Supplementary Materials

Haotian Zhang, Jialong Shi, Jianyong Sun, *Senior Member, IEEE* and Zongben Xu, *Member, IEEE*

I. Details

In the Supplementary Materials, we first describe how to embed GNN into PPLS/D. Second, we talk about the techniques for using PPLS/D$_{\text{GNN}}$ for 3 and more objective problems. Then the algorithm PPLS/D is summarized in Alg. 1. Finally, the ablation study of PLS$_{\text{GNN}}$ is presented.

A. Embedding networks into PPLS/D for 2-obj

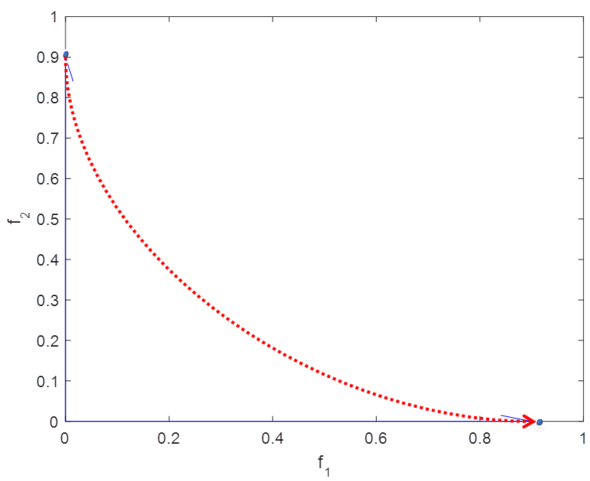

Fig. 1. The preferring vector $[\psi(v_i,\theta), 1-\psi(v_i,\theta)]$ w.r.t. $\frac{FEs}{budget}$ in PLS.

For PLS, the range of the output of network $\psi(\cdot,\theta)$ is (0,1). Then $h_i = \psi(v_i,\theta)f_1^i + (1-\psi(v_i,\theta))f_2^i$ is to calculate feature for each solution. Thus the vector $[\psi(v_i,\theta), 1-\psi(v_i,\theta)]$ can cover the first quadrant. Just as shown in Section IV-D.5, the vector scans the first quadrant (Fig. 1) when $\frac{FEs}{budget}$ varies from 0 to 1. However, PPLS/D decomposes the problem by setting several preference weight, which means for each sub-problem, we do not need to scan the whole the first quadrant. Notice that we use **preferring vector** to represent $[\psi(v_i,\theta), 1-\psi(v_i,\theta)]$ which is different from the **preference weight** $\{\lambda_l\}_{l=1}^{L}$ in PPLS/D.

For instance, if the preference weights for PPLS/D are:

$$\begin{pmatrix} \mathbf{1}: & 0 & 1 \\ \mathbf{2}: & 0.2 & 0.8 \\ \mathbf{3}: & 0.4 & 0.6 \\ \mathbf{4}: & 0.6 & 0.4 \\ \mathbf{5}: & 0.8 & 0.2 \\ \mathbf{6}: & 1 & 0 \end{pmatrix}$$

In PPLS/D, for one sub-problem, one preference weight is used. Then the solutions are classified based on the distance to these weights. A new solution which does not belong to current class will be discarded. It implies that, for a sub-problem in PPLS/D, only part of space is searched.

Actually, we can know the boundary of search space of each sub-problem, that is: the angular bisector of the angle constructing by the adjacent preference weights:

$$\begin{pmatrix} \mathbf{1}: & 0.1096 & 0.8904 \\ \mathbf{2}: & 0.3067 & 0.6933 \\ \mathbf{3}: & 0.5 & 0.5 \\ \mathbf{4}: & 0.6933 & 0.3067 \\ \mathbf{5}: & 0.8904 & 01096 \end{pmatrix}$$

which is shown in Fig. 2. Then we show the solutions obtained by PPLS/D in Fig. 3. Thus we can find that for the first sub-problem, the boundary is: $[0,1]$ and $[0.1096, 0.8904]$. For the second sub-problem, the boundary is that: $[0.1096, 0.8904]$ and $[0.3067, 0.6933]$. For the last sub-problem, the boundary is: $[0.8904, 0.1096]$ and $[1,0]$.

