

QoS-Oriented Mobile Service Composition Over Opportunistic Networks

Qinglan Peng, *Member, IEEE*, John Doe, *Fellow, OSA*, and Jane Doe, *Life Fellow, IEEE*

Abstract—The abstract goes here.

Index Terms—Computer Society, IEEE, IEEEtran, journal, LATEX, paper, template.

1 INTRODUCTION

RECENT YEARS, with the rapid developments of mobile devices and wireless communication technologies, web services are no longer limited to traditional platforms and they are becoming more flexible and pervasive [1]. The hardware of mobile devices will continue make breakthroughs to extend mobile devices capabilities in terms of memory, computational power, storage capacity, and so on [1]. Mobile technologies huge potential brings great opportunities to traditional service computing in the mobile environment [2]. As a result, the global interest of mobile applications is on the rise. Both researchers and industrial companies are inspired to pave the road for mobile Web service provisioning [3], [4] [1].

While mobile device have powerful computing and communication capabilities, the high data rate services, however, drain out the energy of the device much faster than before *[Call for Papers]**.

To achieve the goal of reducing mobile device energy consumption, in this paper we advocate a QoS-oriented mobile service composition over opportunistic approach, where a mobile user in opportunistic can combine and exploit each others resources to boost their computing power and overcome the limitations of their own resources without the communication energy footprint and the extreme centralization of mobile cloud computing (As shown in fig.1) [5]. Its main rationality is three-fold. First, opportunistic user encounters are prevalent and sufficient in daily life [6], which offers plenty of opportunities to exploit nearby mobile worker for task solving [T], [T], [T]. Second, many mobile tasks require huge computational resources or data trasfer (e.g., Tensorflow on mobile, Photoshop on mobile, Online video), for an energy consumption and cost perspective, nearby mobile service provider are more adept at executing them than the online workers, because this paradigm can reduce data transfer over cellular network which consume more energy than device to device (D2D) communications such as Bluetooth, NFC, WiFi-direct and LTE-D2D [] []. Third, D2D communications are promising

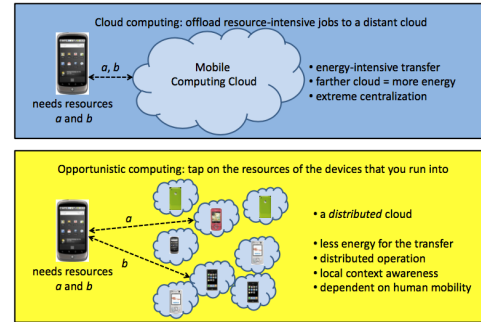


Fig. 1. SensorDistance.

to replenish traditional cellular communications in terms of user throughput increase, cellular traffic reduction and network coverage extension, in this way, users can get better quality of service and save communication fee at the same time [7]. In a word, this framework shares the similar spirit with the emerging paradigm cyber foraging over opportunistic networks, such that mobile users opportunistically exploit nearby device resources to facilitate their computational task processing [8], [9], [10] [11].

To address the aforementioned challenges and concerns, we propose a new approach for service selections and compositions in mobile opportunistic network. The main contributions are:

1) We propose an framework for mobile service provision [mobile service opportunistic network MSON] to address the problem of service selection for mobile service composition in the mobile encounter enviroment where both service requesters and providers are nonstationary. In such environment, mobile user can share mobile services with nearby mobile devices through D2D links.

2) For MSON, we propose a mobile service QoS model for service provision which consider mobile service availability as an important QoS attribute to capture user's mobility behavior.

3) Based on MSON and the proposed mobile service QoS model, we transfer the mobile service composition over opportunistic problem to an optimization problem and propose to utilize the Krill-Herd algorithm (KH) to solve it. We conduct aseries of evaluations to validate that our al-

- M. Shell was with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332. E-mail: see <http://www.michaelshell.org/contact.html>
- J. Doe and J. Doe are with Anonymous University.

Manuscript received April 19, 2005; revised August 26, 2015.

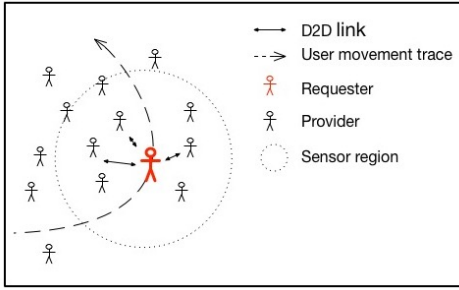


Fig. 2. SensorDistance.

gorithm is approximately optimal and performs much better than other standard composition approaches. We compare it with other population-based optimization methods for both optimality and scalability.

