Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Reputation-based joint optimization of user satisfaction and resource utilization in a computing force network*#

Yuexia FU[1], Jing WANG[1], Lu LU[‡1], Qinqin TANG[2], Sheng ZHANG[3]

[1]*China Mobile Research Institute, Beijing 100053, China*

[2]*Purple Mountain Laboratories, Nanjing 211111, China*

[3]*China Mobile Communications Corporation, Beijing 100033, China*

E-mail: fuyuexia@chinamobile.com; wangjingjc@chinamobile.com; lulu@chinamobile.com; qqtang@bupt.edu.cn; zhangsheng@chinamobile.com

Received Mar. 5, 2023; Revision accepted Aug. 9, 2023; Crosschecked Apr. 2, 2024

Abstract: Under the development of computing and network convergence, considering the computing and network resources of multiple providers as a whole in a computing force network (CFN) has gradually become a new trend. However, since each computing and network resource provider (CNRP) considers only its own interest and competes with other CNRPs, introducing multiple CNRPs will result in a lack of trust and difficulty in unified scheduling. In addition, concurrent users have different requirements, so there is an urgent need to study how to optimally match users and CNRPs on a many-to-many basis, to improve user satisfaction and ensure the utilization of limited resources. In this paper, we adopt a reputation model based on the beta distribution function to measure the credibility of CNRPs and propose a performance-based reputation update model. Then, we formalize the problem into a constrained multi-objective optimization problem and find feasible solutions using a modified fast and elitist non-dominated sorting genetic algorithm (NSGA-II). We conduct extensive simulations to evaluate the proposed algorithm. Simulation results demonstrate that the proposed model and the problem formulation are valid, and the NSGA-II is effective and can find the Pareto set of CFN, which increases user satisfaction and resource utilization. Moreover, a set of solutions provided by the Pareto set give us more choices of the many-to-many matching of users and CNRPs according to the actual situation.

Key words: Computing force network; Resource scheduling; Performance-based reputation; User satisfaction
https://doi.org/10.1631/FITEE.2300156 **CLC number:** TP393

1 Introduction

1.1 Motivation

With the development of the 5th generation mobile communication technology, edge computing, and artificial intelligence, the computing force network (CFN) has become a hot topic in the information technology and communication technology industries (Abbas et al., 2017; Mao et al., 2017; Liu B et al., 2021; Zhang et al., 2022; Di et al., 2023); it is a new type of network, and has the capability to schedule resource requests to the optimal endpoints based on the awareness, control, and management of computing and network resources

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (No. 2022ZD0115303), the 2023 Beijing Outstanding Young Engineers Innovation Studio, China, and the Beijing University of Posts and Telecommunications–China Mobile Research Institute Joint Innovation Foundation (No. CMYJY-202200536)

Electronic supplementary materials: The online version of this article (https://doi.org/10.1631/FITEE.2300156) contains supplementary materials, which are available to authorized users

ORCID: Yuexia FU, https://orcid.org/0009-0001-6096-2039; Jing WANG, https://orcid.org/0009-0006-5747-2115; Lu LU, https://orcid.org/0009-0000-5740-9489; Qinqin TANG, https://orcid.org/0000-0002-3930-7005

to guarantee the quality of service (QoS) for resource users while ensuring the resource utilization of resource providers. CFN incorporates computing force, network, data, intelligence, and other elements to achieve integrated optimization and service delivery through the convergence, awareness, control, and management of multiple elements. Recently, it has gradually become a new trend to consider the computing and network resources of multiple providers as a whole (Fortes, 2010; Monteiro et al., 2014; Chen et al., 2020; Bao et al., 2021; Stoica and Shenker, 2021); it differs from traditional optimization to achieve individual optimality by each provider separately, and can achieve overall optimization in a global perspective through unified scheduling and trading at the global level.

In existing scheduling, the scheduling center and resource providers are often in a trusted environment and there is no malicious competition of misrepresented information. However, for CFN, there are multiple resource providers, and since there are competitive relationships among multiple resource providers and each resource provider considers its own interest, the situation of misrepresenting performance information to compete for resource requests may occur. Thus, in CFN, the evaluation of reliability of resource providers is essential. On the other hand, CFN has different roles, including resource users, resource providers, and scheduling centers (Chen et al., 2020; Bao et al., 2021). Different roles have different goals; resource providers want to maximize their resource utilization and utility, and users want to meet their resource requirements. It is essential to consider how to balance the different requirements of these two roles.

Facing multiple users and multiple computing and network resource providers (CNRPs), we adopt a reputation model based on the beta distribution function to measure the credibility of CNRPs and update the reputation according to the actual performance. Then, we try to find trade-off between user satisfaction and resource utilization to satisfy user requirements and CNRPs' interest. We formalize the problem into a constrained multi-objective optimization problem and find the Pareto set using a modified fast and elitist non-dominated sorting genetic algorithm (NSGA-II). A large number of solutions in the Pareto set provide more choices of the matching decision for users and CNRPs according to the actual situation in CFN, which is the largest difference between our work and previous studies.

1.2 Contributions

The main contributions in this paper are summarized as follows:

1. We propose a reputation-based joint optimization framework to achieve a trade-off between user satisfaction and resource utilization in CFN. First, we introduce a reputation model and a weighted and performance-based reputation update model to evaluate the credibility level of CNRPs; second, we develop a performance- and reputation-based comprehensive evaluation model of CNRP towards different users.

2. We formulate the many-to-many matching decision process between users and CNRPs as a constrained multi-objective optimization problem, to maximize the average user satisfaction and average resource utilization simultaneously.

3. We use NSGA-II to solve the constrained multi-objective optimization problem and obtain the Pareto optimal solutions of CFN, which can meet the actual needs of different users.

4. Numerical simulations validate the proposed model, the problem formulation, and the NSGA-II.

2 Related works

2.1 Optimization for computing and network convergence

The computing and network convergence related architectures include the architecture of the computing and network convergence towards sixth-generation (6G) requirements (Tang et al., 2021), the basic architecture and the working process of comput first networking (Tian et al., 2021), and the architecture and key technologies of computing-aware networking (Yao et al., 2021; Du et al., 2022). As for optimization and scheduling, several approaches and methodologies have been proposed, focusing on modeling and solving the resource scheduling optimization problems in the multi-cloud model through multi-dimensional resource constraints. Kan et al. (2018) adopted a priority-based scheduling method, in which devices are assigned a priority to promote QoS. Benblidia et al. (2019) proposed a fuzzy theory based scheduling method, which focuses on solving the resource uncertainty and dynamics at the edge. Kang et al. (2022) proposed a scheduling framework based deep reinforcement learning to reduce the average response time of tasks and improve the central processing unit (CPU) utilization of resources. Dong et al. (2022) proposed a service optimization scheduling algorithm of CFN based on bulk *k*-dimensional tree (Bkd-tree). As for the multi-objective optimization among multiple resource providers, Yuan et al. (2021) proposed a simulated-annealing-based biobjective differential evolution algorithm to obtain an approximate Pareto optimal set to maximize the profit and minimize the task loss for data centers powered by renewable energy and the smart grid, by jointly determining the split of tasks among multiple Internet service providers (ISPs). Li et al. (2021) proposed and integrated a new double rank-based task sequencing method with a multi-objective heuristic algorithm for multiple workflow scheduling, to find trade-off schedules to execute multiple workflows on cloud computing resources to balance multiple objectives. Liu L et al. (2018) proposed a heterogeneous earliest finish time (HEFT) using technique for order preference through similarity to an ideal solution method. For the unconstrained case, Liu L et al. (2018) presented a three-stage strategy to select the optimal solutions by applying a non-dominated sorting approach. Chaitra et al. (2020) proposed a new dynamic resource provisioning technique in a multi-cloud environment, which uses the lion optimization algorithm, wherein characteristics of nomad and pride lion groups were taken into account. This achieved better results while optimizing multiple objectives such as completion time, average response time, makespan, cost, and average resource utilization.

At present, a few studies have paid attention to multi-objective optimization of resource utilization and user satisfaction among multiple CNRPs from different entities. There are also a few studies on multi-objective optimization in CFN.