Notice that h_i can be re-written as $h_i = [\psi(v_i,\theta), 1-\psi(v_i,\theta)]^{\intercal} \cdot [f_1^i, f_2^i]$. Thus for each sub-problem, the preferring vector $[\psi(v_i,\theta), 1-\psi(v_i,\theta)]$ cannot be out of the boundary. Thus we just need to constrain $\psi(v_i,\theta)$ to make it in the boundary: Mathematically, we denote two boundaries for one sub-problem as low_b and up_b. up_{bx} and low_{bx} denote the first dimension of up_b and low_b respectively. Then we constrain:

$$\begin{aligned} h_i &= [\tilde{\psi}(v_i,\theta), 1-\tilde{\psi}(v_i,\theta)]^{\intercal} \cdot [f_1^i, f_2^i] \\ \tilde{\psi}(v_i,\theta) &= \psi(v_i,\theta) \times (up_{bx} - low_{bx}) + low_{bx} \end{aligned}$$

For instance, for the second sub-problem, we set $h_i = [\tilde{\psi}(v_i,\theta), 1-\tilde{\psi}(v_i,\theta)]^{\intercal} \cdot [f_1^i, f_2^i]$ where $\tilde{\psi}(v_i,\theta) = \psi(v_i,\theta) \times (0.3067 - 0.1096) + 0.1096$.

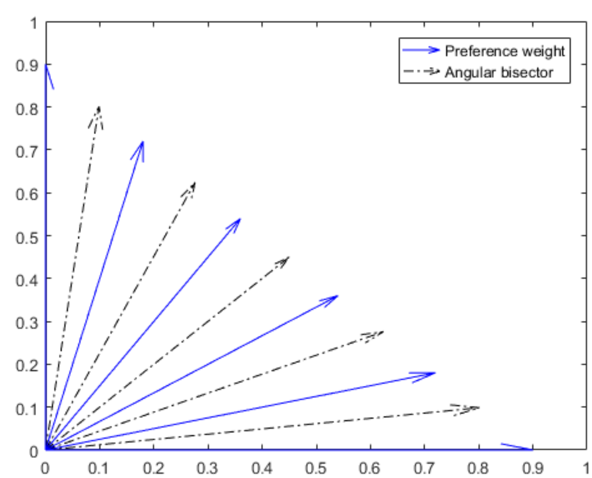

Fig. 2. Preference weight and angular bisector for PPLS/D.

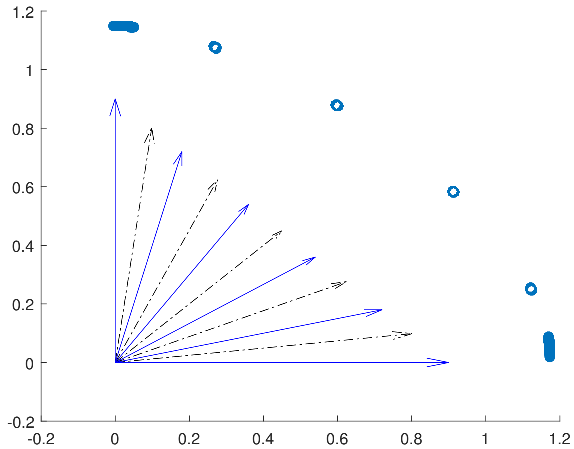

Fig. 3. Preference weight, angular bisector and solutions for PPLS/D.

Notice that the preferring vector varies from $[0,1]$ to $[1,0]$ (shown in Fig. 1) in PLS when $\frac{FEs}{budget}$ varies from 0 to 1. Thus for each subproblem, the preferring vector varies from $[low_{bx}, 1-low_{bx}]$ to $[up_{bx}, 1-up_{bx}]$ after embedding into PPLS/D, which is shown in Fig. 4.

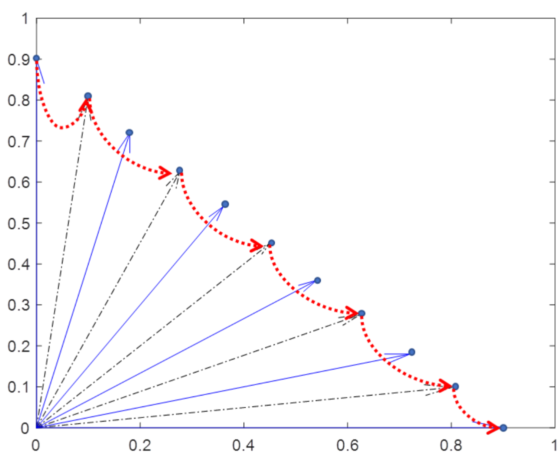

Fig. 4. The preferring vector after embedding.

