2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems

A Parallel Hybrid Intelligent Algorithm for Fuzzy Mean-CVaR Portfolio Model

Chen Li
Computer Network Information Center, Chinese Academy of Science and University of Chinese Academy of Sciences
Beijing, China
lichen@sccas.cn

Zhonghua Lu
Computer Network Information Center, Chinese Academy of Science
Beijing, China
zhlu@cnic.cn

Yonghong Hu
Central University of Finance and Economics
Beijing, China
huyonghong@cufe.edu.cn

Fang Liu
Computer Network Information Center, Chinese Academy of Science
Beijing, China
liufang@sccas.cn

Jue Wang
Computer Network Information Center, Chinese Academy of Science
Beijing, China
wangjue@sccas.cn

***Abstract*—In recent years, fuzzy portfolio selection theory has been well developed and widely applied. Based on the credibility theory, several fuzzy portfolio selection models have been proposed. The fuzzy Mean-CVaR portfolio model is one of the state-of-the-art. However, its' fuzzy nature which increases the computational complexity makes it take a long time to solve. In order to solve the fuzzy Mean-CVaR portfolio model efficiently, a hybrid intelligent algorithm is designed by integrating Genetic Algorithm(GA) with adaptive penalty function, Simulated Annealing Resilient Back Propagation (SARPROP) neural network and fuzzy simulation techniques, and to accelerate the computation speed further, we parallelize the hybrid intelligent algorithm with MPI technology. In order to demonstrate its validity and efficiency, we achieve numerical experiments on the Era supercomputer, and the results are compared with the method which is obtained by integrating traditional GA and fuzzy simulation directly. The results show that hybrid intelligent algorithm can get better performance. Experiments under different processor cores also achieved on the Era supercomputer demonstrate the scalability of the parallel hybrid intelligent algorithm.**

Keywords—Parallel computing, Fuzzy portfolio selection, Fuzzy simulation, SARPROP neural network, GA

I. INTRODUCTION

Portfolio selection theory focuses on the problem that how to generate the optimal portfolio to maximize the investment return. As the key to provide scientific and rational investment strategy for investors, portfolio selection models has been developed with the development of the portfolio selection theory. In 1952, Markowitz [1] proposed the Mean-Variance portfolio model, which opened the door to modern portfolio theory. Since then, many researchers [2-9], including Markowitz, have devoted themselves to the further improvement of the Mean-Variance model. Many other portfolio selection models such as Mean-Semivariance model [2], Chance Constrained model [3], Mean-Absolute Deviation model [5], Mean-Variance-Skewness model [7], Mean-VaR(Value at Risk) model [8] and so on were successively proposed.

The securities market has two significant features: one is randomness, the other is fuzziness. However, before the introduction of fuzzy set theory, most of the portfolio models were based on the stochastic theory, which ignored the fuzziness of the securities market. In 1965, Zadeh [10] proposed the concept of fuzzy sets, and since then, some scholars [11-16] began to realize that the returns of securities should be described as fuzzy variables and use fuzzy models to solve portfolio

978-1-5386-2588-0/17 $31.00 © 2017 IEEE
DOI 10.1109/HPCC-SmartCity-DSS.2017.46

selection problems. Quantifying investment return by expected value and defining the investment risk as CVaR in fuzzy environment, the fuzzy Mean-CVaR portfolio model [17] is one of the state-of-the-art.

Unfortunately, while the fuzzy Mean-CVaR portfolio model can be closer to the actual investment scenario and more accessible to apply, it suffers from the same problem faced by all fuzzy models: it will be difficult to solve the model by traditional mathematical programming methods since the model contains fuzzy variables. Some scholars [18] [19] [20] attempted to address this problem by using traditional GA and fuzzy simulation directly. However, although the optimal solution of the fuzzy models can be obtained, there still exist some limitations for this method to deal with the computational complexity in the fuzzy models, which can't meet the needs of the actual transaction. So, it is of great urgency to provide a new method to solve the fuzzy Mean-CVaR model efficiently.

