff	3	5	-5	4	-2	-1	3	-6	0	-1	1	0	-1	-2	1	1
1	10	*acc*	Rank	3	**1**	15	2	13	10	3	16	8	10	5	8	10	13	5	5
20	10	*acc*	Wins	0	6	4	3	3	1	1	1	2	1	1	1	1	1	1	2
20	10	*acc*	Losses	15	4	1	0	1	0	1	2	1	1	0	0	1	0	1	1
20	10	*acc*	Diff	-15	2	3	3	2	1	0	-1	1	0	1	1	0	1	0	1
20	10	*acc*	Rank	16	3	**1**	**1**	3	5	11	15	5	11	5	5	11	5	11	5
all	all	*acc*	Wins	14	17	12	9	12	8	12	9	15	14	13	11	11	10	11	18
all	all	*acc*	Losses	32	31	41	36	4	18	2	11	5	2	1	4	2	2	2	3
all	all	*acc*	Diff	-18	-14	-29	-27	8	-10	10	-2	10	12	12	7	9	8	9	15
all	all	*acc*	Rank	14	13	16	15	8	12	4	11	4	2	2	10	6	8	6	**1**
10	10	*stab*	Wins	6	4	3	2	1	0	1	0	1	0	0	0	0	0	1	0
10	10	*stab*	Losses	0	1	1	1	1	5	3	0	2	0	1	0	0	0	3	1
10	10	*stab*	Diff	6	3	2	1	0	-5	-2	0	-1	0	-1	0	0	0	-2	-1
10	10	*stab*	Rank	**1**	2	3	4	5	16	14	5	11	5	11	5	5	5	14	11
10	25	*stab*	Wins	1	0	1	0	0	0	0	0	0	2	0	0	0	0	0	2
10	25	*stab*	Losses	0	0	0	4	0	0	0	2	0	0	0	0	0	0	0	0
10	25	*stab*	Diff	1	0	1	-4	0	0	0	-2	0	2	0	0	0	0	0	2
10	25	*stab*	Rank	3	5	3	16	5	5	5	15	5	**1**	5	5	5	5	5	**1**
10	50	*stab*	Wins	0	0	0	0	4	4	4	4	5	4	4	4	5	5	4	4
10	50	*stab*	Losses	12	12	12	12	0	3	0	0	0	0	0	0	0	0	0	0
10	50	*stab*	Diff	-12	-12	-12	-12	4	1	4	4	5	4	4	4	5	5	4	4
10	50	*stab*	Rank	13	13	13	13	4	12	4	4	**1**	4	4	4	**1**	**1**	4	4