B. Embedding networks into PPLS/D for 3 and more-obj

The main problems here are that: $h = \psi(v,\theta)f_1 + (1-\psi(v,\theta))f_2$ and the input of ψ contains $arctan(\frac{f_2 - z_2^*}{f_1 - z_1^*})$. Thus for 3-objective, how to define f_1 and f_2. Notice that the preference weight of an m-objective problem ($\{w_i = [w_{1,i}, \cdots, w_{m,i}]\}_{i=1}^{M}$) are in a simplex $S = \{w_i \in \mathbb{R}^m | \sum_{j=1}^{m} w_{j,i} = 1\} \subseteq \mathbb{R}^{m-1}$. The core idea is degenerating the problem into 2-objective.

Here we denote the input of ψ as $arctan(\frac{I_2 - \tilde{z}_2^*}{I_1 - \tilde{z}_1^*})$ and $h = \psi(v,\theta)I_1 + (1-\psi(v,\theta))I_2$. For 2-objective problems, $I_1 = f_1, I_2 = f_2, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_2^*$. For 3-objective, we can discuss in different cases:

- The preference weight has one zero (such as $w = [0.5, 0, 0.5]$). We can discard the dimension which is zero, and the weight is regarded as [0.5, 0.5] and $I_1 = f_1, I_2 = f_3, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_3^*$.
- The preference weight has more than one zero (such as $w = [0,0,1]$). We can discard one dimension which is zero randomly, and the weight is regarded as [0, 1] and $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$.
- The preference weight has no zero (such as $w = [\frac{1}{4}, \frac{1}{2}, \frac{1}{4}]$). We fix f_3 and aggregate f_2 and f_1: $I_1 = \frac{w_1 f_1 + w_2 f_2}{w_1 + w_2}, I_2 = f_3, \tilde{z}_1^* = \frac{w_1 z_1^* + w_2 z_2^*}{w_1 + w_2}, \tilde{z}_2^* = z_3^*$.

After degenerating into 2-objective problem, we can embed network as the last subsection.

For instance, the preference weights for 3-objective is:

$$\begin{pmatrix} \mathbf{1}: & 0 & 0 & 1 \\ \mathbf{2}: & 0 & \frac{1}{3} & \frac{2}{3} \\ \mathbf{3}: & 0 & \frac{2}{3} & \frac{1}{3} \\ \mathbf{4}: & 0 & 1 & 0 \\ \mathbf{5}: & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \mathbf{6}: & \frac{1}{3} & 0 & \frac{2}{3} \\ \mathbf{7}: & \frac{2}{3} & 0 & \frac{1}{3} \\ \mathbf{8}: & 1 & 0 & 0 \\ \mathbf{9}: & \frac{2}{3} & \frac{1}{3} & 0 \\ \mathbf{10}: & \frac{1}{3} & \frac{2}{3} & 0 \end{pmatrix}$$

which is shown in Fig. 5. The right figure of Fig. 5 represents the preference weights are in a simplex. When there is a zero in the weight (such as the first 4 rows), the weight can be degenerated into

$$\begin{pmatrix} 0 & 1 \\ \frac{1}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{1}{3} \\ 1 & 0 \end{pmatrix}$$

and $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$. The boundary can be obtained by calculating the angular bisector:

$$\begin{pmatrix} 0.191 & 0.809 \\ 0.5 & 0.5 \\ 0.809 & 0.191 \end{pmatrix}$$

Then the embedding is the same as the last section. In some words, we have:

- 1-st row: $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0$ $up_{bx} = 0.191$;
- 2-nd row: $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.191$ $up_{bx} = 0.5$;
- 3-rd row: $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.5$ $up_{bx} = 0.809$;
- 4-th row: $I_1 = f_2, I_2 = f_3, \tilde{z}_1^* = z_2^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.809$ $up_{bx} = 1$;
- 5-th row: $I_1 = \frac{\frac{1}{3}f_1 + \frac{1}{3}f_2}{\frac{1}{3}+\frac{1}{3}}, I_2 = f_3, \tilde{z}_1^* = \frac{\frac{1}{3}z_1^* + \frac{1}{3}z_2^*}{\frac{1}{3}+\frac{1}{3}}, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.5$ $up_{bx} = 0.809$;
- 6-th row: $I_1 = f_1, I_2 = f_3, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.191$ $up_{bx} = 0.5$;
- 7-th row: $I_1 = f_1, I_2 = f_3, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_3^*$ and $low_{bx} = 0.5$ $up_{bx} = 0.809$;
- 8-th row: $I_1 = f_1, I_2 = f_2, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_2^*$ and $low_{bx} = 0.809$ $up_{bx} = 1$;

- 9-th row: $I_1 = f_1, I_2 = f_2, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_2^*$ and $low_{bx} = 0.5$ $up_{bx} = 0.809$;
- 10-th row: $I_1 = f_1, I_2 = f_2, \tilde{z}_1^* = z_1^*, \tilde{z}_2^* = z_2^*$ and $low_{bx} = 0.191$ $up_{bx} = 0.5$;

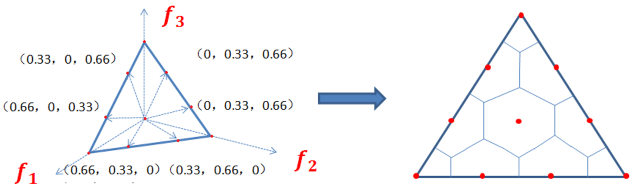

Fig. 5. Space division for 3-objective.

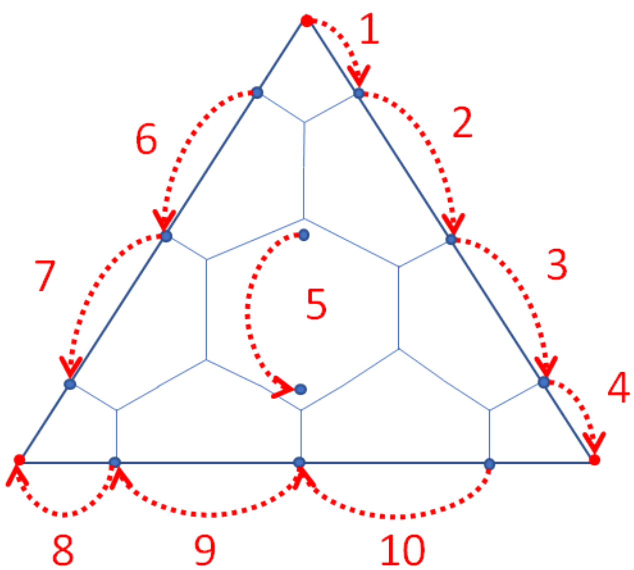

Fig. 6. Preferring vectors after embedding in PPLS/D for 3-objective problems.

Then the preferring vectors are shown in Fig. 6. Notice that for 5-th row, we do not cover all the sub space, thus in practice, we use $low_{bx} = 0.191$ $up_{bx} = 0.809$ for better coverage.

Notice that for more-objective problems, such as m-objective we can also discuss in two cases: First if the preference weight has zeros, then we discard the zero and degenerate into $(m-1)$-objective problems. Second if no zeros in the preference weight, we set: $I_1 = \frac{\sum_{i=1}^{m-1} w_i f_i}{\sum_{i=1}^{m-1} w_i}, I_2 = f_m, \tilde{z}_1^* = \frac{\sum_{i=1}^{m-1} w_i z_i^*}{\sum_{i=1}^{m-1} w_i}, \tilde{z}_2^* = z_m^*$.

C. PPLS/D

PPLS/D is summarized in Alg. 1.

D. Ablation Study

We now study the effectiveness of different parts of PLS_{GNN}. To show the effectiveness of GNN in PLS_{GNN}, we compare it with several algorithms:

- PLS_{NN}: PLS with MLP $\phi(\cdot, \mathbf{w})$ to control K, but selection is random selection;

Algorithm 1: Parallel Pareto Local Search Based on Decomposition (PPLS/D)

Input: An initial set of non-dominated solutions S_0, maximum budget $budget$, and L the process number and preference weight $\{\lambda_l\}_{l=1}^{L}$.