The remainder of this paper is organized as follows. Section II describe the MSON framework and its application scenario. Section III introduces the mobile service composition model. The approach to make service compositions is presented in Section IV. Section V presents experimental results. Section VI reviews the related work. Section VI concludes this paper.

2 MSON AND APPLICATION SCENARIO

In this section, we introduce the mobile service opportunistic network (MSON) formed by multiple mobile users. Within it, users can share services on their own mobile devices through opportunistic mobile networks. Its three main characteristics as follows [1].

1) Locality: An MSON is not established on the Internet, it do not consider user mobility a problem but as an opportunity to exploit. Mobile user in opportunistic network can sense each other's service and establish self-organized local communication network within sensor distance.

2) Dynamicity: MSON participants are not stationary. They can enter or leave the self-organized group at any time.

3) Mobility: Services provided by mobile users are not fixed at the same location. Service requesters are also mobile when invoking a mobile service.

Fig.2 illustrates the working procedure of the mobile service provision over opportunistic network. In an opportunistic network scenario, a mobile task requester R can sense mobile service exposed by nearby devices through D2D link and launches a mobile composition request.

A composer can be implemented and deployed on the requesters mobile device, which is in charge of discovering available mobile services nearby, composing multiple services, and selecting appropriate service candidates for composition tasks. Note that the service candidates are selected for all the tasks before the execution of the composition. During the execution of mobile service compositions, all the component services interact with the composer directly. The communication among mobile devices is based on Bluetooth, WLAN, or other available D2D communication protocols [1].

We should emphasize that, our framework only considers one-hop mechanism for both service provider and

consumer, since realistic dataset analyses reveal that users one-hop neighbors are sufficient [6] and can cover most range of the whole network in a reasonable time period [12], and contacting a user would incur long delay if the maximum D2D communication hops are larger than two [9]. Compared with multi-hop mechanisms in existing researches [13], [14], [15], [16], [17], [18][1], this one-hop feature can lower the network overhead (e.g., no need to transfer a large volume of task contents hop by hop) and ensure framework performance with only local information, which is more practical in real life [11]. Hence, once two mobile devices are out of their sensing distance, they would lose the connection [1].

we use an example depicted in Fig.1 to illustrate the related features of the problem of MSON. Assume mobile user Mike just complete his tour and now he is on the subway to airport. Now he wants to edit the video he recorded and add some effects and share the video clip to his friends. But due to mobile devices' limited battery, if he invoke clip service in his own device, his mobile phone will shutdown before he reach the destination because of run out of energy. As one option, he can upload the video to cloud services to get the video clip, but offloading quest into cloud will result in heavy cellular traffic, that means expensive communication fee and high energy consumption [?], [?], [?]. If Mike participate in MSON and servel video processing services is provided by some nearby mobile devices, Mike can invoke such mobile services on nearby mobile devices through free near field communication techniques. If these services cannot meet his requirement, several services can be composed. For example in Fig. 2, at most three services are needed: 1) video cutting service; 2) video beautifying service; and 3) video share service. Due to users mobility, the availability of service to Mike can vary, we will discuss mobile service availability in next section. In this environment invoking services provided by other mobile users may face new challenges that traditional composition methods cannot handle, thus, a mobile service composition model which can capture mobile services' availability need to be proposed [19].

3 MOBILE SERVICE COMPOSITION MODEL

In this section, we first give some basic definition of mobile web service, then introduce the concept of mobile service availability, and propose specific QoS model for mobile service composition over opportunistic network [19].

3.1 Preliminaries

In order to describe the problem addressed in this paper, we first provide the basic concepts of mobile service composition based on the proposed mobility model.

Definition 1 (Mobile Service): A mobile service is a triple $s = (uuid, Fun, QoS)$, where:

- 1) $uuid$ is the unique identifier of the service;
- 2) Fun is the set of functions s provides, a function includes the input, output, precondition and result of the service;
- 3) $QoS = \{q\}_{j=1}^n$ is a set of quality attributes, including execution cost, response time, reliability, availability, etc [19].

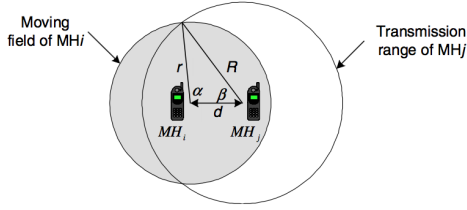


Fig. 3. SensorDistance.

Definition 2 (MSON participant): A MSON participant is mobile service user who can be both service provider and consumer, it can be represented by three-tuple $u = (uuid, P, C)$, where:

- 1) $uuid$ is the unique identifier of the provider;
- 2) P is the set of mobile services exposed by mobile service user u .
- 3) C is the set of perceived mobile services from nearby mobile service users.