Continued on next page

n_t	τ_t	PM	Results	pbest-gbest combination															
				s-s	s-n	s-d	s-r	n-s	n-n	n-d	n-r	r-s	r-n	r-d	r-r	d-s	d-n	d-d	d-r
1	10	*stab*	Wins	1	3	2	3	0	1	0	0	2	0	0	0	0	0	0	1
1	10	*stab*	Losses	0	0	3	0	0	0	0	5	1	0	0	0	0	3	0	1
1	10	*stab*	Diff	1	3	-1	3	0	1	0	-5	1	0	0	0	0	-3	0	0
1	10	*stab*	Rank	3	**1**	14	**1**	6	3	6	16	3	6	6	6	6	15	6	6
20	10	*stab*	Wins	3	1	4	11	3	0	0	0	0	0	0	0	0	0	0	0
20	10	*stab*	Losses	3	1	0	0	1	1	1	3	1	1	3	3	1	1	1	1
20	10	*stab*	Diff	0	0	4	11	2	-1	-1	-3	-1	-1	-3	-3	-1	-1	-1	-1
20	10	*stab*	Rank	4	4	2	**1**	3	6	6	14	6	6	14	14	6	6	6	6
all	all	*stab*	Wins	11	8	10	16	8	5	5	4	8	6	4	4	5	5	5	7
all	all	*stab*	Losses	15	14	16	17	2	9	4	10	4	1	4	3	1	4	4	3
all	all	*stab*	Diff	-4	-6	-6	-1	6	-4	1	-6	4	5	0	1	4	1	1	4
all	all	*stab*	Rank	12	14	14	11	**1**	12	6	14	3	2	10	6	3	6	6	3
10	25	NS	Wins	12	13	13	12	0	3	3	3	1	3	3	3	0	3	3	3
10	25	NS	Losses	2	0	0	0	13	4	4	4	13	4	4	4	14	4	4	4
10	25	NS	Diff	10	13	13	12	-13	-1	-1	-1	-12	-1	-1	-1	-14	-1	-1	-1
10	25	NS	Rank	4	**1**	**1**	3	15	5	5	5	14	5	5	5	16	5	5	5
10	50	NS	Wins	3	8	5	9	0	3	3	3	0	4	4	3	0	3	3	3
10	50	NS	Losses	5	0	0	0	13	0	0	3	13	0	0	2	13	2	2	1
10	50	NS	Diff	-2	8	5	9	-13	3	3	0	-13	4	4	1	-13	1	1	2
10	50	NS	Rank	13	2	3	**1**	14	6	6	12	14	4	4	9	14	9	9	8
1	10	NS	Wins	3	4	10	0	2	2	1	0	3	2	0	0	2	2	2	2
1	10	NS	Losses	9	8	1	0	1	1	2	3	0	1	3	3	1	1	1	0
1	10	NS	Diff	-6	-4	9	0	1	1	-1	-3	3	1	-3	-3	1	1	1	2
1	10	NS	Rank	16	15	**1**	10	4	4	11	12	2	4	12	12	4	4	4	3
all	all	NS	Wins	18	25	28	21	2	8	7	6	4	9	7	6	2	8	8	8
all	all	NS	Losses	16	8	1	0	27	5	6	10	26	5	7	9	28	7	7	5
all	all	NS	Diff	2	17	27	21	-25	3	1	-4	-22	4	0	-3	-26	1	1	3
all	all	NS	Rank	7	3	**1**	2	15	5	8	13	14	4	11	12	16	8	8	5
10	10	all	Wins	13	5	4	3	5	0	5	3	5	3	3	3	3	3	5	3
10	10	all	Losses	0	13	13	13	2	12	4	0	3	0	1	0	0	0	4	1
10	10	all	Diff	13	-8	-9	-10	3	-12	1	3	2	3	2	3	3	3	1	2
10	10	all	Rank	**1**	13	14	15	2	16	11	2	8	2	8	2	2	2	11	8
10	25	all	Wins	15	16	16	12	1	4	3	3	2	11	6	3	2	4	4	11
10	25	all	Losses	4	3	4	16	13	6	4	9	13	4	5	6	14	4	4	4
10	25	all	Diff	11	13	12	-4	-12	-2	-1	-6	-11	7	1	-3	-12	0	0	7
10	25	all	Rank	3	**1**	2	12	15	10	9	13	14	4	6	11	15	7	7	4
10	50	all	Wins	3	10	5	10	8	11	11	12	10	12	13	12	10	13	11	11
10	50	all	Losses	30	24	26	24	13	9	0	3	13	0	0	2	13	2	2	1
10	50	all	Diff	-27	-14	-21	-14	-5	2	11	9	-3	12	13	10	-3	11	9	10
10	50	all	Rank	16	13	15	13	12	9	3	7	10	2	**1**	5	10	3	7	5
1	10	all	Wins	9	12	17	7	2	5	4	0	8	2	1	2	2	2	3	6
1	10	all	Losses	11	8	14	0	3	4	2	14	4	2	3	5	2	6	1	3
1	10	all	Diff	-2	4	3	7	-1	1	2	-14	4	0	-2	-3	0	-4	2	3
1	10	all	Rank	12	2	4	**1**	11	8	6	16	2	9	12	14	9	15	6	4
20	10	all	Wins	3	7	8	14	6	1	1	1	2	1	1	1	1	1	1	2
20	10	all	Losses	18	5	1	0	2	1	2	5	2	2	3	3	2	1	2	2
20	10	all	Diff	-15	2	7	14	4	0	-1	-4	0	-1	-2	-2	-1	0	-1	0
20	10	all	Rank	16	4	2	**1**	3	5	9	15	5	9	13	13	9	5	9	5
all	all	all	Wins	43	50	50	46	22	21	24	19	27	29	24	21	18	23	24	33
all	all	all	Losses	63	53	58	53	33	32	12	31	35	8	12	16	31	13	13	11