2.2 Reputation systems

Reputation systems have been widely studied in several domains such as e-commerce (Resnick et al., 2000; Resnick and Zeckhauser, 2002), the Internet (Buchegger and Le Boudec, 2002), ad-hoc wireless networks (Buchegger and Le Boudec, 2003a, 2003b), and peer-to-peer networks (Xiong and Liu, 2004). Previous works have used reputation as the metric to rate the reliability or trustworthiness of an entity in certain activities according to its past behaviors. Recently, in federated multi-node joint learning, reputation metric was used to assess the reliability of a federated learning worker candidate, thus ensuring reliable worker selection (Song et al., 2022). Additionally, in wireless sensor networks, reputation was introduced to represent the past behavior of each node, and the metric was used as an inherent aspect in predicting nodes' future behaviors (Ganeriwal et al., 2008; Fang et al., 2016). In the Internet of Vehicles, a reputation mechanism was introduced to measure the reliability of vehicles participating in federated learning (Zou et al., 2021). Note that beta distribution is widely used in reputation modeling. Liu XL and Jia (2019) presented an iterative reputation ranking algorithm in terms of beta probability distribution. Fang et al. (2016) proposed a beta-based trust and reputation evaluation system for wireless sensor networks' node trust and reputation evaluation. Song et al. (2022) proposed a reputation model based on the beta distribution function to measure the credibility of local users and a reputation-based scheduling policy with a user fairness constraint.

As for CFNs, there are few related works on reputation-based scheduling. When different CNRPs participate, each CNRP considers only its own interest, and the situation of misrepresenting performance information to compete for resource requests may occur. So, the reliability of CNRPs needs to be evaluated to avoid assigning resource requests to unreliable CNRPs.

3 Reputation-based joint optimization framework of user satisfaction and resource utilization in a computing force network

In this section, we take an overview of the system. We describe the reputation model and the reputation update mechanism, and introduce the performance- and reputation-based comprehensive evaluation model. We also formulate the scheduling decision process into a constrained multi-objective problem.

3.1 System overview

Fig. 1 shows the overview of a many-to-many CFN system. The users act as computing and network resource consumers, requesting resources and proposing requirements. CNRPs provide computing and network resources with various performances in an all-in-one manner. Here, CNRPs may belong to different entities and compete with each other to provide the required resources to users with limited resources, which is a multi-party involvement and mutual distrust environment.

The CFN scheduling center bridges users and CNRPs. When users put forward resource demands, CNRPs report the current resource information to the CFN scheduling center, and then the CFN scheduling center matches the multiple CNRPs and multiple users to meet the users' needs under the resource constraints. In this paper, we assume that one user request will be assigned to one CNRP.

As shown in Fig. 2, we assume that there are N users {user 1, user 2, $\cdots$, user N} and M CNRPs {CNRP 1, CNRP 2, $\cdots$, CNRP M}. User i is modeled as a three-tuple [ra_i, rt_i, $\mathrm{preference}_i$], where ra_i is the required resource amount, rt_i is the required response time, and $\mathrm{preference}_i$ represents the weights of ra_i and rt_i, implying that users may have different preferences for the response time and the required resource amount. In CFN, we could consider different preferences of users to adjust the selection strategy and ensure various users' experience. CNRP j is modeled as a three-tuple [Ra_j, Rt_j, R_j], where Ra_j is the provided resource amount, and Rt_j is the response time, which means the end-to-end time, including the two-way network transmission time from the user to the selected CNRP j and the estimated processing time of the selected CNRP j; the estimated processing time could be obtained based on the task size and the available resource amount. R_j is the reputation, which evaluates the performance reliability of CNRP j.

In the framework, the basis of effective and reliable many-to-many scheduling is to model the reputation of CNRPs accurately, based on the historical performance, not scheduling users' requests to the unreliable CNRPs. First, the reputation model, weighted and performance-based reputation update model, and comprehensive evaluation model of CNRPs are modeled in Sections 3.2, 3.3, and 3.4, respectively. Second, to resolve the many-to-many matching decision process as a multi-objective optimization problem, two objective functions are included: average user satisfaction and average resource utilization.

3.2 Reputation model of CNRP

In this work, we propose a reputation-based computing and network convergence framework and consider

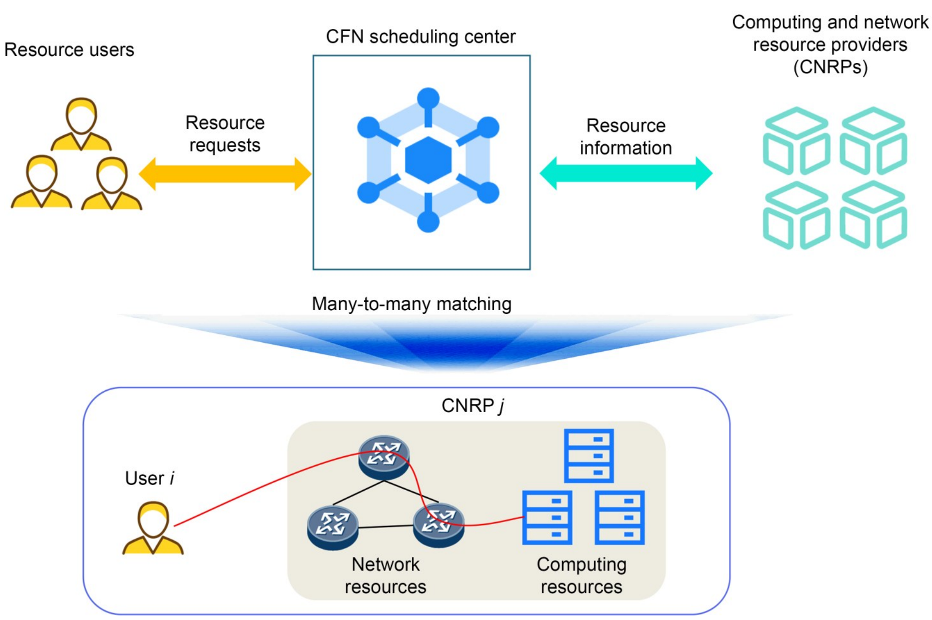

Fig. 1 Overview of a many-to-many computing force network (CFN) system

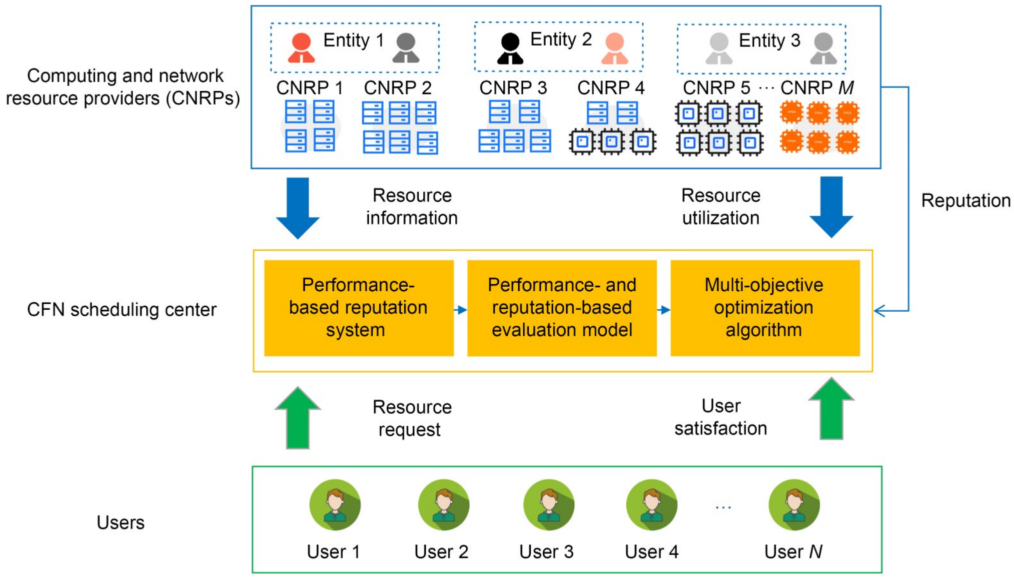

Fig. 2 Framework of reputation-based joint optimization of user satisfaction and resource utilization in a computing force network (CFN)

reputations z as representing the probability that a given CNRP will provide resources as promised based on user i's requirements. This could resist the effects of the unreliable and negative performance of CNRPs. Therefore, the reputations z are contained in the unit interval [0, 1], and the value of z closer to 1 suggests a greater performance.