Recent years have witnessed dramatic changes in High-performance Computing (HPC) to accommodate the increasing computational demand of scientific applications. With the help of HPC technology, the solving process of the fuzzy Mean-CVaR model can be speeded up further

The main contributions of this paper are as follows:

(1) We propose a hybrid intelligent algorithm by integrating GA with adaptive penalty function, SARPROP neural network and fuzzy simulation for solving the fuzzy Mean-CVaR portfolio optimization model efficiently.
(2) We parallelize the hybrid intelligent algorithm with MPI technology to accelerate the computation speed of the fuzzy Mean-CVaR model further.
(3) Based on Shanghai Stock Exchange (SSE 180) dataset and Shanghai Stock Exchange 50 (SSE 50) dataset, we conduct several numerical experiments on the Era supercomputer, and analyze the results rationally. The comparison experiments show that the hybrid intelligent algorithm is effective and valid. Experiments under different processor cores demonstrate that the parallel hybrid intelligent algorithm is scalable.

The rest of this paper is organized as follows. Section II provides an overview of the fuzzy Mean-CVaR model. In Section III, we present the hybrid intelligent algorithm based on SARPROP neural network and GA with adaptive penalty function. The parallel hybrid intelligent algorithm with MPI is shown in Section IV. In Section V, we detail the relevant numerical experiments and analyze the results. Some conclusions are drawn in Section VI.

II. FUZZY MEAN-CVAR PORTFOLIO MODEL

If we describe the returns of securities as fuzzy variables, quantify investment return by expected value and define the investment risk as CVaR, then we obtain the following fuzzy Mean-CVaR model, which was proposed by [17]:

$$\begin{cases} \min \xi_{\mathrm{CVaR}}(\alpha) \\ \text{s.t. } \mathrm{E}\left[\sum_{i=1}^{n} x_i \xi_i\right] \geq r \\ \sum_{i=1}^{n} x_i = 1 \\ 0 \leq x_i \leq 1,\ i=1,2,\ldots,n \end{cases} \tag{1}$$

Where x_i denotes the investment proportion in security i, and ξ_i which is set to triangular fuzzy number, represents the predicted return of the i[th] security for i = 1, 2, ..., n, respectively. $\mathrm{E}\left[\sum_{i=1}^{n} x_i \xi_i\right]$ is the expected value of return on the portfolio, $\xi_{\mathrm{CVaR}(\alpha)}$ represents the CVaR operator which is defined by formula (2) [17] for fuzzy variable $\sum_{i=1}^{n} x_i \xi_i$. r is the minimum value of the return required by the investor, and α is the predetermined confidence level accepted by the investor.

$$\xi_{\mathrm{CVaR}}(\alpha) = \left(\int_{\alpha}^{1} \xi_{\mathrm{VaR}}(\beta) d\beta\right)/(1-\alpha) \tag{2}$$

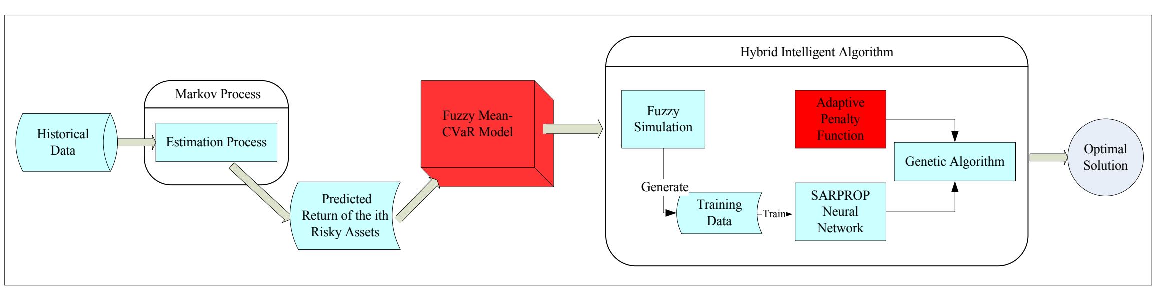


Fig. 1. The overall framework of hybrid intelligent algorithm for solving fuzzy Mean-CVaR model

In Eq. (2), $\xi_{\mathrm{VaR}}(\alpha)=\inf\{x|\mathrm{Cr}\{\xi\le x\}\ge\alpha\}$, and Cr is credibility measure.

The Markov process is used to replace the time series method and expert scoring method to predict the future returns of securities, which makes the model more robust. For details, please refer to [17].

III. Hybrid Intelligent Algorithm Based on SARPROP Neural Network and GA with Adaptive Penalty Function

The overall framework of hybrid intelligent algorithm for solving fuzzy Mean-CVaR model is shown in Fig. 1. Fuzzy simulation generates the training data and testing data for SARPROP neural network which is used for approximating the expected value and CVaR for fuzzy returns. GA with adaptive penalty function is used to search the optimal solution with the help of the trained SARPROP neural network.