Continued on next page

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
all	all	all	Diff	-20	-3	-8	-7	-11	-11	12	-12	-8	21	12	5	-13	10	11	22
all	all	all	Rank	16	8	10	9	12	12	3	14	10	2	3	7	15	6	5	**1**

Table 9.15: Wins and Losses of dMOP2$_{iso}$

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	4	9	11	3	0	0	0	1	1	1	1	0	1	1	2	1
10	10	*acc*	Losses	11	1	1	0	3	3	2	1	2	2	1	3	2	2	2	0
10	10	*acc*	Diff	-7	8	10	3	-3	-3	-2	0	-1	-1	0	-3	-1	-1	0	1
10	10	*acc*	Rank	16	2	**1**	3	13	13	12	5	8	8	5	13	8	8	5	4
10	25	*acc*	Wins	0	0	3	1	3	3	2	3	3	3	9	3	4	3	2	5
10	25	*acc*	Losses	13	13	3	11	0	1	2	0	1	0	0	1	0	0	2	0
10	25	*acc*	Diff	-13	-13	0	-10	3	2	0	3	2	3	9	2	4	3	0	5
10	25	*acc*	Rank	15	15	11	14	4	8	11	4	8	4	**1**	8	3	4	11	2
10	50	*acc*	Wins	0	2	3	0	5	4	4	5	5	4	4	4	4	4	6	4
10	50	*acc*	Losses	14	13	12	14	0	0	0	0	0	0	0	0	4	0	0	1
10	50	*acc*	Diff	-14	-11	-9	-14	5	4	4	5	5	4	4	4	0	4	6	3
10	50	*acc*	Rank	15	14	13	15	2	5	5	2	2	5	5	5	12	5	**1**	11
1	10	*acc*	Wins	0	1	4	10	3	3	4	5	3	4	6	4	2	2	3	3
1	10	*acc*	Losses	15	14	11	5	1	1	0	0	1	0	0	0	2	3	1	3
1	10	*acc*	Diff	-15	-13	-7	5	2	2	4	5	2	4	6	4	0	-1	2	0
1	10	*acc*	Rank	16	15	14	2	7	7	4	2	7	4	**1**	4	11	13	7	11
20	10	*acc*	Wins	0	8	0	8	2	6	4	2	4	6	2	2	6	2	2	2
20	10	*acc*	Losses	14	4	14	4	2	0	0	2	2	0	2	2	0	4	2	4
20	10	*acc*	Diff	-14	4	-14	4	0	6	4	0	2	6	0	0	6	-2	0	-2
20	10	*acc*	Rank	15	4	15	4	8	**1**	4	8	7	**1**	8	8	**1**	13	8	13
all	all	*acc*	Wins	4	20	21	22	13	16	14	16	16	18	22	13	17	12	15	15
all	all	*acc*	Losses	67	45	41	34	6	5	4	3	6	2	3	6	8	9	7	8
all	all	*acc*	Diff	-63	-25	-20	-12	7	11	10	13	10	16	19	7	9	3	8	7
all	all	*acc*	Rank	16	15	14	13	9	4	5	3	5	2	**1**	9	7	12	8	9
10	10	*stab*	Wins	0	10	4	12	0	0	0	0	0	0	0	0	0	0	0	0
10	10	*stab*	Losses	2	0	0	0	2	0	3	3	2	2	3	2	1	2	1	3
10	10	*stab*	Diff	-2	10	4	12	-2	0	-3	-3	-2	-2	-3	-2	-1	-2	-1	-3
10	10	*stab*	Rank	7	2	3	**1**	7	4	13	13	7	7	13	7	5	7	5	13
10	25	*stab*	Wins	0	1	3	0	0	0	0	0	0	0	9	0	3	1	0	4
10	25	*stab*	Losses	4	5	0	4	0	1	1	1	1	1	0	1	0	0	2	0
10	25	*stab*	Diff	-4	-4	3	-4	0	-1	-1	-1	-1	-1	9	-1	3	1	-2	4
10	25	*stab*	Rank	14	14	3	14	6	7	7	7	7	7	**1**	7	3	5	13	2
10	50	*stab*	Wins	0	0	1	0	4	4	4	5	5	4	4	2	2	4	5	4
10	50	*stab*	Losses	12	10	10	13	0	0	0	0	0	0	0	0	3	0	0	0
10	50	*stab*	Diff	-12	-10	-9	-13	4	4	4	5	5	4	4	2	-1	4	5	4
10	50	*stab*	Rank	15	14	13	16	4	4	4	**1**	**1**	4	4	11	12	4	**1**	4
1	10	*stab*	Wins	6	8	3	3	0	0	0	3	0	0	3	0	0	0	0	0
1	10	*stab*	Losses	3	0	0	0	0	2	3	3	2	2	3	2	0	2	2	2
1	10	*stab*	Diff	3	8	3	3	0	-2	-3	0	-2	-2	0	-2	0	-2	-2	-2
1	10	*stab*	Rank	2	**1**	2	2	5	9	16	5	9	9	5	9	5	9	9	9
20	10	*stab*	Wins	0	4	0	5	0	2	0	0	1	1	0	0	2	0	0	0
20	10	*stab*	Losses	2	0	2	0	1	0	0	0	0	0	0	0	0	6	0	4
20	10	*stab*	Diff	-2	4	-2	5	-1	2	0	0	1	1	0	0	2	-6	0	-4

