Co-operative Prediction Strategy for Solving Dynamic Multi-Objective Optimization Problems

Zhihao ZHAO
Department of Computer Science
Hong Kong Baptist University
Hong Kong SAR, China
E-mail: 17250072@life.hkbu.edu.hk

Fangqing Gu
School of Applied Mathematics
Guangdong University of Technology
Guangdong, China
E-mail: fqgu@gdut.edu.cn

Yiu-ming Cheung
Department of Computer Science
Hong Kong Baptist University
Hong Kong SAR, China
E-mail: ymc@comp.hkbu.edu.hk

***Abstract*—Prediction-based evolutionary multi-objective optimization algorithm is one of the most popular optimization algorithms for solving dynamic multi-objective optimization problem. It uses time-series models to predict the future Pareto set based on the past solutions. However, the dimension of the decision variables may be too high to predict. Moreover, a relatively small variance in decision variables may lead to a large difference in the objective space. The optimized Pareto front (PF) may be far from the desired output. To solve these problems, this paper proposes a new co-operative prediction method, which predicts not only the Pareto solution (PS), but also a hyper-plane as an approximation of the prediction of the PF in the objective space. The hyper-plane is used to guide the search process and accelerate the convergence. We compare the proposed algorithm with three existing dynamic optimization algorithms. Experimental results show the effectiveness of the proposed algorithm.**

***Index Terms*—Evolutionary Algorithms, Dynamic Multiobjective Optimization**

I. Introduction

Multi-objective optimization problems refer to the optimization problem with two or more optimization objective functions [1]–[3], which usually conflict to each other. Dynamic multi-objective optimization problem (DMOP) further enhances these definitions that the optimization objectives and constraints are time-varying. Time-varying characteristic brings us great challenge to deal with these problems. Recently, DMOP has received increasing interest in both of industry and academia. Typical works related to DMOP are scheduling [4]–[6], planning [7], [8], control problems [9], [10], and resource allocation optimization [11], [12]. Nevertheless, DMOPs exhibit the difficulty in the aspects of balance between diversity and convergence speed [13], changing environments, and so on.

In general, evolutionary algorithm is the most typical algorithm for solving multi-objective problems. Subsequently, several dynamic multi-objective evolutionary algorithms have been proposed to solve these problems from the different aspects, including memory-based methods [14], classifier-initialization-based methods [15], local search strategies [16] and prediction-based methods [13], [17]. Different strategies aim to focus on different aspects of the problem. Especially, memory-based method utilizes the periodicity of the data stream, e.g. seasonal financial data. Classifier-initialization-based methods filter out the dissatisfied solutions by individual filtering or iterative population updating, towards speeding up the convergence and enhancing the accuracy. After the typical optimization is done, some algorithms adopt local strategy for better convergence and accuracy [18]. For example, solutions are adjusted after each evolutionary steps [19]. Experiments have shown that, when a global search is incorporated with local search, both fast convergence and high accuracy can be achieved.

The prediction-based method utilizes the historical information to predict the potential future Pareto solution (PS). In some cases, the prediction-based methods can achieve fast convergence as well as maintaining a diverse population [13]. Nevertheless, most of the existing prediction-based methods focus only on the straightforward way of PS prediction. Since there is no direct correlation between Pareto front (PF) and PS, a small variance of PS can lead to a considerable variance

978-1-7281-6929-3/20/$31.00 ©2020 IEEE

and error in PF. As far as we know, there are few studies on the prediction of the PS and PF to achieve fast convergence and high accuracy.

In this paper, we therefore propose a novel method called cooperative-prediction guided search strategy with concept drift detection and restarting. Specifically, we propose a novel population searching strategy to predict the PS. Furthermore, a dominant hyperplane which can be seen as the approximation of PF is used to adjust the potential error and accelerate the convergence in an early stage after the changes have been detected. At the beginning of every time step, an initial population is predicted by a PS prediction model, and then the hyperplane guided local search is carried out to adjust the solutions for better convergence locally. The proposed prediction model is in the form of $PF(x_t) = f(x_{t-1}, x_{t-2}, \ldots, x_{t-m})$, where m is the length of the time series data used for prediction. Typically, when the environment is stable, i.e. prediction model $f(.)$ does not change over time t, all the historical data follows the same changing patterns. However, when concept drift occurs, the time series $(x_i, x_{i+1}, x_{i+2}, \ldots,)$ and $(x_j, x_{j+1}, x_{j+2}, \ldots,)$ will have the significant different autocorrelation. Under the circumstances, several works have been done towards solving the concept drift problem in time series data [20]–[22]. For simplicity, we adopt the detection framework conducted by Cavalcante et al. [22] that generally divides the process into feature extraction and concept drift test. Subsequently, we present some test problems, where the test process is divided into several stages. To demonstrate the robust performance of the proposed method, we compare the proposed algorithm with three existing counterparts.