A. Fuzzy Simulation

Since the returns of securities in the fuzzy Mean-CVaR model are fuzzy variables, it will be difficult to compute the expected values and CVaR for fuzzy returns by analytical methods. Fuzzy simulation [21-22] which is an application of Monte-Carlo methods, provides an effective approximation algorithm. The detailed techniques of fuzzy simulation can be found in the book [23]. It will be used to obtain the training data and testing data for the SARPROP neural network in our hybrid intelligent algorithm.

For each portfolio $\boldsymbol{x}=(x_1, x_2, ..., x_n)$, the algorithm to calculate expected values of portfolio return $\mathrm{E}\left[\sum_{i=1}^{n} x_i\,\xi_i\right]$ is shown as Algorithm 1, where $\boldsymbol{\xi}=(\xi_1,\xi_2,\ldots,\xi_n)$ is the fuzzy returns vector predicted by the Markov process.

Algorithm 1 the algorithm to calculate $\mathrm{E}\left[\sum_{i=1}^{n} x_i\,\xi_i\right]$

Input: portfolio $\boldsymbol{x}=(x_1, x_2, ..., x_n)$, a sufficiently large nonnegative real number M, and a sufficiently large positive integer N
Output: $\mathrm{E}\left[\sum_{i=1}^{n} x_i\,\xi_i\right]$
$e_1\leftarrow 0$
$e_2\leftarrow 0$
$h\leftarrow M/N$
for j = 0 to N
  $z_j\leftarrow j*h$
  $\mathrm{Cr}[j]\leftarrow$ calculate $\mathrm{Cr}\left\{\sum_{i=1}^{n} x_i\,\xi_i\ge z_j\right\}$
  $z_j'\leftarrow -M+j*h$
  $\mathrm{Cr}'[j]\leftarrow$ calculate $\mathrm{Cr}\left\{\sum_{i=1}^{n} x_i\,\xi_i\ge z_j'\right\}$
end for
for j = 0 to N−1
  $e_1\leftarrow e_1+h(\mathrm{Cr}[j]+\mathrm{Cr}[j+1])/2$
  $e_2\leftarrow e_2+h(\mathrm{Cr}'[j]+\mathrm{Cr}'[j+1])/2$
end for
return e_1-e_2

Algorithm 2 the algorithm to obtain $\xi_{\mathrm{CVaR}}(\alpha)$

Input: the given level of confidence α and a positive integer N_1
Output: $\xi_{\mathrm{CVaR}}(\alpha)$
$e\leftarrow 0$
for i = 0 to N_1
  $\alpha_i\leftarrow\alpha+i(1-\alpha)/N_1$
  find the minimum value of r_i satisfying $L(r_i)\le\alpha_i$
  $e\leftarrow e+r_i*(1-\alpha)/N_1$
end for
return $e/(1-\alpha)$

The minimum value of r is the approximate value of the VaR for fuzzy returns on the condition $\mathrm{Cr}\left\{\sum_{i=1}^{n} x_i\,\xi_i\le r\right\}\ge\alpha$. Then, based on the formula (3), the algorithm to obtain CVaR for fuzzy returns $\xi_{\mathrm{CVaR}}(\alpha)$ is shown as Algorithm 2. The meanings of variables and parameters in the formula (3) are same as above.

$$L(r)=\frac{1}{2}\left(\max_{1\le k\le N}\left\{\mu_k\,\middle|\,\sum_{i=1}^{n} x_i\,\xi_i\ge r\right\}+1\max_{1\le k\le N}\left\{\mu_k\,\middle|\,\sum_{i=1}^{n} x_i\,\xi_i< r\right\}\right)\quad(3)$$

B. SARPROP Neural Network

Neural network is a kind of adaptive system which is composed of many simple processing units called neurons. It is widely used in function approximation, pattern recognition, image processing and so on [24-25]. Back Propagation (BP) and its variations are widely used as methods for training neural networks. One such variation, Resilient Back Propagation (RPROP), has been proven to be one of the best in terms of speed of convergence. Based on the RPROP, SARPROP [26] can address the problem about converging to local minima which is faced by all gradient descent based methods through using Simulated Annealing (SA) and guarantee the convergence rate at the same time. In our algorithm, the neural network trained by SARPROP is used to approximate the expected values and CVaR of fuzzy returns, which will speed up the hybrid intelligent algorithm significantly.

C. GA with Adaptive Penalty Function

GA is a heuristic algorithm which searches the optimal solution by simulating natural evolutionary process, making it suitable for global optimization problem [23]. In recent years, it has been widely used in portfolio theory. The techniques of GA in detail can be found in papers [27-28].