Continued on next page

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
20	10	*stab*	Rank	13	2	13	**1**	12	3	7	7	5	5	7	7	3	16	7	15
all	all	*stab*	Wins	6	23	11	20	4	6	4	8	6	5	16	2	7	5	5	8
all	all	*stab*	Losses	23	15	12	17	3	3	7	7	5	5	6	5	4	10	5	9
all	all	*stab*	Diff	-17	8	-1	3	1	3	-3	1	1	0	10	-3	3	-5	0	-1
all	all	*stab*	Rank	16	2	11	3	6	3	13	6	6	9	**1**	13	3	15	9	11
10	25	NS	Wins	12	13	13	12	0	3	3	3	1	3	3	3	0	3	3	3
10	25	NS	Losses	2	0	0	0	13	4	4	4	13	4	4	4	14	4	4	4
10	25	NS	Diff	10	13	13	12	-13	-1	-1	-1	-12	-1	-1	-1	-14	-1	-1	-1
10	25	NS	Rank	4	**1**	**1**	3	15	5	5	5	14	5	5	5	16	5	5	5
10	50	NS	Wins	3	8	5	9	0	3	3	3	0	4	4	3	0	3	3	3
10	50	NS	Losses	5	0	0	0	13	0	0	3	13	0	0	2	13	2	2	1
10	50	NS	Diff	-2	8	5	9	-13	3	3	0	-13	4	4	1	-13	1	1	2
10	50	NS	Rank	13	2	3	**1**	14	6	6	12	14	4	4	9	14	9	9	8
all	all	NS	Wins	15	21	18	21	0	6	6	6	1	7	7	6	0	6	6	6
all	all	NS	Losses	7	0	0	0	26	4	4	7	26	4	4	6	27	6	6	5
all	all	NS	Diff	8	21	18	21	-26	2	2	-1	-25	3	3	0	-27	0	0	1
all	all	NS	Rank	4	**1**	3	**1**	15	7	7	13	14	5	5	10	16	10	10	9
10	10	all	Wins	4	19	15	15	0	0	0	1	1	1	1	0	1	1	2	1
10	10	all	Losses	13	1	1	0	5	3	5	4	4	4	4	5	3	4	3	3
10	10	all	Diff	-9	18	14	15	-5	-3	-5	-3	-3	-3	-3	-5	-2	-3	-1	-2
10	10	all	Rank	16	**1**	3	2	13	7	13	7	7	7	7	13	5	7	4	5
10	25	all	Wins	12	14	19	13	3	6	5	6	4	6	21	6	7	7	5	12
10	25	all	Losses	19	18	3	15	13	6	7	5	15	5	4	6	14	4	8	4
10	25	all	Diff	-7	-4	16	-2	-10	0	-2	1	-11	1	17	0	-7	3	-3	8
10	25	all	Rank	13	12	2	9	15	7	9	5	16	5	**1**	7	13	4	11	3
10	50	all	Wins	3	10	9	9	9	11	11	13	10	12	12	9	6	11	14	11
10	50	all	Losses	31	23	22	27	13	0	0	3	13	0	0	2	20	2	2	2
10	50	all	Diff	-28	-13	-13	-18	-4	11	11	10	-3	12	12	7	-14	9	12	9
10	50	all	Rank	16	12	12	15	11	4	4	6	10	**1**	**1**	9	14	7	**1**	7
1	10	all	Wins	6	9	7	13	3	3	4	8	3	4	9	4	2	2	3	3
1	10	all	Losses	18	14	11	5	1	3	3	3	3	2	3	2	2	5	3	5
1	10	all	Diff	-12	-5	-4	8	2	0	1	5	0	2	6	2	0	-3	0	-2
1	10	all	Rank	16	15	14	**1**	4	8	7	3	8	4	2	4	8	13	8	12
20	10	all	Wins	0	12	0	13	2	8	4	2	5	7	2	2	8	2	2	2
20	10	all	Losses	16	4	16	4	3	0	0	2	2	0	2	2	0	10	2	8
20	10	all	Diff	-16	8	-16	9	-1	8	4	0	3	7	0	0	8	-8	0	-6
20	10	all	Rank	15	2	15	**1**	12	2	6	8	7	5	8	8	2	14	8	13
all	all	all	Wins	25	64	50	63	17	28	24	30	23	30	45	21	24	23	26	29
all	all	all	Losses	97	60	53	51	35	12	15	17	37	11	13	17	39	25	18	22
all	all	all	Diff	-72	4	-3	12	-18	16	9	13	-14	19	32	4	-15	-2	8	7
all	all	all	Rank	16	9	12	5	15	3	6	4	13	2	**1**	9	14	11	7	8