Definition 3 (Mobile Service Composition Plan): A service composition plan is a tuple $scp = (T, R)$, where:

- 1) $T = \{t_1, t_2, \dots, t_n\}$ is a set of tasks;
- 2) $R = \{d(t_i, t_j) | t_i, t_j \in T\}$ is a set of relations between tasks in L .

A service composition plan is an abstract description of a business process. Each task t_i can be realized by invoking an individual service. There may be multiple services with different QoS that can be adopted to fulfill each task. R is used to describe the structure of the composition. $d(t_i, t_j) = 1$ represents that the inputs of t_j depend on the outputs of t_i [19].

Definition 4 (composite service instance): A service composition instance is a tuple $csi = (scp, S)$, where: 1) scp is mobile service composition plan which defined in definition 3;

- 2) $S = s_1, s_2, \dots, s_n$ is a set of selected concrete tasks.

3.2 Concept of Mobile Service Availability

In mobile service opportunistic network (MSON) the availability of service to its invoker is highly related to the users mobility. If user i moves outside the transmission range of its neighbouring user j , then user i is unreachable by user j and as a result the services on user i become unavailable to user j either. Here user mobility is utilized to calculate the mobile service availability [20].

Definition 5 (Mobile Service Availability): mobile service availability can be represented by a 3 tuple (I, P, V) , where

- 1) I is the mobile service consumer;
- 2) P is the mobile service provider;
- 3) V is the mobile service availability value between consumer and provider, $V \in [0, 1]$, and $V = 0$ means service provider moves out of service sensor range.

As illustrated in Fig. 2, consider two mobile users MU_i and MU_j of the same transmission range R . Each node moves randomly and it is assumed that the moving field is a circle with a radius of r . d represents the distance between MU_i and MU_j . These three parameters are to be used for calculating the node availability. The transmission range of a node R is known (e.g., pre-defined or changing according to

certain algorithm). Suppose the location (i.e., coordinates) of each mobile host is known (e.g., via GPSglobal positioning system, **How to get user's moving trace, refer S.Deng's work **), then distance d can be calculated using the Euclidean distance formula, i.e., $\sqrt{(x_i - x_j)^2 + y_i - y_j^2}$ where (x_i, x_j) and (y_i, y_j) are the coordinates of MU_i and MU_j respectively. Finally let us discuss how to calculate r [20].

The moving radius of a mobile user r is its speed s multiplied by the average service time t . Here t can be statistically calculated as the average value of last n servings of this component service, namely, $t = \sum_{i=1}^n t_i / n$. The speed of a mobile host s can be calculated based on its moving distance during a period from t_1 to t_2 [21], namely: $s = \sqrt{(x_i - x_j)^2 + y_i - y_j^2} / (t_2 - t_1)$. Then $r = s \times t$ [20].

Knowing the values of these three parameters R , r , and d , the probability of MU_i staying inside the transmission range of MU_j (denoted as P_i^{IN}) can be calculated by

$$P_i^{IN} = \frac{S_i^{IN}}{S_i^T} \quad (1)$$

Namely, P_i^{IN} equals to the area of the MH_i 's moving field inside the transmission range of MU_j (denoted as S_i^{IN}) divided by the overall area of the MU_i 's moving field (S_i^T) [20].

$$\alpha = \arccos\left(\frac{r^2 + d^2 - R^2}{2r \times d}\right) \quad (2)$$

$$\beta = \arccos\left(\frac{R^2 + d^2 - r^2}{2r \times d}\right) \quad (3)$$

Then,

$$\begin{aligned} S_i^{IN} &= \left[\left(\frac{2\beta}{2\pi} \pi R^2 \right) - \left(\frac{R \sin \beta \cos \beta}{2} \right) 2 \right] \\ &\quad + \left[\left(\frac{2\alpha}{2\pi} \pi r^2 \right) - \left(\frac{r \sin \alpha \cos \alpha}{2} \right) 2 \right] \\ &= \beta R^2 + \alpha r^2 - (R^2 \sin \beta \cos \beta + r^2 \sin \alpha \cos \alpha) \end{aligned} \quad (4)$$

There is also

$$S_i^T = \pi r^2 = \pi \times (s \times t)^2 \quad (5)$$

Therefore, we obtain

$$P_i^{IN} = \frac{S_i^{IN}}{\pi s^2 t^2} \quad (6)$$

The probability of MU_i staying inside the transmission range of MU_j (P_i^{IN}) can be calculated. Suppose service s running on MU_i is a candidate service for a task running on MH_j , and the node availability with regard to concrete service s is denoted as $q_{ava}(s)$, then finally there is

$$\begin{aligned} q_{ava}(s) &= P_i^{IN} = \frac{A_i^{IN}}{\pi s^2 t^2} \\ &= \frac{\beta R^2 + \alpha r^2 - (R^2 \sin \beta \cos \beta + r^2 \sin \alpha \cos \alpha)}{\pi s^2 t^2} \end{aligned} \quad (7)$$

Note that, mobile service availability $q_{ava}(s)$ is an important QoS attribute to construct QoS model over MSON for service selection in next section.