To represent the reputation of CNRP j, we consider using the Bayesian formulation (Ganeriwal et al., 2008) to compute the probability of a belief given an observation. Generally, the Bayes theorem is given by (the solution process is provided in Section 1.1 in the supplementary materials)

$$P\left(\frac{\text{Belief}}{\text{Observation}}\right)=\frac{P\left(\frac{\text{Observation}}{\text{Belief}}\right)P(\text{Belief})}{\sum P\left(\frac{\text{Observation}}{\text{Belief}}\right)P(\text{Belief})}. \tag{1}$$

In the framework of CFN, we use belief and observation to represent the reputation and the current performance of a CNRP, respectively. The reputation of CNRP j (Rep_j) is calculated and updated by the CFN scheduling center based on the current performance of CNRP j towards user i's requirements (Per_{ij}):

$$\text{Rep}_j=\frac{P\left(\frac{\text{Per}_{ij}}{\text{Rep}_j}\right)\text{Rep}_j}{\sum P\left(\frac{\text{Per}_{ij}}{\text{Rep}_j}\right)\text{Rep}_j}. \tag{2}$$

To evaluate the next behavior of CNRP j, as for the prior distribution, we assign $P(z)$, which reflects the performance uncertainty about the behavior of CNRP j. Then, based on the historical information, the CFN scheduling center predicts the next behavior of CNRP j.

To facilitate the expression and update of the reputation according to whether the performance of CNRP j reaches the promised performance, the CFN scheduling center characterizes CNRP j's behavior on a binary scale; i.e., a positive behavior means that the actual performance at least reaches the promised performance, and a negative behavior means that the actual performance does not reach the promised performance. Furthermore, the classical beta binomial framework could be used to estimate reputations to evaluate the binary ratings with a two-parameter class of distributions (Gelman et al., 1995; Josang and Ismail, 2002). So, we use beta distribution (Liu XL and Jia, 2019) to represent the reputation of CNRP j, where the beta distribution can be expressed by the gamma function:

$$P(z)=\frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}z^{a-1}(1-z)^{b-1}, \tag{3}$$

where $0\le z\le 1$, $a>0$, and $b>0$.

At the initialization stage of the reputation system, when there is no prior information, CNRP j's prior reputation can be expressed as a uniform distribution over [0, 1], i.e., $P(z)$=uni(0, 1)=Beta(1, 1).

Suppose that the CFN scheduling center has assigned a user's requests to CNRP j (p+q) times, and that the CFN scheduling center characterizes the performance of CNRP j as p positive behaviors and q negative behaviors. With the binary rating model (Fang et al., 2016), the posterior distribution of CNRP j's reputation z can be derived as (the solution process is provided in Section 1.2 in the supplementary materials)

$$P(z)=\text{Beta}(p+1,q+1). \tag{4}$$

It can be seen from Eq. (4) that the posterior distribution of CNRP j's reputation z is still a beta distribution. Based on Eq. (4), the reputation of CNRP j, maintained at the CFN scheduling center, is derived as

$$\text{Rep}_j=\text{Beta}\left(a_j+1,b_j+1\right), \tag{5}$$

where a_j and b_j represent the numbers of the positive and negative behaviors of CNRP j, respectively. The CFN scheduling center maintains a reputation table (a_j, b_j) to represent the reputation of CNRP j.

Based on the above analysis, Rep_j is not a physical quantity but a probability distribution function, so we define a statistical expectation as the reputation value R_j, which evaluates the overall performance reliability of CNRP j, and is derived as (Fang et al., 2016)

$$R_j=E\left[\text{Beta}\left(a_j+1,b_j+1\right)\right]=\frac{a_j+1}{a_j+b_j+2}. \tag{6}$$

3.3 Weighted and performance-based reputation update model

Each time a service is provided, the parameter is updated. Suppose that CNRP j has (c+d) resource deliveries and that they are characterized as c positive behaviors and d negative behaviors. Therefore, the updated reputation of CNRP j can be expressed as (Liu XL and Jia, 2019)

$$\begin{aligned}R_j&=\frac{\text{Bin}(c+d,c)\text{Beta}\left(a_j+1,b_j+1\right)}{\text{Normalization}}\\&=\text{Beta}\left(a_j+c+1,b_j+d+1\right).\end{aligned} \tag{7}$$

So, when we update reputation R_j, it is equivalent to updating the two parameters a_j and b_j as

$$\begin{cases}a_{j\,\text{new}}=a_j+c,\\ b_{j\,\text{new}}=b_j+d.\end{cases} \tag{8}$$

The fact is that there are many factors that may affect the reputation update (Zou et al., 2021). Rather than consider positive and negative behaviors only and characterize CNRP j's behavior on a binary scale, to more accurately evaluate the reputation update and motivate the CNRP to provide services as promised, we consider mainly the following three factors:

1. Freshness weight. The reputation of CNRP changes with time; usually the recent behaviors with more freshness have more significant weights than past events. To reflect the time effect on reputation, a freshness weight wage(l) is defined to illustrate the freshness of behaviors:

$$\text{wage}(l)=\mathrm{e}^{-1/l}, \tag{9}$$

where l is the number of times that CNRP is selected to provide resources for users.

2. Weight effects. The positive behavior increases the reputation of CNRP, and vice versa. To discourage negative behaviors, the negative behaviors should have higher weights on reputation update calculation than the positive behaviors. Specifically, we denote the weights of positive and negative behaviors as ρ and σ, respectively: $\rho\le\sigma$ and $\rho+\sigma=1$.

3. Accuracy contribution. The performance of the service provided by this CNRP affects the reputation update, and when considering the accuracy contribution, we use a continuous distribution from 0 to 1 to more accurately evaluate the reputation update, which is derived from the difference between the actual performance of the l^{th} behavior and the promised performance.

Assume that P_{jl}^* is the actual performance of the l^{th} resource delivery of CNRP j, and that P_{jl} denotes the promised performance.

(1) If $P_{jl}^*>P_{jl}$, it means that the actual performance is better than the promised one, so it is characterized as a positive behavior and has a positive contribution

denoted as $P\_\mathrm{Ac}P_{jl}^*$, and the greater the degree of variation, the greater the value of $P\_\mathrm{Ac}P_{jl}^*$.

(2) If $P_{jl}^* < P_{jl}$, this means that the actual performance is worse than the promised one, so it is characterized as a negative behavior and has a negative contribution denoted as $N\_\mathrm{Ac}P_{jl}^*$, and the greater the degree of variation, the smaller the value of $N\_\mathrm{Ac}P_{jl}^*$.

(3) When $P_{jl}^* = P_{jl}$, this behavior has no effect on the reputation update. The update is divided into positive and negative contributions:

$$P\_\mathrm{Ac}P_{jl}^* = \log_2\left(1 + \frac{P_{jl}^* - P_{jl}}{P_{jl}}\right), \tag{10}$$

$$N\_\mathrm{Ac}P_{jl}^* = \log_2\left(1 + \frac{P_{jl} - P_{jl}^*}{P_{jl}}\right), \tag{11}$$

where $P\_\mathrm{Ac}P_{jl}^*, N\_\mathrm{Ac}P_{jl}^* \in (0, 1)$.

Therefore, we can update the weighted reputation according to CNRP j's contributions $P\_\mathrm{Ac}P_{jl}^*$ and $N\_\mathrm{Ac}P_{jl}^*$ as follows:

$$\begin{cases} a_{j\,\mathrm{new}} = \mathrm{wage}(l)\cdot a_j + \rho\cdot P\_\mathrm{Ac}P_{jl}^*, \\ b_{j\,\mathrm{new}} = \mathrm{wage}(l)\cdot b_j + \sigma\cdot N\_\mathrm{Ac}P_{jl}^*, \end{cases} \tag{12}$$

where wage(l) is the aging weight of the positive and negative reputations, which generally takes values in the range of (0, 1) (Buchegger and Le Boudec, 2003b; Ganeriwal et al., 2008).

We will elaborate on the usage of the reputation metric in the context of CNRP evaluation in Section 3.4.

3.4 Performance- and reputation-based comprehensive evaluation model of CNRPs

To measure the general level of CNRPs, we construct an overall evaluation model based on performance and reputation. In this paper, we consider two aspects to construct the overall evaluation model.