Table 9.16: Wins and Losses of dMOP2$_{dec}$

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	10	*acc*	Wins	12	12	12	12	0	0	0	4	0	0	0	0	6	0	2	0
10	10	*acc*	Losses	0	0	0	0	7	6	4	4	4	7	5	6	4	4	4	5
10	10	*acc*	Diff	12	12	12	12	-7	-6	-4	0	-4	-7	-5	-6	2	-4	-2	-5
10	10	*acc*	Rank	**1**	**1**	**1**	**1**	15	13	8	6	8	15	11	13	5	8	7	11
10	25	*acc*	Wins	12	12	12	9	0	1	1	0	4	0	0	4	0	0	0	0
10	25	*acc*	Losses	0	0	0	0	4	4	3	4	3	6	8	3	4	6	4	6
10	25	*acc*	Diff	12	12	12	9	-4	-3	-2	-4	1	-6	-8	1	-4	-6	-4	-6
10	25	*acc*	Rank	**1**	**1**	**1**	4	9	8	7	9	5	13	16	5	9	13	9	13
10	50	*acc*	Wins	5	10	7	3	0	0	0	0	0	0	0	0	0	0	0	0
10	50	*acc*	Losses	0	0	0	0	4	1	3	2	0	0	2	4	3	4	1	1
10	50	*acc*	Diff	5	10	7	3	-4	-1	-3	-2	0	0	-2	-4	-3	-4	-1	-1
10	50	*acc*	Rank	3	**1**	2	4	14	7	12	10	5	5	10	14	12	14	7	7
1	10	*acc*	Wins	9	12	12	12	0	1	0	0	0	0	0	1	0	1	1	0
1	10	*acc*	Losses	0	0	0	0	4	3	4	8	4	4	4	4	4	3	3	4
1	10	*acc*	Diff	9	12	12	12	-4	-2	-4	-8	-4	-4	-4	-3	-4	-2	-2	-4
1	10	*acc*	Rank	4	**1**	**1**	**1**	9	5	9	16	9	9	9	8	9	5	5	9
20	10	*acc*	Wins	0	6	4	3	3	1	1	1	2	1	1	1	1	1	1	2
20	10	*acc*	Losses	15	4	1	0	1	0	1	2	1	1	0	0	1	0	1	1
20	10	*acc*	Diff	-15	2	3	3	2	1	0	-1	1	0	1	1	0	1	0	1
20	10	*acc*	Rank	16	3	**1**	**1**	3	5	11	15	5	11	5	5	11	5	11	5
all	all	*acc*	Wins	38	52	47	39	3	3	2	5	6	1	1	6	7	2	4	2
all	all	*acc*	Losses	15	4	1	0	20	14	15	20	12	18	19	17	16	17	13	17
all	all	*acc*	Diff	23	48	46	39	-17	-11	-13	-15	-6	-17	-18	-11	-9	-15	-9	-15