Constraints handling is the key of GA in the process of solving the fuzzy Mean-CVaR model. The simplest method is to remove solutions that don't satisfy the constraints, but this will increase the time spent on infeasible solutions. So, to speed up the process of finding the optimal solution, we add constraint handling techniques to GA.

The penalty function method which is the most popular constraint handling technique, is a method that combine the objective function and constraints into a penalty function to remove the constraints in the model. It is easy to implement, and there is no harsh requirement for the problem [29]. In order to obtain strong generality, we choose the adaptive penalty function method which uses the information obtained in the search process as feedback to guide the adjustment of the penalty factor, to handle the constraints in fuzzy Mean-CVaR model. The adaptive penalty function is defined as follow:

$$\Phi(x)=\xi_{\mathrm{CVaR}}(\alpha)+\gamma(\mathrm{t})*(\max\left(0,\mathrm{r}-\mathrm{E}\left[\sum_{\mathrm{i}=1}^{\mathrm{n}} x_i\xi_i\right]\right)^2+|1-\sum_{\mathrm{i}=1}^{\mathrm{n}} x_i| \quad (4)$$

Where $\gamma(\mathrm{t})$ is the penalty factor in the t^{th} generation and is updated by Formula (5), $\lambda_1>\lambda_2$. $\xi_{\mathrm{CVaR}}(\alpha)$ and $\mathrm{E}\left[\sum_{\mathrm{i}=1}^{\mathrm{n}} x_i\xi_{\mathrm{i}}\right]$ are approximated by the trained SARPROP neural network.

$$\gamma(\mathrm{t}+1)=\begin{cases}\lambda_1\gamma(\mathrm{t}), & \text{case 1}\\ \lambda_2\gamma(\mathrm{t}), & \text{case 2}\\ \gamma(\mathrm{t}), & \text{the remaining cases}\end{cases} \quad (5)$$

In case 1, best individuals found in the past t generations are both feasible solutions, while case 2 represents that best individuals found in the past t generations are both infeasible solutions.

GA with adaptive penalty function is the last step of the hybrid intelligent algorithm. It uses the trained SARPROP neural network and adaptive penalty function to evaluate the individual fitness in the population, and applies roulette wheel method to generate the next population. Simultaneously, it changes the population with a certain probability (crossover and mutation). The optimal individual is selected as the optimal solution of the model at last.

D. The Hybrid Intelligent Algorithm

Algorithm 3 describes the hybrid intelligent algorithm based on SARPROP neural network and GA with adaptive penalty function.

Algorithm 3 hybrid intelligent algorithm

Input: historical data on returns of securities
Output: the optimal solution $\boldsymbol{x}^* = (x_1^*, x_2^*, \ldots, x_n^*)$ of the model
for i=1 to the given size
    $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \leftarrow$ random number sequence
    data[i][0: n−1] $\leftarrow \boldsymbol{x} = (x_1, x_2, \ldots, x_n)$
    data[i][n] ←calculate $\mathrm{E}\left[\sum_{\mathrm{i}=1}^{\mathrm{n}} x_i\xi_i\right]$ by Algorithm 1
    data[i][n+1] ←calculate $\xi_{\mathrm{CVaR}}(\alpha)$ by Algorithm 2
end for
nn ← initialize a SARPROP neural network
while Iterations ≤ MaxIter and mse > DesiredErr
    train_on_data (nn, data)
end while
saveWeights (weightsfile)
for i=1 to PopSize
    Individual[i] ← initialize individual in population
end for
for i=1 to PopSize
    Obj[i][0:1] ← nn.outPut(Individual[i])
end for
evaluation (0)
while Iters ≤ Gen
    selection ()
    crossover ()
    mutation ()
    for i=1 to PopSize
        Obj[i][0:1] ←nn. outPut (Individual[i])
    end for

26:     evaluation (Iters)
27: **end while**
28: $\boldsymbol{x}^* = (x_1^*, x_2^*, \ldots, x_n^*)$ ←optimal solution of the model

The algorithm procedure is: First, we generate the training set by Algorithm 1 and Algorithm 2 (lines 1-6). Then, the SARPROP uses the training set to train the neural network to get a SARPROP neural network to approximate the expected value and CVaR for fuzzy returns, and save the weight matrixes to weightsfile (lines 7-11). After that, we initialize individuals in the population for GA (lines 12-14). Next, we give the rank order of the individuals according to the adaptive penalty function values with the help of SARPROP neural network ***nn***, select the individuals by roulette wheel method, and update the population by crossover () and mutation () (lines 15-27). Finally, we take the optimal individual as the optimal solution for the model (line 28).