3.4.1 General performance of CNRPs

Resource performance is a comprehensive evaluation of CNRP's computing resource performance and network resource performance. As defined in Section 3.1, user i's requirement is [ra_i, rt_i, $\mathrm{preference}_i$], where ra_i and rt_i are the requirements related to network and computing resources respectively, and $\mathrm{preference}_i$ is defined as the weights of ra_i and rt_i in the range of [φ_i, $1-\varphi_i$]. Under the condition in which each user will be assigned to one CNRP, the general performance score S_{ij} of CNRP j towards user i is derived as

$$S_{ij} = \varphi_i \mathrm{Sa}_{ij} + (1 - \varphi_i)\mathrm{St}_{ij}, \tag{13}$$

$$\mathrm{Sa}_{ij} = \mathrm{Ra}_j - \mathrm{ra}_i, \tag{14}$$

$$\mathrm{St}_{ij} = \mathrm{Rt}_j - \mathrm{rt}_i, \tag{15}$$

where Sa_{ij} represents CNRP j's resource amount performance score towards user i, St_{ij} represents the response time score towards user i, and φ_i is used to represent the weight of Sa_{ij}.

Furthermore, the normalized S_{jl}^* (Dong et al., 2022) is derived as

$$S_{jl}^* = \frac{S_{ij} - S_{ij\,\min}}{S_{ij\,\max} - S_{ij\,\min}}, \tag{16}$$

where $S_{ij\,\max}$ and $S_{ij\,\min}$ are the maximum and minimun values of the general performance score of CNRP j towards user i, respectively.

For the general performance aspect, for user i, the candidate list of CNRP j, which could meet user i's performance requirements, could be generated based on the following constraint:

$$\mathrm{Sa}_{ij} \geq 0,\ \mathrm{St}_{ij} \leq 0. \tag{17}$$

3.4.2 Reputation of CNRPs

In Sections 3.2 and 3.3, the reputation model and reputation update model have been proposed, respectively. Reputation is an important factor in the overall evaluation model that can measure the cumulative performance of CNRPs in the process of providing resources in the past. To conduct a more scientific comprehensive evaluation, reputation is normalized as

$$R_j^* = \frac{R_j - R_{j\,\min}}{R_{j\,\max} - R_{j\,\min}}, \tag{18}$$

where R_j^* is the normalized reputation of CNRP j, $R_j^* \in$ [0, 1]. $R_{j\,\max}$ is the maximum reputation and $R_{j\,\min}$ is the minimum reputation.

Assume that for user i, the number of CNRPs in the candidate list is nc_i. Then, the reputation threshold ψ_i for user i is not smaller than the average reputation score in the candidate list of CNRPs:

$$\psi_i = \frac{\sum R_j^*}{\mathrm{nc}_i}. \quad (19)$$

The reputation score SR_{ij} of CNRP j towards user i is derived as

$$\mathrm{SR}_{ij} = R_j^* - \psi_i. \quad (20)$$

Furthermore, the normalized SR_{ij}^* is derived as

$$\mathrm{SR}_{ij}^* = \frac{\mathrm{SR}_{ij} - \mathrm{SR}_{ij\,\min}}{\mathrm{SR}_{ij\,\max} - \mathrm{SR}_{ij\,\min}}, \quad (21)$$

where $\mathrm{SR}_{ij}^* \in [0, 1]$, and $\mathrm{SR}_{ij\,\max}$ and $\mathrm{SR}_{ij\,\min}$ are the maximum and minimum values of the reputation score of CNRP j towards user i, respectively.

For the reputation aspect, for user i, the candidate list of CNRP j with a better reputation could be generated based on the following constraint:

$$\mathrm{SR}_{ij}^* \geq 0. \quad (22)$$

Finally, the overall evaluation O_{ij} of CNRP j in the candidate list can be expressed as

$$O_{ij} = \alpha_1 \cdot S_{ij}^* + \alpha_2 \cdot \mathrm{SR}_{ij}^*, \quad (23)$$

where $O_{ij} \in [0, 1]$. α_1 and α_2 are the weights of CNRP j's general performance and the reputation of CNRPs respectively, and $\alpha_1+\alpha_2=1$. α_1 and α_2 are used to reconcile the impact of reputation on the overall evaluation of CNRP; the larger the α_2, the more significant the impact of the historical behavior on the prediction of CNRP performance.

3.5 Problem formulation

As specified in Section 3.1, there are N users {user 1, user 2, $\cdots$, user N} and M CNRPs {CNRP 1, CNRP 2, $\cdots$, CNRP M}, user i is defined as [ra_i, rt_i, $\mathrm{preference}_i$], and CNRP j is modeled as [Ra_j, Rt_j, R_j]. So, the overall resource amount γ of the system is

$$\gamma = \sum_{j=1}^{M} \mathrm{Ra}_j. \quad (24)$$

The average resource utilization function AU is derived as

$$\mathrm{AU} = \frac{1}{\gamma} \sum_{i=1}^{N} D_{ij} \cdot \mathrm{ra}_i \quad (25)$$

$$\text{s.t.} \quad D_{ij} \in \{0, 1\}, \quad (26)$$

where D_{ij} is a binary selection indicator, $D_{ij} = 1$ represents that CNRP j is selected by the CFN scheduling center to provide service for user i, and $D_{ij} = 0$, otherwise.

The average user satisfaction function AS is derived as

$$\mathrm{AS}_i = D_{ij} \cdot O_{ij}, \quad (27)$$

$$\mathrm{AS} = \sum_{i=1}^{N} \mathrm{AS}_i, \quad (28)$$

where AS_i is user i's satisfaction towards CNRP j. AS is the average user satisfaction of all users who use the resources.

Finally, we formulate the many-to-many matching decision process of users and CNRPs as a multi-objective optimization problem, which includes two objective functions—maximizing average user satisfaction AS in Eq. (28) and average resource utilization AU in Eq. (25). The optimal problem P1 can be formulated as follows:

$$\max \{\mathrm{AS}, \mathrm{AU}\} \quad (29)$$

$$\text{s.t.} \quad \begin{aligned} &\text{C1: } \sum_{i=1}^{N} D_{ij} \cdot \mathrm{ra}_i \leq \mathrm{Ra}_j, \text{ for } j = 1, 2, \cdots, M, \\ &\text{C2: } \sum_{j=1}^{M} D_{ij} \leq 1, \text{ for } i = 1, 2, \cdots, N. \end{aligned} \quad (30)$$

C1 means that for CNRP j, the resource amount provided to users should not exceed the resource amount Ra_j.

C2 means that user i's request is assigned to, at most, one CNRP.

4 Problem solution based on NSGA-II

Multi-objective optimization problems are common in several fields, and it is impossible for each objective function to achieve the best result at the same time. In general, there are two methods to solve multi-objective optimization problems. The first method is to transform the problem into a single-objective problem by giving each objective a different weight (Mostafa et al., 2013). However, when users' demands change, the weight values need to be reset, and the algorithms need to be rerun. Each objective restricts one another through decision variables, sometimes even contradictory goals, which makes the topological structure of the weighted objective function complicated.

The second method is to use a multi-objective evolutionary algorithm (Jara, 2014; Peng et al., 2017) to obtain a set of optimal solutions, which is called the Pareto set. Then, the user can make further decisions based on the actual needs (which is more important to users, user satisfaction or resource utilization). In this way, the system does not need to rerun the algorithms even if the users' preferences change. In fact, NSGA-II is widely used to solve online problems (Miriyala et al., 2018; Niu et al., 2018; Rehani and Garg, 2018). NSGA-II, proposed by Deb et al. (2002), has become one of the most popular multi-objective evolution algorithms, and we learn from these studies. NSGA-II is shown in the supplementary materials.

We now provide details of NSGA-II and use it to address problem P1:

1. Parameter initialization of NSGA-II. The main parameters of NSGA-II, i.e., the population size PS, the maximum number of iterations G, the probability of crossover CR, and the probability of mutation PM, are initialized.

2. Generating the initial population. In this work, the initial population consists of PS chromosomes. Each chromosome represents a set of matching and scheduling decisions. We design it so that each gene corresponds to one user. That is to say, the number of genes of each chromosome in the population is equal to the number of users. Suppose that user i's request is assigned to one CNRP. Then the values for its associated genes satisfy $V \in \{0, 1, \cdots, M\}$. So, the initial population is $[I_1, I_2, \cdots, I_N]$.