all	all	*acc*	Rank	4	**1**	2	3	14	8	10	11	5	14	16	8	6	11	6	11
10	10	*stab*	Wins	12	12	12	12	0	0	0	4	0	0	0	0	6	0	3	0
10	10	*stab*	Losses	0	0	0	0	7	6	4	4	4	7	5	7	4	4	4	5
10	10	*stab*	Diff	12	12	12	12	-7	-6	-4	0	-4	-7	-5	-7	2	-4	-1	-5
10	10	*stab*	Rank	**1**	**1**	**1**	**1**	14	13	8	6	8	14	11	14	5	8	7	11
10	25	*stab*	Wins	12	12	12	7	0	0	0	0	2	0	0	1	2	0	0	0
10	25	*stab*	Losses	0	0	0	0	4	3	3	4	3	4	6	3	3	4	4	7
10	25	*stab*	Diff	12	12	12	7	-4	-3	-3	-4	-1	-4	-6	-2	-1	-4	-4	-7
10	25	*stab*	Rank	**1**	**1**	**1**	4	10	8	8	10	5	10	15	7	5	10	10	16
10	50	*stab*	Wins	3	8	6	2	0	0	0	0	0	0	0	0	0	0	0	0
10	50	*stab*	Losses	0	0	0	0	4	0	3	1	0	0	2	4	2	2	0	1
10	50	*stab*	Diff	3	8	6	2	-4	0	-3	-1	0	0	-2	-4	-2	-2	0	-1
10	50	*stab*	Rank	3	**1**	2	4	15	5	14	9	5	5	11	15	11	11	5	9
1	10	*stab*	Wins	9	12	12	12	0	1	1	0	0	0	0	1	0	1	1	0
1	10	*stab*	Losses	0	0	0	0	4	3	4	9	4	4	4	4	4	3	3	4
1	10	*stab*	Diff	9	12	12	12	-4	-2	-3	-9	-4	-4	-4	-3	-4	-2	-2	-4
1	10	*stab*	Rank	4	**1**	**1**	**1**	10	5	8	16	10	10	10	8	10	5	5	10
20	10	*stab*	Wins	3	1	4	11	3	0	0	0	0	0	0	0	0	0	0	0
20	10	*stab*	Losses	3	1	0	0	1	1	1	3	1	1	3	3	1	1	1	1
20	10	*stab*	Diff	0	0	4	11	2	-1	-1	-3	-1	-1	-3	-3	-1	-1	-1	-1
20	10	*stab*	Rank	4	4	2	**1**	3	6	6	14	6	6	14	14	6	6	6	6
all	all	*stab*	Wins	39	45	46	44	3	1	1	4	2	0	0	2	8	1	4	0
all	all	*stab*	Losses	3	1	0	0	20	13	15	21	12	16	20	21	14	14	12	18
all	all	*stab*	Diff	36	44	46	44	-17	-12	-14	-17	-10	-16	-20	-19	-6	-13	-8	-18
all	all	*stab*	Rank	4	2	**1**	2	12	8	10	12	7	11	16	15	5	9	6	14