For each process $l \in \{1, \cdots, L\}$**, do independently in parallel:**
  For $\forall x \in S_0$, $explore(x) \leftarrow$ false;
  $S_l' \leftarrow S_0$; $S_l \leftarrow S_l'$; $FEs \leftarrow 0$;
  while S_l' *is not empty* ***and*** $FEs \leq budget$ **do**
    $x \leftarrow argmax_{x \in S_l'} f^{te}(x|\lambda_l, z^*)$
    $SuccessFlag \leftarrow FALSE$;
    for *each* x' *in the neighborhood of* x **do**
      Evaluate x'; $FEs \leftarrow FEs + 1$;
      if $f^{te}(x'|\lambda_l, z^*) > max_{x \in S_l} f^{te}(x|\lambda_l, z^*)$ **then**
        $S_l \leftarrow S_l \bigcup \{x'\}$; $explore(x') \leftarrow$ false;
        Remove the dominated solutions from S_l;
        $SuccessFlag \leftarrow TURE$;
        Break;
      end
    end
    if $SuccessFlag == FALSE$ **then**
      for *each* x' *in the neighborhood of* x **do**
        if x' *is not dominated by any solution of* S_l **then**
          $S_l \leftarrow S_l \bigcup \{x'\}$; $explore(x') \leftarrow$ false;
          Remove the dominated solutions from S_l;
        end
      end
    end
    $explore(x) \leftarrow$ true;
    $S_l' \leftarrow \{x \in S_l | explore(x) = \text{false}\}$;
  end
return $S \leftarrow \bigcup_{l=1}^{L} S_l$.

- PLS_{NN}-H1: PLS with MLP $\phi(\cdot, \mathbf{w})$ to control K; for selection part, for each solution x_i, the feature $h_i = \frac{1}{2}(f_1^i + f_2^i)$ is used and then aggregate as $y_i = \frac{\sum_{j=1}^{|S'|} e_{ij} h_j}{\sum_{j=1}^{|S'|} e_{ij}}$, as in PLS_{GNN}, where $e_{ij} = \frac{1}{1+dis(x_i, x_j)}$; then, the solution with the largest y_i is selected; and
- PLS_{NN}-H2: the setting is the same as for PLS_{NN}-H1 except for $E = \{e_{ij}\}$ being the identity matrix.

Here PLS_{NN} aims to study the effectiveness of ϕ and PLS_{NN}-H1, PLS_{NN}-H2 aim to study the effectiveness of ψ. Table I shows the results on the two-objective UBQPs. The last line summarizes the average of percentages of improvement comparing with the PLS. PLS_{NN} improves the performance of PLS, and PLS_{NN}-H performs better than PLS_{NN} by adding the heuristic strategy for selection. However, PLS_{GNN} still delivers the best performance. The results show that MLP and GNN both improve performance.

TABLE I
HYPER-VOLUMES ($\times 10^{12}$) FOR PLS, PLS$_{\text{NN}}$, PLS$_{\text{NN}}$-H1, PLS$_{\text{NN}}$-H2, PLS$_{\text{GNN}}$-1, AND PLS$_{\text{GNN}}$-2 ON MUBQPS

Instance	Bud	PLS	PLS$_{\text{NN}}$	PLS$_{\text{NN}}$-H1	PLS$_{\text{NN}}$-H2	PLS$_{\text{GNN}}$-1 $N_d = 1, C_d = +\infty$	PLS$_{\text{GNN}}$-2 $N_d = 1, C_d = +\infty$	PLS$_{\text{GNN}}$-1 $N_d = 5, C_d = 10$	PLS$_{\text{GNN}}$-2 $N_d = 5, C_d = 10$
200-0.8	$5e4$	1.092	1.091	1.104	1.106	1.108	**1.112**	1.107	1.107
	$1e5$	1.103	1.105	1.112	1.111	1.119	**1.123**	1.115	1.115
300-0.8	$3e5$	1.206	1.211	1.209	1.212	1.228	**1.239**	1.219	1.223
	$5e5$	1.216	1.222	1.214	1.214	1.235	**1.240**	1.229	1.227
500-0.8	$8e5$	1.395	1.397	1.417	1.425	1.452	**1.493**	1.418	1.415
	$1e6$	1.409	1.414	1.424	1.423	1.460	**1.495**	1.435	1.422
Avg. of Per. of imp.		0%	0.25%	0.77%	0.91%	2.33%	**3.58%**	1.35%	1.18%