IV. Parallel Hybrid Intelligent Algorithm Based on SARPROP Neural Network and GA with Adaptive Penalty Function

In order to accelerate the hybrid intelligent algorithm for solving fuzzy Mean-CVaR model, the MPI technology was adopted to parallel hybrid intelligent algorithm.

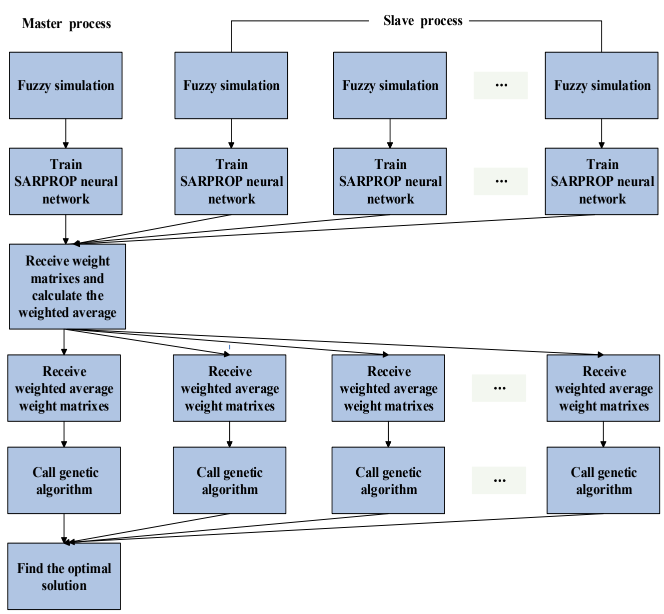


Fig. 2. The schematic diagram of parallel hybrid intelligent algorithm

By using MPI, the procedure of solving fuzzy Mean-CVaR model is divided into subtasks, which are assigned to different processes respectively. For load balancing, we can divide fuzzy simulation tasks evenly into different subtasks, and assign one process to generate the training data and testing data of one subtasks. Then, each process also need to complete the subtask of training the neural network and using GA with adaptive penalty function to search the optimal solution.

A schematic diagram of parallel hybrid intelligent algorithm using m processes is shown in Fig. 2. There are two kinds of processes: the master process and the slave process. In fuzzy simulation, each process performs Num/m times simulations to generate training set and testing set (Num represents the size of the training set and testing set). After the datasets are generated, each process uses SARPROP to train the neural network and sends weight matrixes to the master process when the training is finished. The master process calculates the weighted average of all weight matrixes to generate a new neural network, and then broadcast the weight matrixes to the slave processes. At last, the master process selects the best individual from individuals which are considered to have the highest fitness by each process, and takes it as the solution of portfolio selection.

V. Numrical Analysis

In order to illustrate the validity and efficiency of the hybrid intelligent algorithm, we achieve a quantitative comparison of our algorithm and the method which is obtained by integrating traditional GA and fuzzy simulation directly (Fuzzy-GA-only method) on the Era supercomputer. Also, we conduct experiments under different processor cores to demonstrate the scalability of the parallel hybrid intelligent algorithm.

A. Experimental Environment and Parameter Settings

TABLE I. System Parameters of “Era”

Parameter	**ConfRev**
Operating System	CentOS release 6.4(Final)
Linux Kernel	2.6.32-358.el6
Complier	Intel Compiler 2013_sp1.0.080
MPI Library	Intel MPI 4.1.3.049
Network	56Gbps FDR InfiniBand
Blade Compute Node	2 Intel Xeon E5-2680V2(10 cores|2.8Ghz) 64 GB DDR3 ECC 1866MHz memory

The experiments are carried out on a blade computing system of the Era supercomputer at Computer Network Information Center of Chinese Academy of Sciences, the detailed system parameters are shown in Table I.

For the fuzzy Mean-CVaR model, the minimum value of the return required by the investor r is set to -0.0186%, and the predetermined confidence level accepted by the investor α is 0.9.

For SSE 50 experimental dataset, the parameters in hybrid intelligent algorithm are set as follows: a) fuzzy simulation: 6000 cycles; b) SARPROP neural network: 2560 training data, 2560 testing data, 2 hidden layers, 30 hidden neurons, 2 output neurons, expected mean square error 10^{-10} and maximum number of iterations 1500; c) GA: population size 600, evolution times 2000, cross probability 0.6 and mutation probability 0.2.