Continued on next page

$\mathbf{n_t}$	τ_t	**PM**	**Results**	**pbest-gbest combination**															
				s-s	**s-n**	**s-d**	**s-r**	**n-s**	**n-n**	**n-d**	**n-r**	**r-s**	**r-n**	**r-d**	**r-r**	**d-s**	**d-n**	**d-d**	**d-r**
10	25	NS	Wins	12	13	13	12	0	3	3	3	1	3	3	3	0	3	3	3
10	25	NS	Losses	2	0	0	0	13	4	4	4	13	4	4	4	14	4	4	4
10	25	NS	Diff	10	13	13	12	-13	-1	-1	-1	-12	-1	-1	-1	-14	-1	-1	-1
10	25	NS	Rank	4	**1**	**1**	3	15	5	5	5	14	5	5	5	16	5	5	5
10	50	NS	Wins	3	8	5	9	0	3	3	3	0	4	4	3	0	3	3	3
10	50	NS	Losses	5	0	0	0	13	0	0	3	13	0	0	2	13	2	2	1
10	50	NS	Diff	-2	8	5	9	-13	3	3	0	-13	4	4	1	-13	1	1	2
10	50	NS	Rank	13	2	3	**1**	14	6	6	12	14	4	4	9	14	9	9	8
1	10	NS	Wins	2	12	0	2	2	0	2	2	1	1	2	2	2	2	0	2
1	10	NS	Losses	10	0	0	8	1	3	1	1	1	1	1	1	1	1	3	1
1	10	NS	Diff	-8	12	0	-6	1	-3	1	1	0	0	1	1	1	1	-3	1
1	10	NS	Rank	16	**1**	10	15	2	13	2	2	10	10	2	2	2	2	13	2
all	all	NS	Wins	17	33	18	23	2	6	8	8	2	8	9	8	2	8	6	8
all	all	NS	Losses	17	0	0	8	27	7	5	8	27	5	5	7	28	7	9	6
all	all	NS	Diff	0	33	18	15	-25	-1	3	0	-25	3	4	1	-26	1	-3	2
all	all	NS	Rank	10	**1**	2	3	14	12	5	10	14	5	4	8	16	8	13	7
10	10	all	Wins	24	24	24	24	0	0	0	8	0	0	0	0	12	0	5	0
10	10	all	Losses	0	0	0	0	14	12	8	8	8	14	10	13	8	8	8	10
10	10	all	Diff	24	24	24	24	-14	-12	-8	0	-8	-14	-10	-13	4	-8	-3	-10
10	10	all	Rank	**1**	**1**	**1**	**1**	15	13	8	6	8	15	11	14	5	8	7	11
10	25	all	Wins	36	37	37	28	0	4	4	3	7	3	3	8	2	3	3	3
10	25	all	Losses	2	0	0	0	21	11	10	12	19	14	18	10	21	14	12	17
10	25	all	Diff	34	37	37	28	-21	-7	-6	-9	-12	-11	-15	-2	-19	-11	-9	-14
10	25	all	Rank	3	**1**	**1**	4	16	7	6	8	12	10	14	5	15	10	8	13
10	50	all	Wins	11	26	18	14	0	3	3	3	0	4	4	3	0	3	3	3
10	50	all	Losses	5	0	0	0	21	1	6	6	13	0	4	10	18	8	3	3
10	50	all	Diff	6	26	18	14	-21	2	-3	-3	-13	4	0	-7	-18	-5	0	0
10	50	all	Rank	4	**1**	2	3	16	6	10	10	14	5	7	13	15	12	7	7
1	10	all	Wins	20	36	24	26	2	2	3	2	1	1	2	4	2	4	2	2
1	10	all	Losses	10	0	0	8	9	9	9	18	9	9	9	9	9	7	9	9
1	10	all	Diff	10	36	24	18	-7	-7	-6	-16	-8	-8	-7	-5	-7	-3	-7	-7
1	10	all	Rank	4	**1**	2	3	8	8	7	16	14	14	8	6	8	5	8	8
20	10	all	Wins	3	7	8	14	6	1	1	1	2	1	1	1	1	1	1	2
20	10	all	Losses	18	5	1	0	2	1	2	5	2	2	3	3	2	1	2	2
20	10	all	Diff	-15	2	7	14	4	0	-1	-4	0	-1	-2	-2	-1	0	-1	0
20	10	all	Rank	16	4	2	**1**	3	5	9	15	5	9	13	13	9	5	9	5
all	all	all	Wins	94	130	111	106	8	10	11	17	10	9	10	16	17	11	14	10
all	all	all	Losses	35	5	1	8	67	34	35	49	51	39	44	45	58	38	34	41
all	all	all	Diff	59	125	110	98	-59	-24	-24	-32	-41	-30	-34	-29	-41	-27	-20	-31
all	all	all	Rank	4	**1**	2	3	16	6	6	12	14	10	13	9	14	8	5	11