The remainder of this paper is organized as follows: Section II gives the detailed description of the proposed algorithm. Section III provides the empirical studies of the proposed algorithm in comparison with the existing ones. Finally, we draw a conclusion in Section IV.

II. The Proposed Co-operative Prediction Strategy

We propose a co-operative prediction strategy (COPS) for solving DMOP problems. Accordingly, Algorithm 1 describes the framework of the proposed algorithm, denoted as COPS-DMOP for short. A new selection operator called hyper-plane distance based selection operator is presented. The optimization process is divided into two stages. The first stage is the fast adaption and fine-tuning stage, for which the distance-based selection (DBS) operator is proposed to achieve a satisfying global distribution. The basic idea of this DBS operator is that a hyper-plane is predicted as an approximation of the PF, and the distance of every individual of the population of the plane is calculated. The closer of the distance between the individual to the plane, the similar it is to the actual distribution of the PS. Moreover, in order to prevent local optimum, the population is divided into several portions along the objective axis with the maximum variance. Only a limited number of individuals with the smallest distance values in every portion are selected. The second stage uses the typical evolutionary algorithm to output the final non-dominated solutions.

Algorithm 1 The proposed COPS-DMOP algorithm

Input:
1) The population size: N.
2) The maximum number of generations max_gen.
3) The number of generations of the first stage T_1.

Output:
The non-dominated solution in population P_t

Initialization:
Uniformly choose N individuals from decision space as the initial population P_0, set $flag = 0$ and $t = 1$.
while $t < max\_gen$ **do**
  if change is not detected **then**
    if $t - flag \leqslant T_1$ **then**
      $P_t = DBS(\Theta, P_{t-1})$
    else
      Optimize the problem by using NSGA-II to obtain P_t.
    end if
  else
    $flag = t$.
    Apply PPS [13] to update population and get the new population P_t.
    Update the parameter Θ of the hyper-plane according to its definition.
  end if
  $t = t + 1$.
end while

Algorithm 2 DBS Operator

Input:

1) The hyper-plane parameters Θ.

2) The old population P_{old} with a size N.

Output:

The new population P_{new}.

```
Initialization:
  Equally partition the population P_old into pr sub-populations P_in^1, P_in^2, P_in^3, ..., P_in^pr along objective axis with the largest variance.
for each P_old^i do
    Apply gaussian mutation on each individual of P_old^i to generate new solutions Q_old^i.
    Calculate the distance of every individual of P_in^i ⋃ Q_old^i to the hyperplane with parameters Θ.
    Sort the distances of sub-population.
    Select N/pr individuals with the smallest distance to hyperplane with parameters Θ and set them as P_s^i.
    P_new = P_new ∪ P_s^i.
end for
```

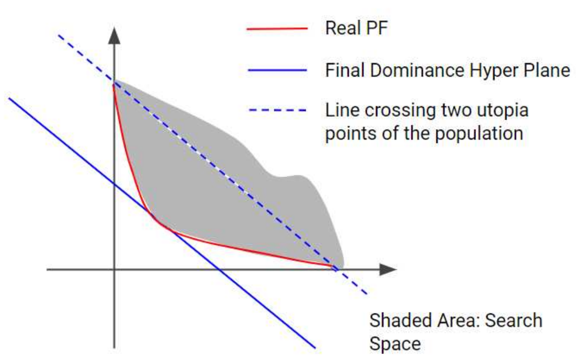

Figure 1: Example of the dominant hyper-rectangle for bi-objective problems.

A. Definition of Dominance Hyper-Plane

As the topology of the hyperplane is relatively simple, we therefore use the simplified hyperplane called dominant hyperplane to simulate the real PF and guide the search process. The definition of the dominant hyper-rectangle is as follows: Given a population P and a hyper-rectangle H, when no individual in P dominates any solutions on H, we call H as the dominant hyper-rectangle of P. This basically ensures that the closer solution is to the hyper-rectangle, the better accuracy it has. For bi-objective problems, the hyper-rectangle is simply the tangent line to the PF formed by current population, as illustrated in Figure 1. The dominant hyperplane should be parallel to the plane crossing the utopia points which have the optimal values in corresponding objectives.

