Cooperative Co-evolution Strategies with Time-dependent Grouping for Optimization Problems in Smart Grids

Junpeng Su[1], Han Huang[1], Zhifeng Hao[2]

1. South China University of Technology   2. Shantou University

A number of real-world optimization problems are still outstanding in the energy domain [1]. For example, the bidding optimization problem in day-ahead local energy markets aims at the minimization of end-users' costs [2]. Moreover, the flexibility management of home appliances can be modeled as a mixed-integer nonlinear programming problem [3]. To solve these optimization problems, cooperative coevolution strategies with time-dependent grouping (CCS-TG) are proposed.

[1] Keke Gai, Yulu Wu, Liehuang Zhu, Meikang Qiu, and Meng Shen. 2019. Privacy-preserving energy trading using consortium blockchain in smart grid. IEEE Transactions on Industrial Informatics 15, 6 (2019), 3548-3558.
[2] Fernando Lezama, Joao Soares, and Zita Vale. 2019. Optimal bidding in local energy markets using evolutionary computation. In 2019 20th International Conference on Intelligent System Application to Power Systems (ISAP). IEEE, New Delhi, India, 1-6.
[3] Fernando Lezama, Joao Soares, Bruno Canizes, and Zita Vale. 2020. Flexibility management model of home appliances to support DSO requests in smart grids. Sustainable Cities and Society 55 (2020), 1-12.

Algorithm 1 presents the search procedure of cooperative co-evolution strategy with time-dependent grouping for optimal bidding strategy in local energy markets. Inspired by cooperative co-evolution frameworks, the grouping function *time_dependent_grouping(n, lbs, ubs)* groups the decision variables into subcomponents based on the time period in the day-ahead market. The operator *quantity_search* adjusts the bid quantities, while the operator *price_search* searches for the optimal bid price.

Algorithm 1 CCS-TG-I

1: $groups \leftarrow time\_dependent\_grouping(n, lbs, ubs)$;
   /* lbs and ubs are variables boundaries, n represents the problem dimension */
2: Generate an initial solution X randomly;
3: Evaluate X and record the best-ever solution X^*;
4: **while** termination criterion is not met **do**
5:    $\xi \leftarrow randi(size(groups))$;
6:    $indicies \leftarrow groups[\xi]$;
7:    $X_c \leftarrow quantity\_search(X, indicies)$;
8:    $X_c \leftarrow price\_search(X_c, indicies)$;
9:    Evaluate X_c and record the best-ever solution X^*;
10:   $X \leftarrow elitist\_selection(X_c, X)$;
11: **end while**
12: **return** the best-ever solution X^*;

Algorithm 2 CCS-TG-II

1: $groups \leftarrow time\_dependent\_grouping(n, lbs, ubs)$;
2: Randomly generate an initial population P of size N_p;
3: Evaluate each solution X in P and record the best-ever solution X^*;
4: **while** termination criterion is not met **do**
5:   **for** each solution X in P **do**
6:     $X_c \leftarrow starting\_period\_search(X)$;
7:     **for** $i \leftarrow 1$ to $size(groups)$ **do**
8:       $indicies \leftarrow groups[i]$;
9:       $X_c \leftarrow intensity\_search(X, indicies)$;
10:     **end for**
11:     Evaluate X_c and record the best-ever solution X^*;
12:     $X \leftarrow elitist\_selection(X_c, X)$;
13:   **end for**
14:   Reduce the population size when the number of fitness evaluations exceeds the threshold ζ;
15: **end while**
16: **return** the best-ever solution X^*;

The pseudocode of CCS-TG-2 for flexible management problem of home appliances is given in Algorithm 2. Different from CCS-TG-1, the grouping function in CCS-TG-2 is not applied to all decision variables, but groups the intensity variables of real-time devices based on the time period. Specifically, CCS-TG-2 first randomly modifies the starting period of shifting devices, and then searches for the optimal intensities of appliances with reduction capabilities.

THANK YOU