TABLE I
EXAMPLES OF AGGREGATION FUNCTIONS FOR QoS

	Time	Price	Reliability
\rightarrow	$\sum_{i=1}^n q_{time}(s_i)$	$\sum_{i=1}^n q_{price}(s_i)$	$\prod_{i=1}^n q_{rel}(s_i)$
\oplus	$\text{Max}\{q_{time}(s_i)\}$	$\sum_{i=1}^n q_{price}(s_i)$	$\prod_{i=1}^n q_{rel}(s_i)$
\otimes	$\sum_{j=1}^n p_j \times q_{time}(s_i)$	$\sum_{j=1}^n p_j \times q_{price}(s_i)$	$\sum_{i=1}^n p_j \times q_{rel}(s_i)$
$*$	$k \times q_{time}(s_i)$	$k \times q_{price}(s_i)$	$[q_{rel}(s_i)]^k$

Fig. 4. SensorDistance.

3.3 QoS Model for Mobile Service Composition

For mobile service consumer to select concrete service, they must consider QoS [22], [23], [24]. Common QoS attributes include response time, price, reliability, and reputation, mobile service availability is an important QoS attribute introduced in this paper to describe user's mobility behavior. for mobile service composition in this paper. and they can be classified into two categories: 1) positive and 2) negative (denoted as Q^+ and Q^-). For positive attributes, larger values indicate better performance (e.g., reputation and availability), while for negative attributes, smaller values indicate better performance (e.g., price and response time) [25].

For a composite service instance csi , the value of each QoS attribute is determined by the QoS values of its concrete components and orchestration patterns. Table I lists the aggregation functions for response time, price, and availability for sequential, parallel, choice, and loop composition patterns, where p_j represents the execution probability of the j th branch statement in the choice structure, and k is the expected number of iterations of the loop. More aggregation functions can be found in [26] and [27].

In order to facilitate ranking of different composite service instances in terms of QoS, simple additive weighting (SAW) is used as the QoS utility function to map the QoS vector into a real value. SAW first normalizes the QoS attribute values into real values between 0 and 1, through comparison with the maximal and minimal values; then it sums the normalized values multiplied with a preference weight w_t . According to SAW, the QoS utility of a composite service instance csi can be calculated using (1), where, $q_t(csi)$ is the aggregated value of the t -th QoS attribute of csi , and $q_{t,max}$ and $q_{t,min}$, respectively, denote the maximal and minimal possible aggregated values of the t th QoS attribute [25].

$$U(csi) = \sum_{q_t \in Q^-} \frac{q_{t,max} - q_t(csi)}{q_{t,max} - q_{t,min}} \times w_t + \sum_{q_t \in Q^+} \frac{q_t(csi) - q_{t,max}}{q_{t,max} - q_{t,min}} \times w_t \quad (8)$$

3.4 Problem Formulation

Base on the above discussion, we can give the definition of the MSON service composition problem.

Definition 6 (MSON Service Composition): Given a service composition request req by a mobile user u , sense and select suitable concrete services provided by near mobile service provider to achieve an optimal service composition instance

csi with the best QoS value. Meanwhile, csi should be the optimal in terms of QoS utility and also satisfy the users QoS constraints (qc), that is

$$\begin{aligned} & \text{Max. } U(csi) \\ & \text{subject to } q_t(csi) \geq qc_t \forall q_t \in Q^+ \\ & \quad q_t(csi) \leq qc_t \forall q_t \in Q^- \end{aligned} \quad (9)$$

Theorem 1: The service composition problem in MSSC (Definition 6) is NP-hard.

Proof: The standard integer program to find the smallest value of a given objective function $F(\Theta)$ with a feasible parameter follows [28]:

$$\begin{aligned} & \text{inf } F(\Theta) \\ & \text{subject to } \theta_I \in \{1, 2, 3, \dots, N\} \end{aligned} \quad (10)$$

This means that the feasible set of parameter vectors is constrained by $\theta_i \in 1, 2, \dots, N$ with integer values. The optimal solution Θ^* satisfies the following conditions.