Most of the parameters in hybrid intelligent algorithm for SSE 180 experimental dataset are the same as SSE 50 experimental dataset, except that the training data and the testing data are both 1024.

B. Experiment Results and Analysis

Based on the historical data of SSE 180 index constituent stocks and SSE 50 index constituent stocks from April 1, 2015 to April 6, 2017, we implement some numerical experiments on the Era supercomputer to compare our algorithm with Fuzzy-GA-only method from the aspects of computing time and portfolio performance. Also, we test the parallel hybrid intelligent algorithm on 32, 64, 128, 512, 1024 processor cores of Era supercomputer respectively, and calculate the speedup and parallel efficiency under different processor cores.

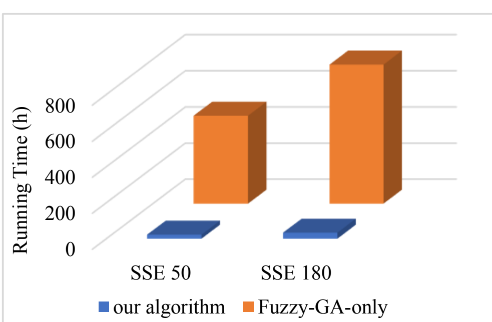


Fig. 3. Single-core computation time of two algorithms in different experimental datasets

TABLE II. INVESTMENT RETURN AND RISK (DAY ADJUSTMENT)

Dataset	Algorithm	Return (%)	CVaR
SSE 180	our algorithm	0.185	0.00517
	Fuzzy-GA-only	0.17728	0.0048763
SSE 50	our algorithm	0.358684	0.009533
	Fuzzy-GA-only	0.345097	0.009259

As shown in Table II, the investment returns of Fuzzy-GA-only method in SSE 180 dataset and SSE 50 dataset are 0.17728% and 0.345097% while our optimal portfolios' investment returns are 0.185% and 0.358684%. So, compared with Fuzzy-GA-only method, the investment returns of the optimal portfolios built by the hybrid intelligent algorithm are higher. Although the risk of the optimal portfolios built by our algorithm is slightly higher, but from the perspective of the risk on unit return, our algorithm can obtain similar results with Fuzzy-GA-only method, which demonstrates that the hybrid intelligent algorithm is effect.

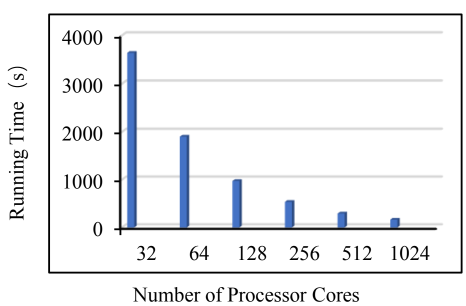
Fig. 3 shows the single-core calculation time of the two algorithms in different experimental datasets. Form Fig. 3, we can see that the single-core calculation time of our algorithm is significantly less than the time spent by Fuzzy-GA-only method. In detail, the calculation time of Fuzzy-GA-only method in SSE 50 dataset is 487.75h while it is 22.477h by using our algorithm, which is reduced by nearly 21.7 times. The calculation time of Fuzzy-GA-only method in SSE 180 dataset is 770.22h, however, with our algorithm, the time is 33.34h, which is reduced by about 23.1 times. Results in Fig. 3 indicate that, when obtaining a similar solution, our algorithm is much faster than Fuzzy-GA-only method, which show that the hybrid intelligent algorithm is better than Fuzzy-GA-only method.

The computing time of the parallel hybrid intelligent algorithm under different processor cores is shown in Fig. 4. In Fig. 4(a), we notice that, with 32 processor cores, the model calculating time is 3637.39s, while, it will cost 161.68s with 1024 processor cores, which is reduced by nearly 21.5 times. We also notice that in Fig. 4(b), with 32 processor cores, the time that we get the optimal portfolio is 2415.42s, however, with 1024 processor cores, the time is 114.54s, which is reduced by about 20 times. Results in Fig. 4 indicate that the parallel hybrid intelligence algorithm reduces the time it takes to solve the fuzzy Mean-CVaR model significantly.