Moreover, we generate the population in three steps to maintain population diversity. First, we generate a chromosome in which all genes are 0. This means that not all users' requests are met. Second, we produce a chromosome with the same value j, which means that all users' requests are allocated to the same CNRP j. Finally, the remaining PS−2 chromosomes are developed randomly. The initial population is shown in Fig. S1 in the supplementary materials.

3. Evaluating the solutions. After initializing the population, we evaluate each chromosome by Eqs. (25) and (28). Note that there may be some chromosomes that do not satisfy the constraints (17), (22), (26), and (30); we call these chromosomes infeasible solutions.

Here, we define a variable called constraint violation (CV). The value of CV is 0 for feasible solutions, and the value of CV for infeasible solutions can be calculated as follows:

$$\mathrm{CV1}_i = \begin{cases} 0, & \text{if } \mathrm{Sa}_{ij} \geq 0, \mathrm{St}_{ij} \geq 0, \mathrm{SR}_{ij}^* \geq 0, \\ O_{ij}, & \text{otherwise}, \end{cases} \tag{31}$$

$$\mathrm{CV2}_j = \begin{cases} 0, \ \text{if} \sum_{i=1}^{N} D_{ij} \cdot \mathrm{ra}_i \leq \mathrm{Ra}_j, \\ \dfrac{\sum_{i=1}^{N} D_{ij} \cdot \mathrm{ra}_i - \mathrm{Ra}_j}{\mathrm{Ra}_j}, \ \text{otherwise}, \end{cases} \tag{32}$$

$$\mathrm{CV} = \sum_{i=1}^{N} \mathrm{CV1}_i + \sum_{j=1}^{M} \mathrm{CV2}_j, \tag{33}$$

where $\mathrm{CV1}_i$, $\mathrm{CV2}_j$, and CV denote the degree of constraint violation of user i on the average user satisfaction, the degree of constraint violation of CNRP j on the average resource utilization, and the degree of constraint violation of each chromosome on both average user satisfaction and average resource utilization, respectively. The value of CV can reflect the distance of the infeasible solutions from the feasible domain.

4. Fast non-dominated sorting approach. In this step, we stratify that the population and all chromosomes will be planned into different fronts. Individuals of the same Pareto rank are on the same level. The non-dominated solution x means that no other solutions dominate this solution in the population. Since problem P1 is a constraint problem, we first redefine the concept of Pareto dominance (see Section 2.1 in the supplementary materials) (Cui et al., 2019).

5. Crowding distance calculation. After stratifying the population, to find the solutions with a lower degree of similarity, the crowding distance (CD) is calculated to evaluate the density of solutions surrounding the specific solution (the solution process is provided in Section 2.2 in the supplementary materials). The computation of CD can be understood through Fig. S2 in the supplementary materials, using a crowding strategy to calculate the density of other individuals around a particular individual on the same non-dominated front. For solution x_i, two closest solutions

on the same frontier are considered as vertices to form a rectangle.

6. Tournament selection. For any two solutions x_1 and x_2, x_1 is better than x_2 if any of the following conditions is established. The selection operation (Cui et al., 2019) is given as follows:

(1) The domination rank of x_1 is smaller than that of x_2;

(2) x_1 and x_2 belong to the same front, but $CD_{x_1} > CD_{x_2}$.

According to these two principles, all solutions are given a final rank. We select individuals from the population to form the parent population. The better the sorting, the higher the probability of being selected as the next generation chromosome.

7. Crossover and mutation operators. In this step, we randomly choose two parents x_1 and x_2 in the parent population and exchange their chromosomes with each other in a certain way to form two new offspring, x_1^* and x_2^*. The crossover operation is shown in Section 2.3 in the supplementary materials.

After the crossover operation, a new individual is formed with a certain probability of genetic mutation. We randomly select a chromosome for mutation, and the probability of mutation determines whether mutation is performed or not. We use polynomial variation in this step.

8. Generating new population. In this step, new population will be generated. The offspring population is generated after the crossover and mutation operations in step 7. We use the same method as in step 3 to estimate their objective function values and the values of constraint violations.

Then, the whole population, including parent and offspring population, is sorted according to step 4, and CD of each chromosome is calculated by step 5. Finally, the whole population is sorted by step 6. According to the elite selection strategy (Deb et al., 2002), we select only the best PS individuals from the entire population to create a new parent population.

Finally, the algorithm judges whether the termination condition is met. If it is true, the first front is outputted as the optimal solution. Otherwise, the algorithm returns to step 4, and the process loops until the termination condition is satisfied.

The key procedure of NSGA-II is shown in Fig. 3.

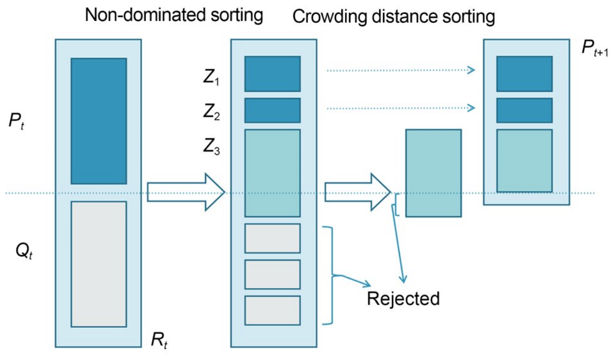

Fig. 3 NSGA-II procedure

P_t: a parent population after t iterations; Q_t: an offspring population after t iterations; R_t: a combined population from P_t+Q_t after t iterations; Z_i: one of the best non-dominated sets; P_{t+1}: a new parent population

5 Performance evaluation

To verify the performance of the proposed reputation-based NSGA-II in solving multi-objective optimization problems in a many-to-many environment, we analyze the influences of different parameters, such as the number of users and the resource amount, to confirm the performance of NSGA-II in solving problem P1. In addition, the effect of reputation is verified. We have simulated two cases, one where the number of users and the resource amount are comparable, and the other where the number of users is large and the resource amount is low.

5.1 Simulation setup

As shown in Fig. 2, the CFN scheduling center is centrally distributed, and our assumptions are that the number of users is 25 and that the number of CNRPs is 10. The users' requests are in a unified distribution over [1, 5] and the resource amount of CNRPs is in a unified distribution over [4, 10]. Moreover, the four parameters used in NSGA-II are listed in Table 1. To begin with, the reputations of CNRPs are in a uniform distribution over [0, 1].

Table 1 Parameters of NGSA-II

Parameter	Value
Population size PS	100
Maximum number of iterations G	100
Probability of crossover CR	0.9
Probability of mutation PM	0.1

5.2 Pareto fronts

As shown in Fig. 4, we can observe that the proposed NSGA-II is able to obtain a balance between average user satisfaction and resource utilization. First, distinct rank values denote different Pareto solution sets, where solutions within each set are mutually non-dominated. Second, solutions in higher-ranked sets are invariably dominated by at least one solution from a lower-ranked set. Conversely, solutions in lower-ranked sets remain non-dominated by any solution in higher-ranked sets. That is to say, the proposed NSGA-II can find more optimal decision-making options for the system, and provide more choices for the many-to-many matching of users and CNRPs, according to the actual situation.

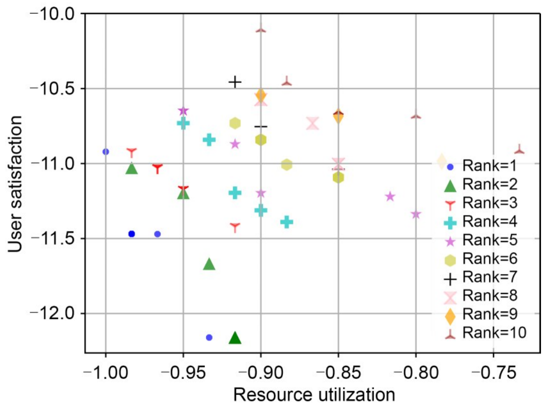

Fig. 4 Pareto fronts of user satisfaction and resource utilization

5.3 Impact of the number of users

Fig. 5 shows the average user satisfaction and average resource utilization when the number of users varies. We use the no-reputation scheme, in which only the resource utilization ratio is optimized and the effect of reputation is ignored, as a baseline. As for Fig. 5a, the number of users varies from 1 to 40, and the number of CNRPs is 10. Obviously, the average resource utilization increases with the increase of the number of users. The reason is that the amount of required resources increases, so more resources are used.