- 1) Θ^* belongs to the feasible set.
- 2) $\forall \Theta, F(\Theta^*) \leq F(\Theta)$.

For the problem of selecting optimal services with best QoS value while considering mobility, the vector $\Theta = (1, \dots, n)$ can describe a possible solution as a service composition with n tasks. An element i in corresponds to a selected service from the candidates for the i th task. The evaluation function for the parameter vector is as follows:

$$F(\Theta) = \prod_{\theta_i} rt_{\theta_i} \quad (11)$$

The target of the mobile service composition problem in MSON is to find to obtain the biggest $F(\Theta)$. Thus, the problem is equivalent to the integer program described in (2). An integer programming problem is known to be NP-hard. Then the service composition problem in MSSC is NP-hard.

For such a problem, integer programming can be utilized to obtain the optimal solution. However, they might cost much more time with the increment of problem size. In mobile environment, the requirement on runtime is essential since the environment parameters for computation may vary much within a short time. Therefore, although integer programming can obtain the optimal result, it is not suitable to the problem due to its poor scalability. So one possible way to obtain a satisfactory solution in an accepted execution time is to design a heuristic search method and find the near optimal solution. Foreexample, meta-heuristic algorithms such as GAs and PSO, can be utilized to solve this problem. Among them, we find that KH algorithm can reduce the search space and return high approximate optima. Thus, we propose a solution method based on it to find an approximate optimal solution in polynomial time [1].

4 COMPOSITION ALGORITHM

In this section, we will illustrate our algorithm for making mobile service compositions over opportunistic based on the Krill-Herd algorithm [1].

1) Overview of KrillHerd: KH algorithm is a novel meta-heuristic swarm intelligence optimization method which is based on the simulation of herding krill individuals [29]. For KH, each population consists of krill individuals; each krill individuals represents a feasible mobile service composition solution. The position of an individual krill is determined by three main factors: 1) movement induced by other individuals; 2) foraging behavior; and 3) random diffusion. Fig.4 illustrate the overall process of the KH algorithm.

Table I shows the analogous term matches between the KH and mobile service composition problem. As shown in table, the position vector of each krill individual in the population corresponds to a feasible mobile service composition. The krill individual with the best position corresponds to the optimal mobile service composition. The motion induced by other krill individuals means to learn from other mobile service compositions, that is, to construct the current service composition by utilizing other service compositions experience. Similarly, the foraging motion is to learn from the current optimal service composition. The KH optimization's target is to find the krill individual with the best position, which means to find the best mobile service composition with the best fitness value. Therefore, once the optimal krill individual is found, the best mobile service composition is obtained.

It is known that an optimization algorithm should be capable of searching spaces of arbitrary dimensionality. Therefore, the following Lagrangian model is generalized to an n dimensional decision space:

$$\frac{dX_i}{dt} = N_i + F_i + D_i \quad (12)$$

where $X_i = (x_{i1}, x_{i2}, \dots, x_{in})$ is the position vector of the i -th krill individual (feasible service composition), n is the number of tasks in the service composition; x_{ij} is the selected candidate for the j -th task in solution X_i ; N_i is the motion induced by other krill individuals; F_i is the foraging motion, and D_i is the physical diffusion of the i -th krill individual.

2) Fitness Function: The penalty technique is used to drive the evolution toward constraint satisfaction [30]. In this paper, the penalty function is defined in (4) and (5), and it measures the negative total normalized distance from $q(csi)$ to the QoS constraints q_c when the constraint is violated. Then, in (6), the fitness function is defined as the weighted sum of the QoS utility $U(csi)$ and the penalty function $P(csi)$, where the penalty weight ($w_{pen} \times gen$) increases with the generation number gen . In this way, in the early generations, krill individuals violating the constraints but with high utility values can still be considered, and in the late generations, krill individuals violating the constraints are severely punished. Fitness function is defined as follow:

$$P(csi) = - \sum_{q_t \in Q} \frac{\Delta q_t}{q_{t,max} - q_{t,min}} \quad (13)$$

$$\Delta q_t = \begin{cases} q_{c_t} - q_t(csi) & \text{if} \dots \\ q_{c_t}(csi) - q_{c_t} & \text{if} \dots \\ 0 & \text{else} \dots \end{cases} \quad (14)$$

$$F(csi) = U(csi) + w_{pen} \times gen \times P(csi) \quad (15)$$

3) Motion Induced by Other Krill Individuals: This step is to optimize each composition by learning from other feasible compositions. The direction of motion induced by other krill individuals (feasible compositions) is evaluated by target swarm density (target effect), local swarm density (local effect), and repulsive swarm density (repulsive effect). For a krill individual (feasible composition), this operation can be defined as

$$N_i^{new} = N^{max} \alpha_i + \omega_n N_i^{old} \quad (16)$$

where

$$\alpha_i = \alpha^{local} + \alpha^{target} \quad (17)$$

where N^{max} is the maximum induced speed. According to the experimental values of the maximum induced speed, we set N^{max} to 0.01 (ms^{-1}) in this paper, $\omega_n \in [0, 1]$ the inertia weight of the induced motion; N_i^{old} is the induced motion of the previous iteration; α_i is the direction of the induced motion; α^{local} is the local effect provided by neighbor mobile service compositions and α^{target} is the target direction effect provided by the local optimal composition result.