9.5 Summary

This chapter discussed the DVEPSO algorithm, which is an adaptation of SMOO VEPSO for DMOO. Similar to VEPSO, DVEPSO has two layers, namely a top layer that manages the top-level tasks and a lower layer that consists of the sub-swarms that handle the lower level tasks. On the lower level, the sub-swarms of DVEPSO checks the environment to determine whether a change has occurred, in addition to optimising the objectives that are performed by the sub-swarms of VEPSO. On the top level, in addition to the tasks of knowledge sharing and archive management performed by VEPSO, DVEPSO also responds to changes in the environment that were detected by the sub-swarms.

The optimisation process of DVEPSO is guided by local and global guides. Various ways of updating the local and global guides exist. This chapter investigated the influence of the various guide update approaches on the performance of DVEPSO. The results indicated that guide update approaches that incorporate Pareto-dominance knowledge outperformed the guide update approach of the original VEPSO algorithm that does not incorporate Pareto-dominance. The guide update approach that achieved the overall best performance was p_s-g_r. With this approach, the local guide is updated in such a way that the particle's fitness is measured with regards to only the objective function that the specific swarm optimises. Only if an improvement in the fitness of the current local guide can be obtained, the guide is updated, and no Pareto-dominance information is used. The global guide is updated if the new pbest is non-dominated with respect to the global guide, by randomly selecting either the pbest or the corresponding global guide.

The next chapter investigates the influence of other parameters on the performance of DVEPSO. These parameters include knowledge sharing swarm topologies, approaches to manage boundary constraint violations and approaches to respond to environment changes.