On the other hand, the impact of the number of users on the average user satisfaction can be seen with the folding line in Fig. 5a; when the number of users increases from 1 to 25, the average satisfaction decreases significantly. The reason is that there is a small

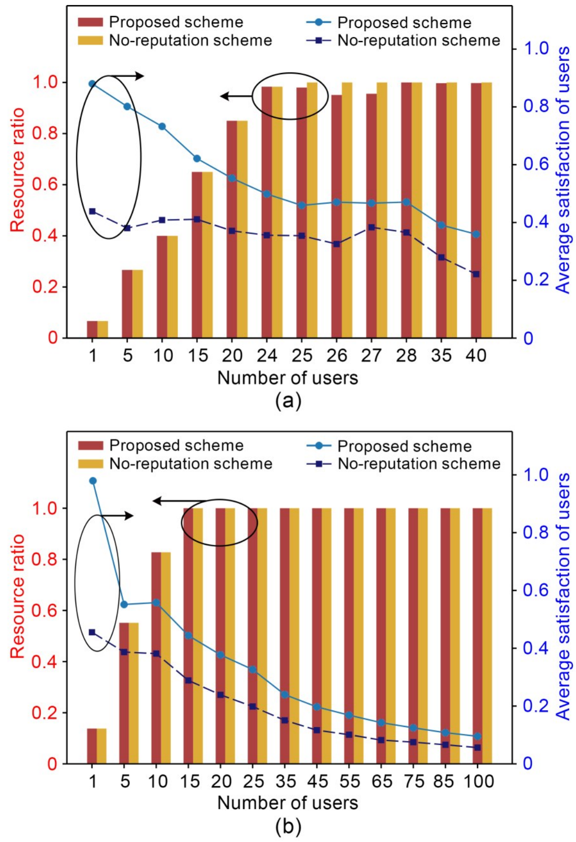

Fig. 5 User satisfaction and resource utilization with respect to different numbers of users: (a) the number of users varies from 1 to 40 when the number of CNRPs is 10; (b) the number of users varies from 1 to 100 when the number of CNRPs is 5

number of users in this range, so the proposed algorithm could select the CNRP with the best performance while guaranteeing the resource utilization. However, with the increase of the number of users, the proposed algorithm begins to select other CNRPs that are not the best but could meet users' needs, to improve the average resource utilization. When there are more than 28 users, there are no available resources for some users, and the user satisfaction is 0, causing the average user satisfaction to decrease.

As for Fig. 5b, the number of users varies from 1 to 100, the number of CNRPs is 5. Concerning the average resource utilization, when the number of users increases from 1 to 15, the average resource utilization increases to 100%. The reason is that the amount of required resources increases, so more resources are used. When the number of users is more than 15, the average resource utilization is stable at 100%. The reason for this is that when the resource utilization has reached almost 100%, the average resource utilization remains the same even though the number of users increases.

Concerning user satisfaction, when the number of users increases from 1 to 15, the average satisfaction decreases significantly. The reason is that there are a small number of users in this range, so the CNRP with the best performance could be selected to ensure user satisfaction while guaranteeing the resource utilization. However, with the increase of the number of users, the average resource utilization has reached 100% and more users' requests cannot be met, causing the user satisfaction to decrease.

Furthermore, we compare the two different conditions, with or without reputation, in two objective functions. Obviously, when we consider the reputation of CNRPs, the average user satisfaction is always higher and the resource utilization is not affected by the reputation.

5.4 Impact of resource amount

Fig. 6 shows the average user satisfaction and average resource utilization when the number of CNRPs varies. As for Fig. 6a, the number of CNRPs varies from 3 to 35, and the number of users is 25. The average user satisfaction increases with the increase of the number of CNRPs as shown in the folding line. When there is a small number of CNRPs, few users could be provided resources, so the user satisfaction is low. When the number of CNRPs increases, there are more available resources provided for more users, causing the average user satisfaction to increase.

The impact of the amount of resources on the average resource utilization is shown in Fig. 6a. When there is a small number of CNRPs, the number of CNRPs is smaller than the users' needs and all of the resources are used, so from 3 to 9 the resource utilization is about 100%; from 11 to 35, since the number of CNRPs increases and is larger than users' needs, the resource utilization decreases gradually.

In Fig. 6b, the number of CNRPs varies from 1 to 10, and the number of users is 100. The average user satisfaction increases slowly with the increase of the number of CNRPs as shown in the folding line. The reason is that, because the number of CNRPs is small, few users could be provided resources, so the user satisfaction is low. When the number of CNRPs increases,

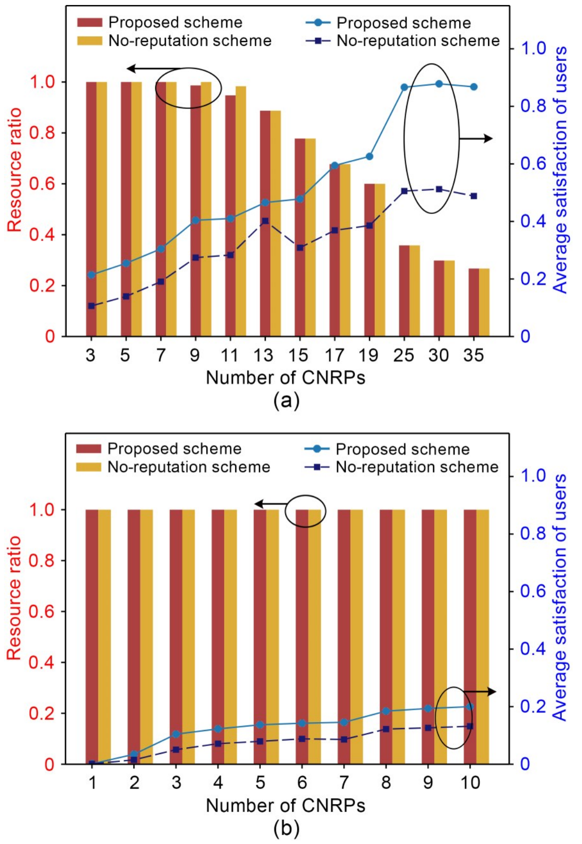

Fig. 6 User satisfaction and resource utilization with respect to different numbers of CNRPs: (a) the number of CNRPs varies from 3 to 35 when the number of users is 25; (b) the number of CNRPs varies from 1 to 10 when the number of users is 100

more available resources are provided for more users, causing the average user satisfaction to increase.

The average resource utilization is 100% all the time, since the number of CNRPs is small and cannot meet 100 users' requirements.

Furthermore, we compare the two different conditions, with or without reputation. Obviously, when we consider the reputation of CNRPs, the average user satisfaction is always higher and the resource utilization is not affected by the reputation.

5.5 Impact of reputation of CNRPs

Here, we set 10 time intervals, and users arrive at different time intervals in turn. Then, we set two conditions: the first is that each CNRP has a 5% probability of downward mutation (performance degradation) in performance parameters at each interval, and the mutated parameter follows a semi-normal distribution with a mean of 1 and a variance of 5; the second is that each CNRP has a 5% probability of upward mutation (performance improvement) in performance parameters at each interval, and the mutated parameter follows a semi-normal distribution with a mean of 1 and a variance of 5. With the reputation scheme, after satisfying user's request in the current interval, the reputations of the CNRPs will be updated, considering their actual performances. For the next interval, the target CNRPs may change.

Fig. 7 shows the average user satisfaction when the performance of CNRPs varies from −0.32 to −1.47 and from 0.5 to 3.0, when the number of users is 10 and the number of CNRPs is 40.

The impact on the average user satisfaction when the CNRP performs worse than promised one can be seen in Fig. 7a. Obviously, the average user satisfaction decreases with the decrease of performance. What is more, when considering reputation, there is a slight decrease with the decrease of performance, but it is relatively stable. In contrast, when reputation is not considered, which means that the performance degradation of CNRPs during past intervals cannot be considered, there is a significant decline with the decrease of CNRPs' performance. The results mean that in such cases where CNRPs' actual performance is worse than the promised one, the reputation scheme could react to the performance change quickly and remove the potential CNRP with bad performance, finally helping find new CNRPs with a good reputation to guarantee the users' satisfaction.