B. Prediction Parameters and Models

The proposed method is inspired by population prediction strategy (PPS), which regards the population as a combination of a central point and scaffold, where the central point is predicted using the typical time-series models and scaffold is calculated based on the variance of the population in the past time steps. We adopt the idea of PS prediction used in PPS, but in order to speed up the convergence and adjust the population using information from the PF, the hyperplane is also predicted for the use of DBS operator. Theoretically speaking, any time-series models can be used. Here, for simplicity, we utilize Autoregression model to predict the parameter of the hyperplane. The parameters of the hyperplane are predicted independently (e.g. for bi-objective problems, the expression of plane is $Ax + By + C = 0$, and the parameters A, B, C are predicted independently). One important thing is that not all the parameters of the hyperplane should be predicted. Only those related to the normal vector should be predicted according to the definition of the dominant hyperplane.

C. Co-operative Prediction Strategy

At the start of every generation, the proposed COPS-DMOP first detects whether there are changes in optimization problem. If a change is detected, population for next generation is predicted as specified in Algorithm 1, and the hyperplane parameters are also predicted based on historical data using the stored historical parameters with the same time span as population prediction.

The COPS divides the optimization process in every time step into two stages. The first stage adopts the proposed DBS operator for a fixed number of T_1 generations for fast-adaption and fine-tuning. In DBS, the population is first doubled by self-copying, and then Gaussian mutation with high mutation rate is applied. The function of Gaussian mutation is to randomly search through the nearby decision space and the fine tuning of the population in guidance of the predicted hyper-plane. The second step of DBS is to divide the population into pr

Table I: The details of concept drift problems

Concept Drift Type	Generations(10^3)	Problem Type	n_t(parameter)
Correlation Changes (CRC)	0-2	FDA3	10
	2-4	ZJZ	10
	4-6	FDA2	10
Magnitude Changes (MAC)	0-2	ZJZ	5
	2-4	ZJZ	10
	4-6	ZJZ	2
Correlation -Magnitude (CMC)	0-2	FDA3	10
	2-4	ZJZ	2
	4-6	FDA2	5

[a]Concept Drift Problems (corresponding to CRC, MAC, CMC).

Table II: The value of MIGD metrics obtained by the compared algorithms.

Instance	NO	RIND	PPS	PPR	COPS-DMOP
FDA2	10	0.3284	0.3284	**0.3277**	0.3284
	20	0.3279	0.3279	**0.3273**	0.3279
	50	0.3150	0.3150	**0.3140**	0.3150
FDA3	10	0.1567	2.4534	0.1158	**0.0721**
	20	0.0436	1.0192	0.0213	**0.0197**
	50	0.0066	0.0143	0.0050	**0.0046**
ZJZ	10	0.3879	0.2584	0.3856	**0.1918**
	20	0.3670	0.1865	0.3436	**0.1534**
	50	0.2540	**0.1137**	0.2316	0.1218
FDA4	10	0.1007	0.7958	**0.0953**	0.1624
	20	0.0761	0.4399	0.0740	**0.0705**
	50	0.0665	0.1801	0.0660	**0.0653**
FDA5	10	0.1550	0.8670	**0.1513**	0.7504
	20	0.1190	0.5197	0.1173	**0.1166**
	50	0.1070	0.2572	0.1068	**0.1057**
CRC	10	0.3191	0.2712	0.3290	**0.1897**
	20	0.2492	**0.1420**	0.2320	0.1460
	50	0.2017	**0.1311**	0.1900	0.2008
MAC	10	0.3994	0.3035	0.4182	**0.2131**
	20	0.3840	**0.2144**	0.3846	0.2268
	50	0.3161	**0.1512**	0.2975	0.1868
CMC	10	0.3256	0.2846	0.3223	**0.2460**
	20	0.2595	0.2015	0.2633	**0.2002**
	50	0.2290	**0.1789**	0.2317	0.1873

sub-populations along objective axis with the largest variance. Then, in every sub-population, we select N/pr individuals with the smallest distance to the predicted hyperplane, thus N individuals in total are selected and returned for use in the next generation.

A half-qualified population is passed to the second stage through the initial stage. This population is not non-dominance set, but it has generally good global distribution and can prevent local optimal and the distance selector can ensure the overall shape of the population in the objective space. The second stage uses the typical EMO algorithm: NSGA-II.