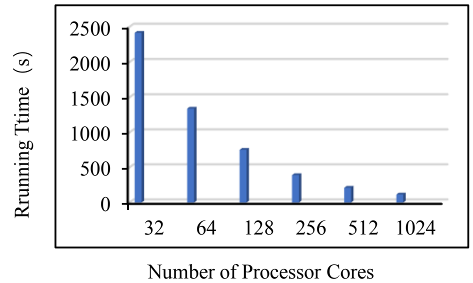
With same parameter settings, the speedup is defined as the ratio of the runtime obtained with 32 processor cores to the time computed with multi-core. The parallel efficiency is defined as the ratio of the current speedup to the number of processor cores used. Based on the runtime of 32 processor cores, we calculate the speedup and parallel efficiency under different processor cores. The results are summarized in Fig. 5-6.

It can be seen from Fig. 5 that the speedup under different processor cores increases approximately linearly. It is because of the quite limited amount of communication between processor cores. The slave process only needs to send 31N+M double type data to the master process in the entire parallel process, where N is the number of securities, M is a constant that is independent of N. That is, the communication of parallel hybrid intelligent algorithm is O(N), which makes the parallel program still have a good speedup when executed on more processor cores.

As shown in Fig. 6, when using 64 processor cores, the parallel efficiency of the two datasets is maximized. The maximum parallel efficiency of SSE 180 dataset is 95.79% while it is 90.15% in SSE 50 dataset. When the number of processor cores is larger, the parallel efficiency in the two datasets begins to decrease obviously since the overhead time, mostly related to communications among the processor cores. When using 1024 processor cores, we notice that the parallel efficiency of the two datasets reach the minimum. The minimum parallel efficiency in SSE 180 dataset and SSE 50 dataset are 70.3% and 65.9% respectively. It is obvious to us that the parallel efficiency in different datasets preponderates over 65%, which shows the parallel algorithm has strong extensibility under different processor cores.



(a) SSE 180 dataset



(b) SSE 50 dataset

Fig. 4. Parallel computation time in different processor cores

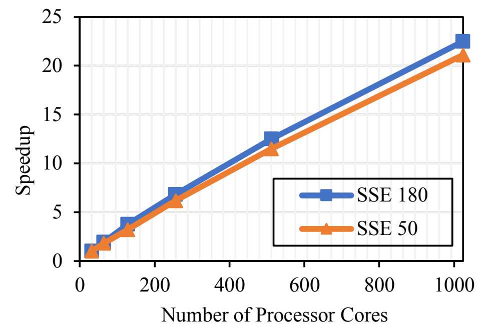


Fig. 5. The speedup for different processor cores

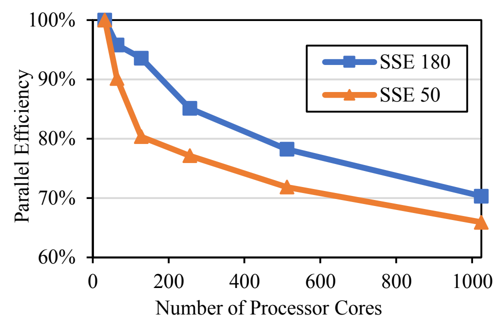


Fig. 6. The parallel efficiency for different processor cores

VI. CONCLUSION

In this paper, we presented a hybrid intelligent algorithm to solve the fuzzy Mean-CVaR portfolio model efficiently. In order to accelerate the computation of hybrid intelligent algorithm in solving fuzzy Mean-CVaR model further, we adopted MPI technology to parallel. The comparison experiments show that the hybrid intelligent algorithm is effect and it is better than Fuzzy-GA-only method. Experiments under different processor cores demonstrate that the parallel hybrid intelligent algorithm is able to scale well when even executed on thousands of processor cores.

ACKNOWLEDGMENT

This work was partially supported by National Key R&D Plan of China (No. 2017YFB0202202) and the State Key Program of National Natural Science Foundation of China (No. 91530324).

REFERENCES

[1] H. Markowitz, "Portfolio selection," Journal of Finance, vol. 7, pp. 77-91, 1952.

[2] B. Rom, K. Ferguson, "Post-modern portfolio theory comes of age," Journal of Investing, vol. 3, pp. 11-17, 1994.

[3] S. Li, "An insurance and investment portfolio model using chance constrained programming," Omega International Journal of Management Science, vol. 23, pp. 577-585, 1995.

[4] P.C. Fishbum, "Mean-risk analysis with risk associated with below-target returns," The American Economic Review, vol. 67(2), pp. 116-126, 1977.

[5] H. Konno, H. Shirakawa, H. Yamazaki, "A mean-absolute deviation-skewness portfolio optimization model," Annals of Operations Research, vol. 45, pp. 205–220, 1993.

