Unconstrained Many-Objective Nurse Scheduling

- Twelve objectives are defined by means of the penalties $p_k$.

$$ f_k = \begin{cases} \left(1 + \sum_{i=1}^{N} p_{ki}\right)^{\frac{1}{\epsilon}} & k = 1,2,\ldots,6 \\ \left(1 + \sum_{i=1}^{D} p_{ki}\right)^{\frac{1}{\epsilon}} & k = 7,8,\ldots,12 \end{cases} $$

- The problem is formulated as $\mathcal{C}$MOOP.

$$ \max \cdot f(x) = [f_1(x), f_2(x), \ldots, f_2(x)]^T $$

- The initialization and the following operators release the problem constraints.

**Discussion**

- Distribution of $\text{FFS}$ at final generation:

  Depending on the biaxial plane of interest, we have observed cases where PPD-NSGA-II (proposed) has been advantageous, NSGA-II (conventional) has been advantageous, and both superiority/inferiority has not been determined.

- Distribution of $\text{FFS}$ on $s_1$-$s_2$ plane with generation progression:

  FFS given by PPD-NSGA-II converges to POS faster than NSGA-II.

  However, several solutions by PPD-NSGA-II have been lost at the final generation. Since the observation has been made on the S2-S1 plane, the lost solutions are not necessarily good, but it is considered that some degradation has occurred.

- Comparison of Norm and MS values of FFS at the final generation:

  For convergence to POS (Norm), PPD-NSGA-II is advantageous, and the spread of the solution set is almost equivalent.

**Improvement of NSGA-II based on Pareto Partial Dominance (PPD)** that makes it easier to determine the dominance/inferiority relationships between individuals [Sato2010].

- The subset size of the objective functions to be used for Pareto partial dominance is required to specify before the optimization in a form of a combination list.

- An appropriate value of the subset size according to the complexity of the problem is unknown.

The contents of the combination list greatly influence the result. Furthermore, the creation of the combination list requires the user to have a high skill.

- NSGA-II-PPD performs ND sorting for all objective functions at a specific generation cycle, and preserves parents as an archive set $A$ for the next generation. This process is able to generate child individuals having the same contents as the already existing individual in $A$. As a result, the same individuals increases in the first front set.

**Improvement #1: Subset Size Scheduling (SSS)**

- The combination list is eliminated from the algorithm.

**SSS is improved.** The parameter $r$ is given by the following equations.

**Improvement #2: Killing Unwanted Individuals Given by Mating**

- Compare each individual in $C_i$ created by the mating and all individuals in $A$.

- If it is an individual with the same content, kill it.

- This can be easily done by setting the value of all objective functions of such the individual to 0.

- Thus, it is strictly necessary to determine the dominance/inferiority relations between all individuals.

- The penalty for the objective function $f_k$ is determined by the following equation:

  $$ f(x) - f(y) \geq \epsilon \Rightarrow \exists \epsilon \in R : f_k(x) > f_k(y) $$

- Although a framework of NSGA-II can be applied to this problem, too many objective functions. $\Rightarrow \text{uMaOP}$

**Nurse Scheduling in Actual Hospital**

- Distribution of the first front set (FFS) at the final generation expressed on biaxial planes sliced out from the hyper space with 12 objective functions.

**Pareto Partial Dominance with Linear Subset Size Scheduling**

- Sato’s $\text{PPD}$ makes it easier to determine the dominance/inferiority relationships between individuals [Sato2010].

- The subset size of the objective functions to be used for Pareto partial dominance is required to specify before the optimization in a form of a combination list.

- An appropriate value of the subset size according to the complexity of the problem is unknown.

The contents of the combination list greatly influence the result. Furthermore, the creation of the combination list requires the user to have a high skill.

- NSGA-II-PPD performs ND sorting for all objective functions at a specific generation cycle, and preserves parents as an archive set $A$ for the next generation. This process is able to generate child individuals having the same contents as the already existing individual in $A$. As a result, the same individuals increases in the first front set.

**Improvement #1: Subset Size Scheduling (SSS)**

- The combination list is eliminated from the algorithm.

**SSS is improved.** The parameter $r$ is given by the following equations.

**Improvement #2: Killing Unwanted Individuals Given by Mating**

- Compare each individual in $C_i$ created by the mating and all individuals in $A$.

- If it is an individual with the same content, kill it.

- This can be easily done by setting the value of all objective functions of such the individual to 0.

**Discussion**

- Distribution of $\text{FFS}$ at final generation:

  Depending on the biaxial plane of interest, we have observed cases where PPD-NSGA-II (proposed) has been advantageous, NSGA-II (conventional) has been advantageous, and both superiority/inferiority has not been determined.

- Distribution of $\text{FFS}$ on $s_1$-$s_2$ plane with generation progression:

  FFS given by PPD-NSGA-II converges to POS faster than NSGA-II.

  However, several solutions by PPD-NSGA-II have been lost at the final generation. Since the observation has been made on the S2-S1 plane, the lost solutions are not necessarily good, but it is considered that some degradation has occurred.

- Comparison of Norm and MS values of $\text{FFS}$ at the final generation:

  For convergence to POS (Norm), PPD-NSGA-II is advantageous, and the spread of the solution set is almost equivalent.