Furthermore, taking into consideration that the prediction model is valid only when the problem type remains unchanged (i.e. it is predictable with the same changing pattern), we add a simple concept drift p-test when the change of the problem is detected. We adopt the framework proposed by Cavalcante et al. [22]. The Mean Square Error (MSE) of the central point prediction model is recorded as the feature of the time series in the current time step. Subsequently, in the next time step, a p-test is conducted using the historical MSE and the current error. If there is a sharp changing which does not pass the p-test, all the historical information will be removed including the historical PS and the historical dominance-hyper-plane.

III. Experimental Results

A. Performance Metrics, Test instances and Settings

In our study, we compare the proposed algorithm with the other prediction-based methods. We use the variant of inverted generational distance (IGD) called MIGD as the performance indicator to evaluate and compare the quality of solutions obtained by different algorithms. The inverted generational distance (IGD) is an index to measure the distance between the real Pareto Optimal Front represented by P^{t*} at time t , and the approximate Pareto Optimal Front P^t obtained by the algorithm. P^{t*} and P^t are both at the end of every time step (i.e. the last generation before the environment changes). The definition of IGD can be expressed as:

$$IGD\left(P^{t*}, P^t\right) = \frac{\sum_{v \in P^{t*}} d\left(v, P^t\right)}{|P^{t*}|}. \tag{1}$$

We adopt the variant MIGD of IGD defined in [17], which is the means of inverted generational distance. It evaluates the quality of the population after the optimization step in every time step. The definition of MIGD can be stated as:

$$\text{MIGD} = \frac{1}{|T|} \sum_{t \in T} \text{IGD}\left(P^{t*}, P^t\right). \tag{2}$$

In this paper, we take some typical benchmark problems widely used in other researches conducted in [13]. Furthermore, in order to test its performance over concept drift environment, we give 3 new test problems modified