The effect of neighbor mobile service compositions can be regarded as an attractive tendency among the feasible compositions for a local search. Such effect is determined as

$$\alpha_i^{local} = \sum_{j=1}^k \hat{F}_{i,j} \hat{X}_{i,j} \quad (18)$$

$$\hat{X}_{i,j} = \frac{X_i - X_j}{\|X_i - X_j\| + \varepsilon} \quad (19)$$

$$\hat{F}_{i,j} = \frac{F_i - F_j}{F_{worst} - F_{best}} \quad (20)$$

$$(21)$$

where F_{worst} and F_{best} are the worst and best fitness values of mobile service compositions at the current iteration, respectively. F_i represents the fitness value of the i -th mobile service composition calculated by (3), F_j is the fitness value of the j -th ($j = 1, 2, \dots, k$) neighbor mobile service composition, and k is the number of the neighbor mobile service compositions. For avoiding the singularities, a small positive number $\varepsilon = 0.0001$ is added to the denominator.

The neighbor service compositions of the i th service composition is determined by a sensing distance (ds) around it. The sensing distance for each service composition can be calculated in each iteration as follows:

$$d_{s,i} = \frac{1}{5N} \sum_{j=1}^N \|X_i - X_j\| \quad (22)$$

where $d_{s,i}$ is the sensing distance for the i -th service composition and N is the population size of the feasible solution set. If the Euclidean distance of two service composition vectors is less than $d_{s,i}$, they are assumed as neighbors.

To evaluate the target effect of each krill individual, the optimal individual of the current iteration with the best fitness value is taken into account by using

$$\alpha_i^{target} = C^{best} \hat{F}_{best} \hat{X}_{best} \quad (23)$$

where, C^{best} is the effective coefficient of the mobile service composition with the best fitness to the i -th krill individual. This coefficient is defined since α^{target} leads the solution to the local optima and it should be more effective than other neighbor mobile service compositions. The value of C^{best} is defined as follow:

$$C^{best} = 2(rand + \frac{I}{I_{max}}) \quad (24)$$

where $rand \in [0,1]$ is a random value to enhance exploration, I is the actual iteration count, and I_{max} is the maximum number of iterations.

3) Foraging Motion: Besides getting knowledge from its neighbors, a mobile service composition also gets knowledge from global optima, which is expressed as food attraction in the foraging motion. That means the KH algorithm not only gets partial knowledge but also global knowledge. The foraging motion is influenced by two main factors, 1) new food locations, 2) previous experiences about food locations. For the i -th mobile service composition, this motion can be defined as follows:

$$F_i = V_f \beta_i + \omega_f F_i^{old} \quad (25)$$

where V_f is the foraging speed (empirically set to 0.02 in this paper), $\omega_f [0,1]$ is the inertia weight of foraging motion, and F_i^{old} is the foraging motion in the previous iteration. β_i is the direction of the foraging motion

$$\beta_i = \beta_i^{food} + \beta_i^{best} \quad (26)$$

where β_i^{food} is the attractive food and β_i^{best} is the effect of the best fitness value of the i -th mobile service composition so far.

In KH, the virtual center of food concentration is approximately calculated according to the fitness value distribution of krill individuals, as it is inspired by "center of mass." It is formulated as follows:

$$X^{food} = \frac{\sum_{i=1}^N \frac{1}{F_i} X_i}{\sum_{i=1}^N \frac{1}{F_i}} \quad (27)$$

The food attraction for the i -th service composition can be determined as follows:

$$\beta_i^{food} = C^{food} F_{food} X_{food} \quad (28)$$

where C^{food} is the food coefficient. As the effect of food in the krill herding decreases during the time, the food coefficient is determined as

$$C^{food} = 2(1 - \frac{I}{I_{max}}) \quad (29)$$

The food attraction is defined to possibly attract the mobile service composition to get close to the global optimum. Based on this definition, the service compositions normally gather around the global optima after some iterations. This can be considered as an efficient global optimization strategy to help improve the convergence of the KH algorithm. The effect of the fitness value of the i -th mobile service composition can be calculated as follows:

$$\beta_i^{best} = \hat{F}_{i,best} \hat{X}_{i,best} \quad (30)$$

where $\hat{X}_{i,best}$ is the best previously generated position of the i -th krill individual.