[6] C.D. Feinstein, M.N. Thapa, "Notes: A reformation of a mean-absolute deviation portfolio optimization," Management Science, vol. 39, pp. 1552-1553, 1993.

[7] L. Yu, S. Wang, K. Lai, "Neural network-based mean-variance-skewness model for portfolio selection," Computers and Operations Research, vol. 35, pp. 34–46, 2008.

[8] P. Artzner, F. Delbaen, J.M. Eber, et al, "Coherent measure of risk," Mathematical Finance, vol. 9(3), pp. 203-228, 1999.

[9] R.T. Rockafellar, S. Uryasev, "Conditional value-at-risk for general loss distributions," Journal of Banking and Finance, vol. 26, pp. 1443-1471.

[10] L.A. Zadeh, "Fuzzy sets," Information and Control, vol. 8(3), pp. 338-353, 1965.

[11] M.R. Simonelli, "Indeterminacy in portfolio selection," European Journal of Operational Research, vol. 163(1), pp. 170-176, 2005.

[12] M. Arenas, A. Bilbao, M. Rodriguez, "A fuzzy goal programming approach to portfolio selection," European Journal of Operational Research, vol. 133(2), pp. 287-297, 2001.

[13] H. Tanaka, P. Guo, I. Türksen, "Portfolio selection based on fuzzy probabilities and possibility distributions," Fuzzy Sets and Systems, vol. 111(3), pp. 387-397, 2000.

[14] X. Huang, "Portfolio selection with fuzzy returns," Journal of Intelligent and Fuzzy Systems, vol. 18(4), pp. 383-390, 2007.

[15] B. Liu, "Theory and practice of uncertain programming," Springer Berlin Heidelberg, 2009.

[16] G.H. Chen, X.L. Liao, X. Yu, "Multi-objective portfolio selection model based on intuitionistic fuzzy programming," Fuzzy Systems and Mathematics, vol. 2, pp. 129-135, 2012.

[17] M. Li, Y.H. Hu, "Optimization of mean-CVaR portfolio optimization based on credibility theory-the modeling and empirical," Statistics and Information Forum, vol. 31(12), pp. 23-29, 2016.

[18] Z. Qin, X. Li, X. Ji, "Portfolio selection based on fuzzy cross-entropy," Journal of Computational and Applied Mathematics, vol. 228 (1), pp. 139-149, 2009.

[19] X. Huang, "Mean-entropy models for fuzzy portfolio selection," IEEE Transactions on Fuzzy Systems, vol. 16, pp. 1096–1101, 2008.

[20] X. Huang, "Mean-semivariance models for fuzzy portfolio selection," Journal of Computational and Applied Mathematics, vol. 217, pp.1-8, 2008.

[21] B. Liu, K. Iwamura, "Chance constrained programming with fuzzy parameters," Fuzzy Sets and Systems, vol. 94(2), pp. 227-237, 1998.

[22] B. Liu, Y.K. Liu, "Expected value of fuzzy variable and fuzzy expected value models," IEEE Transactions on Fuzzy Systems, vol. 10(4), pp. 445-450, 2002.

[23] B.C. Liu, R.Q. Zhao, G. Wang, "Uncertain planning and application," Tsinghua University Press, 2003.

[24] K. Hornik, M. Stinchcombe, H. White, "Multilayer feedforward networks are universal approximators," Neural Networks, vol. 2(5), pp. 359-366, 1989.

[25] G. Cybenko, "Approximation by superposition of sigmoidale function," Mathematics of Control Signals and Systems, vol. 2(2), pp. 304-314, 1989.

[26] N.K. Treadgold, T.D. Gedeon, "The sarprop algorithm: a simulated annealing enhancement to resilient back propagation," 1996.

[27] J.H. Holland, "Adaptation in natural and artificial systems," MIT Press, 1992.

[28] D.E. Goldberg, "Genetic algorithms in search, optimization and machine learning," MA: Addison-Wesley, vol. xiii(7), pp. 2104-2116, 1989.

[29] L. Wang, H. He, Y.H. Jin, "Study on intelligent constraint processing technology," Chemical Industry Automation and Instrumentation, vol. 35(1), pp. 1-7, 2008.

[30] P.H. Yao, Y.H Hu, Z.H Lu, et al, "Improved parallel randomized quasi monte carlo algorithm of asian option pricing on MIC architecture," International Symposium on Distributed Computing and Applications to Business, Engineering and Science. IEEE, pp. 157-161, 2015.