Similar results can be observed in Fig. 7b, where the impact on the average user satisfaction can be observed when the CNRP performance is better than the promised one. Obviously, the average user satisfaction increases with the increase of CNRP's performance. In addition, when considering reputation, the average user satisfaction has a significant increase with the increase of CNRP's performance. In contrast, when reputation is not considered, which means that the performance improvement of CNRPs during past intervals cannot be considered, the average user satisfaction gradually increases with the increase of CNRPs' performance. The results mean that, in such cases where CNRPs' real performance is better than the promised one, the reputation scheme react to the performance change quickly and add the potential CNRP

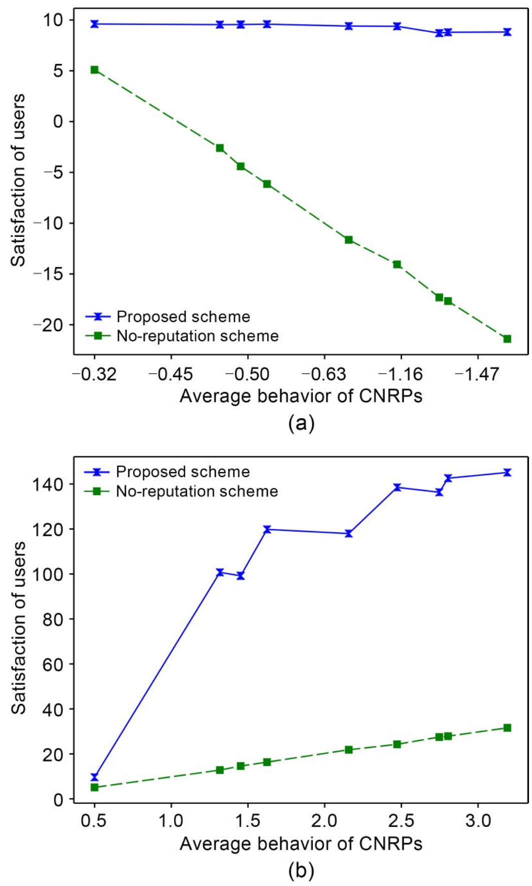

Fig. 7 Impacts of the reputation of CNRPs when the number of users is 10 and the number of CNRPs is 40: (a) user satisfaction when the performance of CNRP decreases; (b) user satisfaction when the performance of CNRP increases

with better performance, finally helping find new CNRPs with a good reputation to improve the users' satisfaction.

Fig. 8 shows the average user satisfaction when the performance of CNRPs varies from −0.18 to −0.97 and from 0.50 to 2.25, when the number of users is 100 and the number of CNRPs is 5.

Similarly, as shown in Fig. 8a, as the average performance of CNRP decreases, the proposed scheme is still able to find the optimal matching scheme quickly compared to the no-reputation scheme, significantly increasing the average user satisfaction of all served users (only some of the 100 users will be served because the number of CNRPs is not large enough to serve all users). As shown in Fig. 8b, as the average performance of CNRP increases, the proposed scheme is still able to find the optimal matching scheme quickly compared to the no-reputation scheme, so that the

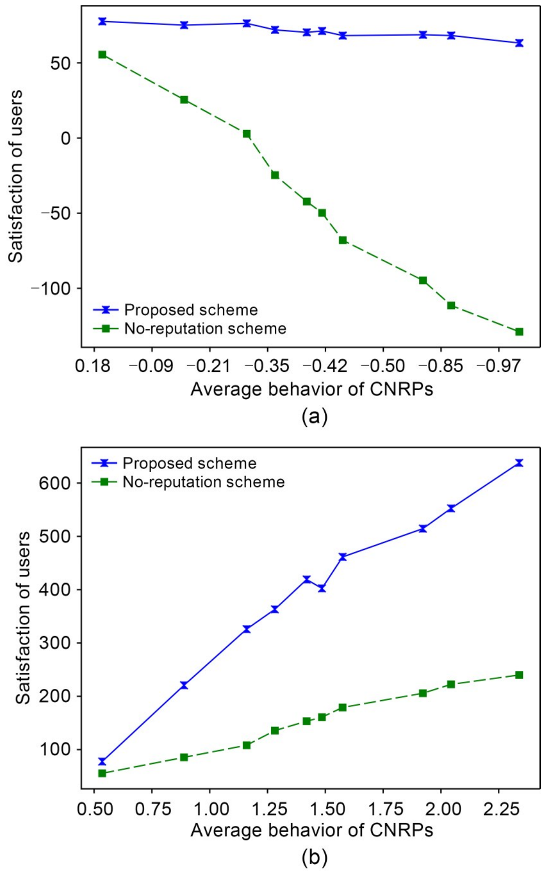

Fig. 8 Impacts of the reputation of CNRPs when the number of users is 100 and the number of CNRPs is 5: (a) user satisfaction with respect to the performance of CNRP worsening; (b) user satisfaction with respect to the performance of CNRP increasing

average user satisfaction of all the served users can be relatively stable.

6 Conclusions

Driven by emerging applications that place higher demands on both computing and networks, CFN has become a hot research subject. In this paper, we presented a reputation-based joint optimization framework to balance user satisfaction and resource utilization in CFN. First, we introduced the reputation system and proposed a weighted and performance-based reputation update model to evaluate the reliability degree of CNRPs, and developed a performance- and reputation-based comprehensive evaluation model of CNRP towards different numbers of users. Second, we formulated the many-to-many matching decision process between users and CNRPs as a constrained multi-objective optimization problem, and leveraged NSGA-II to accommodate user satisfaction and resource utilization jointly. Last, the simulation results showed that the proposed model, problem formulation, and the NSGA-II are valid to obtain the Pareto set to jointly optimize user satisfaction and resource utilization, considering the reputation metric. In the future, we could consider the fairness problem in the many-to-many matching process, and choose the CNRP with fewer cumulative service times while satisfying the user's needs. We could consider the incentive mechanism in the CFN, based on the service performance of the CNRPs, to decide whether to issue incentives to motivate more CNRPs to participate and promote the development of CFN.

Contributors

Yuexia FU and Lu LU designed the research. Yuexia FU and Qinqin TANG designed and conducted the simulations. Yuexia FU and Lu LU drafted the paper. Jing WANG helped organize the paper. Yuexia FU, Lu LU, and Sheng ZHANG revised and finalized the paper.

Conflict of interest

All the authors declare that they have no conflict of interest.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

Abbas N, Zhang Y, Taherkordi A, et al., 2017. Mobile edge computing: a survey. *IEEE Int Things J*, 5(1):450-465. https://doi.org/10.1109/JIOT.2017.2750180

Bao QZ, Ren XX, Liu CF, et al., 2021. Resource trading with hierarchical game for computing-power network market. Proc 5th Int Joint Conf, p.94-109. https://doi.org/10.1007/978-3-030-85896-4_8

Benblidia MA, Brik B, Merghem-Boulahia L, et al., 2019. Ranking fog nodes for tasks scheduling in fog-cloud environments: a fuzzy logic approach. Proc 15th Int Wireless Communications & Mobile Computing Conf, p.1451-1457. https://doi.org/10.1109/IWCMC.2019.8766437

Buchegger S, Le Boudec JY, 2002. Performance analysis of the CONFIDANT protocol. Proc 3rd ACM Int Symp on Mobile Ad Hoc Networking & Computing, p.226-236. https://doi.org/10.1145/513800.513828

Buchegger S, Le Boudec JY, 2003a. Coping with False Accusations in Misbehavior Reputation Systems for Mobile Ad-Hoc Networks. EPFL Technical Report IC/2003/31, Elsevier, Lausanne, Switzerland.

Buchegger S, Le Boudec JY, 2003b. The effect of rumor spreading in reputation systems for mobile ad-hoc networks. Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, Article 10.