4) Physical Diffusion: This step aims to keep the diversity of generated mobile service compositions in each iteration and avoid an early convergence. The physical diffusion of the service compositions is considered to be a random process. This motion can be expressed in terms of a maximum diffusion speed and a random directional vector, it can be formulated as follows:

$$D_i = D^{max} \delta \quad (31)$$

where D^{max} is the maximum diffusion speed and δ is a random directional vector with $\delta_i \in [1,1]$. In this paper, the maximum diffusion speed is randomly generated in $[0.002, 0.01]$. The better service composition is initialized, the fewer movements are required to find the optimal answer. The effect of the induced motion from other mobile service compositions and foraging motion gradually decreases with increasing iterations. Thus, (19) is modified to gradually decrease the random speed over iterations, that is

$$D_i = D^{max} (1 - \frac{I}{I_{max}}) \delta \quad (32)$$

5) Position Update: In KH, defined motions frequently change the positions of krill individual/mobile service compositions to get better fitness value. The foraging motion and induced motion by other krill individuals/mobile service compositions contain two global and two local strategies. These are working in parallel to make KH a powerful optimization algorithm. Using different effective parameters of the motion over time, the position vector of a krill individual/mobile service composition at time $t + \Delta t$ can be calculated as:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt} \quad (33)$$

where Δt is a scale factor of the speed vector and is set according to a search space, it can be obtained from

$$\Delta t = C_t \sum_{j=1}^d (UB_j - LB_j) \quad (34)$$

where d is the total number of tasks in each mobile service composition, and UB_j and LB_j are upper and lower bounds of candidate services for the j -th task, respectively. C_t is a constant value to scale the searching

space. Lower values of C_t lead to slower motion of krill individuals/monile service compositions, i.e., a thorough search.

5 SIMULATION AND EVALUATION

This section first describes simulation settings, then evaluates the effectiveness and scalability of our algorithm [25].

5.1 Simulation Setting

Since there is no available realistic datasets involving both user D2D contact records and user interest preference records, we attempt to simulate the scenarios for content transcription services by integrating realistic user contact traces with user interest preference in the folksonomy datasets. Specifically, we consider two typical user contact traces: Infocom06 and MIT Reality, where mobile users with Bluetooth-enabled devices periodically detect their peers nearby, and record contacts over several days. The reasons for selecting them are two-fold. First, the inter-encounter time of majority of users in them follows Exponential distribution, as evidenced by the previous researches [19], [22]. Second, Infocom06 and MIT Reality reflect diverse network scenarios (i.e., a dense conference and a sparse campus, respectively). Then, according to the user amount in user contact traces, we randomly select the corresponding number of user interest preference from the CiteULike dataset for simulation [11].

In addition, since the existing location-based user mobility datasets such as UCSD and Dartmouth do not involve user D2D contact records, we create a synthetic scenario for location-based services. That is, we consider the virtual city scenario built in ONE simulator where users move on the roads to visit our specified locations in terms of the Working Day Movement Model which captures the exponential property of user inter-encounter time [28]. In detail, we uniformly pick several locations in the city map, and randomly select the corresponding number of locations from Dartmouth dataset to match them. Then, we randomly select the visiting frequencies of a user with respect to those selected locations from the dataset for a mobile user in the scenario. In this context, each mobile user in the simulation will decide the next visiting location according to the (normalized) visiting frequencies, and will get there using the shortest path on the map [1].

5.2 Scalability Evaluation

5.3 Effectiveness Evaluation

6 RELATED WORK

Service composition is a significant area of research. As our proposal targets the problem of mobile service composition, we first briefly review the work on how to make such composition in traditional Internet environment, then we review some recent work on it in dynamic/mobile environments [19].

6.1 Traditional Web Service Composition

With the increasing popularity of employing service composition technologies for distributed systems, researchers started to pay more attention to quality of service (QoS)-aware Web services in recent years. Many algorithms and models are proposed to solve a service selection and composition problem, including integer programming, genetic algorithm (GA), and particle swarm optimization (PSO).