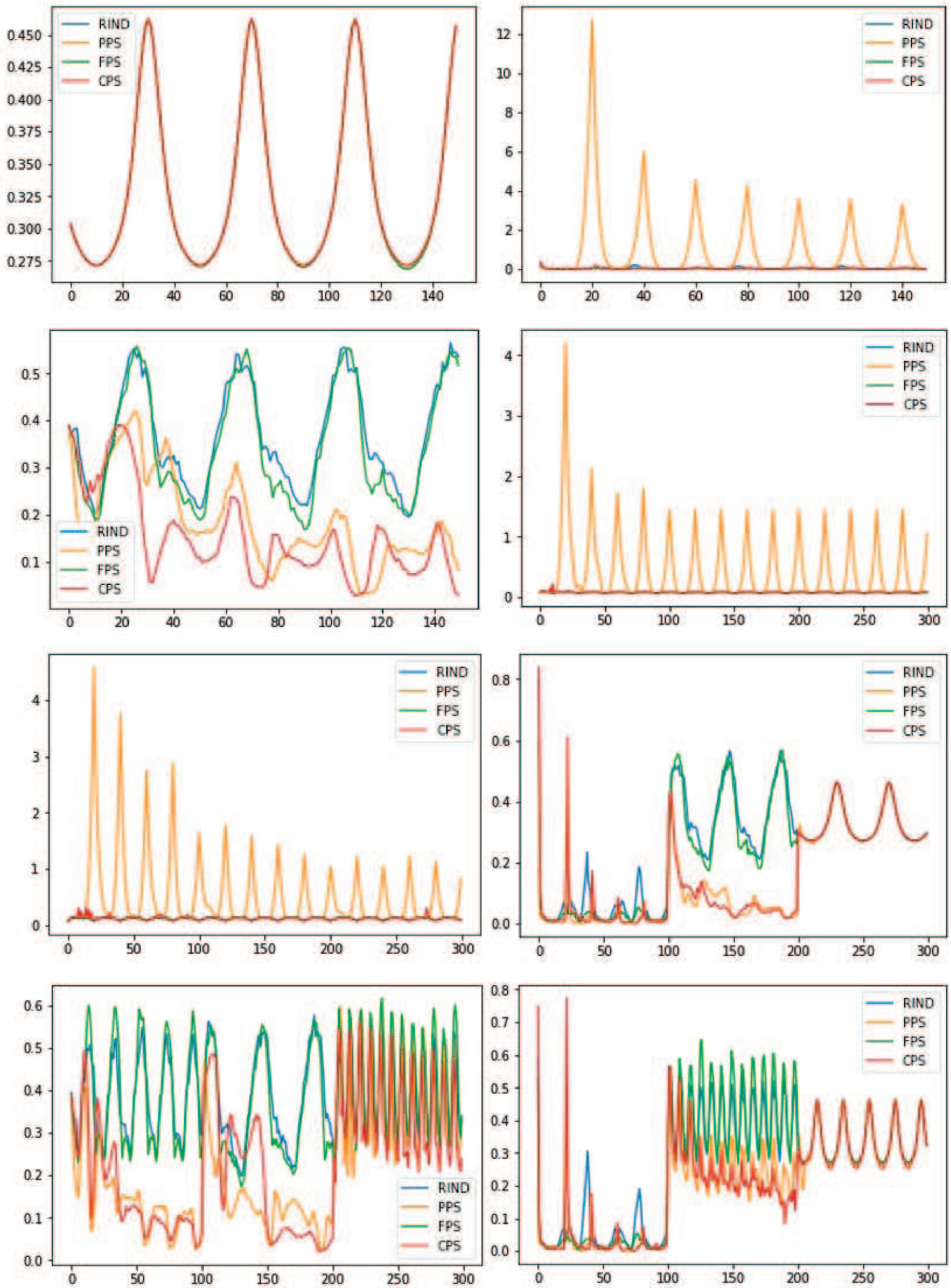

Figure 2: IGD-Generation when the number τ_t of generations is set at 20 (The same order as Table II, from left to right, from top to bottom).

from the benchmark problems: correlation changing problem, varied changing the magnitude, and the aggregation of these two changes. The detailed information is stated in Table I. In general, different problems have the different PF shape with the varied distributions, and the same problem type can have different severity of changing with all similar PF (i.e. population distribution). In all experiments, the population size is set at 100, and $t = \frac{1}{n_t} \left\lfloor \frac{\tau}{\tau_T} \right\rfloor$ where n_t, τ, τ_t are the severity of changes, current generation, and number of generations, respectively. For FDA and ZJZ problems, n_t is set at 10. As for concept drift problems, n_t may change according to Table I. In DBS, the hypermutation rate is set at 0.9 by a rule of thumb, which only keeps a small portion unchanged for EA step. In addition, the partition size is set at 50, with 2 individuals selected in every partition. In order to meet the requirements of all problem settings, the fine-tuning generation in COPS is set to at $T_1 = 5$.

B. Experimental Results

We conducted two experiments: (1) the common FDA and ZJZ problems, and (2) the concept drift problems. The first experiment evaluates the general performance of the proposed method over the different kinds of test problems, while the second experiment is to evaluate its performance when the changing pattern changes over time, and to test the robustness when sudden unpredictable changes occur. Subsequently, The proposed COPS-DMOP is compared with the existing three algorithms, i.e. Randomly Reinitialization (RIND), PPS, and Prediction-Based Population Re-initialization (PPR). We take the average of 5 runs as the final results, with 3000 generations in each run for 2 objective problems, and 6000 generations for 3 objective and concept drift problems. The value of MIGD results are listed in Table II, and the IGD-generation graph is also provided in Figure 2. It can be seen that the performance of the proposed algorithm is competent in comparison with the existing counterparts. This implies that the predicted hyperplane can guide the search process, whereby accelerating the convergence and correcting the prediction errors. Also, error tolerance has been improved compared to the existing methods because the proposed COPS-DMOP has an error correction mechanism while the others not. In addition, as COPS-DMOP takes the concept drift into consideration, it turns out that the COPS-DMOP has a better result in an unstable environment.

IV. Conclusion

We have proposed the COPS-DMOP algorithm, which not only predicts not only the PS, but also estimates the PF by a hyperplane. The distance between the individual and hyperplane is used as the selection operator to replace the non-dominance sorting selection operator in an early stage of optimization. The hyperplane has been used to guide the search process and accelerate the convergence. We compared the proposed algorithm with three dynamic optimization algorithms. Experimental results have shown the effectiveness of the proposed algorithm.

Acknowledgment

This work was supported by the National Science Foundation of China under Grants: 61672444 and 61703108, by Hong Kong Baptist University (HKBU), Research Committee, Initiation Grant, Faculty Niche Research Areas (IG-FNRA) 2018/19 under Grant RC-FNRA-IG/18-19/SCI/03, by the Innovation and Technology Fund of Innovation and Technology Commission of the Government of the Hong Kong SAR under Project ITS/339/18, by the SZSTI under Grant JCYJ20160531194006833, and by the Natural Science Foundation of Guangdong Province 2017A030310467.