Chaitra T, Agrawal S, Jijo J, et al., 2020. Multi-objective optimization for dynamic resource provisioning in a multi-cloud environment using lion optimization algorithm. Proc 20th Int Symp on Computational Intelligence and Informatics, p.000083-000090. https://doi.org/10.1109/CINTI51262.2020.9305822

Chen YF, Li ZY, Yang B, et al., 2020. A Stackelberg game approach to multiple resources allocation and pricing in mobile edge computing. *Fut Gener Comput Syst*, 108: 273-287. https://doi.org/10.1016/j.future.2020.02.045

Cui LZ, Xu C, Yang S, et al., 2019. Joint optimization of energy consumption and latency in mobile edge computing for Internet of Things. *IEEE Int Things J*, 6(3):4791-4803. https://doi.org/10.1109/JIOT.2018.2869226

Deb K, Pratap A, Agarwal S, et al., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput*, 6(2):182-197. https://doi.org/10.1109/4235.996017

Di Z, Luo T, Qiu C, et al., 2023. In-network pooling: contribution-aware allocation optimization for computing power network in B5G/6G era. *IEEE Trans Netw Sci Eng*, 10(3): 1190-1202. https://doi.org/10.1109/TNSE.2022.3225292

Dong YQ, Guan CC, Chen YL, et al., 2022. Optimization of service scheduling in computing force network. Int Conf on Service Science, p.147-153. https://doi.org/10.1109/ICSS55994.2022.00031

Du ZP, Li ZQ, Duan XD, et al., 2022. Service information informing in computing aware networking. Int Conf on Service Science, p.125-130. https://doi.org/10.1109/ICSS55994.2022.00027

Fang WD, Zhang CL, Shi ZD, et al., 2016. BTRES: beta-based trust and reputation evaluation system for wireless sensor networks. *J Netw Comput Appl*, 59:88-94. https://doi.org/10.1016/j.jnca.2015.06.013

Fortes JAB, 2010. Sky computing: when multiple clouds become one. Proc 10th IEEE/ACM Int Conf on Cluster, Cloud and Grid Computing, Article 4. https://doi.org/10.1109/CCGRID.2010.136

Ganeriwal S, Balzano LK, Srivastava MB, 2008. Reputation-based framework for high integrity sensor networks. *ACM Trans Sens Netw*, 4(3):1-37. https://doi.org/10.1145/1362542.1362546

Gelman A, Carlin JB, Stern HS, et al., 1995. Bayesian Data Analysis. Chapman and Hall/CRC, New York, USA. https://doi.org/10.1201/9780429258411

Jara EC, 2014. Multi-objective optimization by using evolutionary algorithms: the *p*-optimality criteria. *IEEE Trans Evol Comput*, 18(2):167-179. https://doi.org/10.1109/TEVC.2013.2243455

Josang A, Ismail R, 2002. The beta reputation system. Proc 15th Bled Electronic Commerce Conf, p.502-2511.

Kan TY, Chiang Y, Wei HY, 2018. Task offloading and resource allocation in mobile-edge computing system. Proc 27th Wireless and Optical Communication Conf, p.1-4. https://doi.org/10.1109/WOCC.2018.8372737

Kang KX, Ding D, Xie HM, et al., 2022. Adaptive DRL-based task scheduling for energy-efficient cloud computing. *IEEE Trans Netw Serv Manag*, 19(4):4948-4961. https://doi.org/10.1109/TNSM.2021.3137926

Li F, Seok MG, Cai WT, 2021. A new double rank-based multi-workflow scheduling with multi-objective optimization in cloud environments. IEEE Int Parallel and Distributed Processing Symp Workshops, p.36-45. https://doi.org/10.1109/IPDPSW52791.2021.00015

Liu B, Mao JW, Xu L, et al., 2021. CFN-dyncast: load balancing the edges via the network. IEEE Wireless Communications and Networking Conf Workshops, p.1-6. https://doi.org/10.1109/WCNCW49093.2021.9420028

Liu L, Fan Q, Buyya R, 2018. A deadline-constrained multi-objective task scheduling algorithm in mobile cloud environments. *IEEE Access*, 6:52982-52996. https://doi.org/10.1109/ACCESS.2018.2870915

Liu XL, Jia SW, 2019. An iterative reputation ranking method via the beta probability distribution. *IEEE Access*, 7:540-547. https://doi.org/10.1109/ACCESS.2018.2885551

Mao YY, You CS, Zhang J, et al., 2017. A survey on mobile edge computing: the communication perspective. *IEEE Commun Surv Tut*, 19(4):2322-2358. https://doi.org/10.1109/COMST.2017.2745201

Miriyala SS, Subramanian VR, Mitra K, 2018. TRANSFORM-ANN for online optimization of complex industrial processes: casting process as case study. *Eur J Oper Res*, 264(1):294-309. https://doi.org/10.1016/j.ejor.2017.05.026

Monteiro A, Teixeira C, Pinto JS, 2014. Sky computing: exploring the aggregated cloud resources—part II. Proc 9th Iberian Conf on Information Systems and Technologies, p.1-6. https://doi.org/10.1109/CISTI.2014.6876862

Mostafa HA, El-Shatshat R, Salama MMA, 2013. Multi-objective optimization for the operation of an electric distribution system with a large number of single phase solar generators. *IEEE Trans Smart Grid*, 4(2):1038-1047. https://doi.org/10.1109/TSG.2013.2239669

Niu XX, Wang HC, Hu S, et al., 2018. Multi-objective online optimization of a marine diesel engine using NSGA-II coupled with enhancing trained support vector machine. *Appl Therm Eng*, 137:218-227. https://doi.org/10.1016/j.applthermaleng.2018.03.080

Peng CD, Liu HL, Gu FQ, 2017. An evolutionary algorithm with directed weights for constrained multi-objective optimization. *Appl Soft Comput*, 60:613-622. https://doi.org/10.1016/j.asoc.2017.06.053

Rehani N, Garg R, 2018. Meta-heuristic based reliable and green workflow scheduling in cloud computing. *Int J Syst Assur Eng Manag*, 9(4):811-820. https://doi.org/10.1007/s13198-017-0659-8

Resnick P, Zeckhauser R, 2002. Trust among strangers in Internet transactions: empirical analysis of eBay's reputation system. In: Baye M (Ed.), The Economics of the Internet and E-Commerce. Emerald Group Publishing Limited, Bingley, England, p.127-157.

Resnick P, Kuwabara K, Zeckhauser R, et al., 2000. Reputation systems. *Commun ACM*, 43(12):45-48. https://doi.org/10.1145/355112.355122

Song ZD, Sun HG, Yang HH, et al., 2022. Reputation-based federated learning for secure wireless networks. *IEEE Int Things J*, 9(2):1212-1226. https://doi.org/10.1109/JIOT.2021.3079104

Stoica I, Shenker S, 2021. From cloud computing to sky computing. Proc Workshop on Hot Topics in Operating Systems, p.26-32. https://doi.org/10.1145/3458336.3465301

Tang XY, Cao C, Wang YX, et al., 2021. Computing power network: the architecture of convergence of computing and networking towards 6G requirement. *China Commun*, 18(2):175-185. https://doi.org/10.23919/JCC.2021.02.011

Tian L, Yang MZ, Wang SG, 2021. An overview of compute first networking. *Int J Web Grid Serv*, 17(2):81-97. https://doi.org/10.1504/IJWGS.2021.114566

Xiong L, Liu L, 2004. PeerTrust: supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Trans Knowl Data Eng*, 16(7):843-857. https://doi.org/10.1109/TKDE.2004.1318566

Yao HJ, Lu L, Duan XD, 2021. Architecture and key technologies for computing-aware networking. *ZTE Technol J*, 27(3):7-11 (in Chinese). https://doi.org/10.12142/ZTETJ.202103003

Yuan HT, Bi J, Zhou MC, et al., 2021. Biobjective task scheduling for distributed green data centers. *IEEE Trans Autom Sci Eng*, 18(2):731-742. https://doi.org/10.1109/tase.2019.2958979

Zhang P, Peng MG, Cui SG, et al., 2022. Theory and techniques for "intellicise" wireless networks. *Front Inform Technol Electron Eng*, 23(1):1-4. https://doi.org/10.1631/FITEE.2210000

Zou Y, Shen F, Yan F, et al., 2021. Reputation-based regional federated learning for knowledge trading in blockchain-enhanced IoV. IEEE Wireless Communications and Networking Conf, p.1-6. https://doi.org/10.1109/WCNC49053.2021.9417347

List of supplementary materials

1 Reputation-based joint optimization of user satisfaction and resource utilization

2 Problem solution based on NSGA-II

Fig. S1 Initial population

Fig. S2 Schematic of the Pareto rank of the solution space

Algorithm S1 NSGA-II