References

[1] H.-L. Liu, F. Gu, and Q. Zhang, "Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 3, pp. 450–455, 2014.

[2] Y.-M. Cheung, F. Gu, and H.-L. Liu, "Objective extraction for many-objective optimization problems: algorithm and test problems," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 5, pp. 755–772, 2016.

[3] H.-L. Liu, Y. Wang, and Y.-m. Cheung, "A multi-objective evolutionary algorithm using min-max strategy and sphere coordinate transform," *Automation and Soft Computing*, vol. 15, no. 3, pp. 361–384, 2009.

[4] K. Deb, U. B. Rao N., and S. Karthik, "Dynamic multi-objective optimization and decision-making using modified nsga-ii: A case study on hydro-thermal power scheduling," in *Evolutionary Multi-Criterion Optimization*, 2007, pp. 803–817.

[5] M. Abello, L. Bui, and Z. Michalewicz, "An adaptive approach for solving dynamic scheduling with time-varying number of tasks, part II," in *2011 IEEE Congress of Evolutionary Computation*, 07 2011, pp. 1711 – 1718.

[6] J. Eaton, S. Yang, and M. Gongora, "Ant colony optimization for simulated dynamic multi-objective railway junction rescheduling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 2980–2992, 2017.

[7] M. Mavrovouniotis, F. M. Muller, and S. Yang, "Ant colony optimization with local search for dynamic traveling salesman problems," *IEEE Transactions on Cybernetics*, vol. 47, no. 7, pp. 1743–1756, 2017.

[8] M. Liu, "Robotic online path planning on point cloud," *IEEE Transactions on Cybernetics*, vol. 46, no. 5, pp. 1217–1228, 2016.

[9] B. Andrés-Toro, J. Giron-Sierra, P. Fernández, J. Lopez-Orozco, and E. Besada, "Multiobjective optimization and multivariable control of the beer fermentation process with the use of evolutionary algorithms," *Journal of Zhejiang University. Science*, vol. 5, pp. 378–389, 2004.

[10] Z. Zhang, "Multiobjective optimization immune algorithm in dynamic environments and its application to greenhouse control," *Applied Soft Computing*, vol. 8, pp. 959–971, 2008.

[11] W. K. Mashwani and A. Salhi, "Multiobjective evolutionary algorithm based on multimethod with dynamic resources allocation," *Applied Soft Computing*, vol. 39, pp. 292–309, 2016.

[12] P. A. Navrátil, H. Childs, D. S. Fussell, and C. Lin, "Exploring the spectrum of dynamic scheduling algorithms for scalable distributed-memoryray tracing," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 6, pp. 893–906, 2014.

[13] A. Zhou, Y. Jin, and Q. Zhang, "A population prediction strategy for evolutionary dynamic multiobjective optimization," *IEEE Transactions on Cybernetics*, vol. 44, no. 1, pp. 40–53, 2014.

[14] C. Goh and K. C. Tan, "A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 103–127, 2009.

[15] W. Hu, M. Jiang, X. Gao, K.-c. Tan, and Y.-m. Cheung, "Solving dynamic multi-objective optimization problems using incremental support vector machine," in *2019 IEEE Congress on Evolutionary Computation (CEC)*, 2019, pp. 2794–2799.

[16] M. Mavrovouniotis, F. Neri, and S. Yang, "An adaptive local search algorithm for real-valued dynamic optimization," in *2015 IEEE Congress on Evolutionary Computation*, 2015, pp. 1388–1395.

[17] A. Muruganantham, K. C. Tan, and P. Vadakkepat, "Evolutionary dynamic multiobjective optimization via kalman filter prediction," *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 2862–2873, 2016.

[18] A. Lara, G. Sanchez, C. A. C. Coello, and O. Schutze, "Hcs: A new local search strategy for memetic multiobjective evolutionary algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, pp. 112–132, 2010.

[19] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 28, no. 3, pp. 392–403, 1998.

[20] G. Oliveira, R. Cavalcante, G. Cabral, L. Minku, and A. Oliveira, "Time series forecasting in the presence of concept drift: A pso-based approach," in *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, 2017, pp. 239–246.

[21] M. Lavielle and G. Teyssière, "Detection of multiple change-points in multivariate time series," *Lithuanian Mathematical Journal*, vol. 46, no. 3, pp. 287–306, 2006.

[22] R. C. Cavalcante, L. L. Minku, and A. L. I. Oliveira, "FEDD: feature extraction for explicit concept drift detection in time series," in *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, pp. 740–747.