Tan et al. [31] introduced a Petri-net-based approach to compose services to meet the need of the business requirement. Wang et al. [32] proposed a constraint-aware service composition method in which service constraints are well taken care. The proposed solution includes a graph search-based algorithm and two novel preprocessing methods. Deng et al. [33] proposed correlation-aware service pruning method to address the QoS-correlation problem in service selection and composition. In order to provide more service compositions to meet users requirements, they proposed top-k service composition methods [1], [2]. Sun and Zhao [34] presented a constraint decomposition-based approach for service composition. They computed the utility of a composite service from the utilities of component services. The constraints of component services can be decomposed from the constraints of a composite service. Deng et al. [3] utilized GA to create service mashup in terms of cost performance in a cloud environment.

The above methods provide us with the inspiration of transferring a service composition problem to an optimization problem and utilize different methods to obtain an optimal or near-optimal solution. However, the above methods can only be used for traditional services because they do not consider service providers and requesters mobility.

6.2 Mobile Web Service Composition

In traditional composition, an implicit assumption is that services are running on stationary powerful servers. This assumption may, however, be invalidated by even modern mobile devices equipped with various sensors and computation-powers such as smart phones.

Mobile Web services result from advancements in mobile devices and communication technologies combined with latest developments in the area of SOA. Various mobile Web service provisioning approaches and applications are successfully prototyped in areas of mobile health, collaborative learning, entertainment, context-aware computing, location-based computing, and personal information exchange. At present, there are some service composition protocols carried out in dynamic pervasive environments [35]. The dynamic service composition protocol is implemented through distributed brokers. As for users mobility prediction, opportunistic computing is a way to obtain mobile users movement for service composition in dynamic environment. Groba and Clarke [36] report a novel service composition protocol that allocates and invokes service providers opportunistically to minimize the impact of topology changes and to reduce failure. Passarella et al. [37] give an analytical model that depicts the service invocation process between seekers and providers in an opportunistic network. Their works all provide technical support for implementing mo-

mobile service interactions among mobile devices they form a foundation of this work.

As mentioned earlier, service discovery and composition in mobile environments are relatively new areas of research. Han and Zhang [38] design and implement a service composition protocol based on the dynamic source routing protocol. It goes beyond best-effort approaches and considers QoS in real-time systems. Although their solution considers traditional QoS parameters such as delay and cost, they do not consider user-based features of a mobile device such as users mobility. Wang [39] provides a solution that incorporates a service providers mobility prediction. This provides more reliable composition in terms of mobility and the most stable services that can be selected from several services of the same type. This is an important aspect of composition that provides a valid extension to the work in this paper. However, the work [39] fails to consider the optimal QoS service composition.

Our proposed approach goes beyond existing ones by proposing a new mobile service provisioning architecture named MSSC where both service requesters and providers are nonstationary. Based on MSSC, we find a service composition with an optimal response time for all mobile users.

6.3 Mobile Opportunistic Network

Opportunistic networking is one of the most interesting evolutions of the multihop networking paradigm. Indeed, while MANET represents an engineering approach to hide node mobility by constructing stable end-to-end paths as in the Internet, opportunistic networks do not consider node mobility a problem but as an opportunity to exploit. In opportunistic networks the mobility of nodes creates contact opportunities among nodes, which can be used to connect parts of the network that are otherwise disconnected. Specifically, according to this paradigm (also referred to as delay-tolerant or challenged networks), nodes can physically carry buffered data while they move around the network area until they get in contact with a suitable next-hop node (i.e., until a forwarding opportunity exists). Thus, as opposed to a MANET, a node keeps on storing data when no good next hop exists. This implies that, with the opportunistic paradigm, a data can be delivered from a source toward a destination, even if an end-to-end path between them never exists, by exploiting the sequence of connectivity graphs generated by nodes movement [40], [41]. This is a relatively young paradigm, and opportunistic network research is still ongoing. Therefore, one may argue that its impact is still to be proved. However, given that we can consider vehicular ad hoc networking (VANET) one of the most advanced and concrete developments of the opportunistic networking paradigm, we can claim that the opportunistic network paradigm already has a significant role in the computer networking field. In addition to VANET, other scenarios, motivating opportunistic network use, are discussed in [42] ["Applications in Delay-Tolerant and Opportunistic Networks"]. Opportunistic networking looks very suitable for communications in pervasive environments where the environment is saturated by devices (with shortrange wireless technologies) that can self-organize in a network for local interactions among users. In these scenarios, the

network is generally partitioned in disconnected islands, which might be interconnected by exploiting the nodes mobility.

This implies a shift from legacy packet-based communication, toward message-based communication, bringing along new opportunities for application protocol design.

7 CONCLUSION

aaaasdad [?] [11].

REFERENCES

- [1] S. Deng, L. Huang, J. Taheri, J. Yin, M. Zhou, and A. Y. Zomaya, "Mobility-aware service composition in mobile communities," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 3, pp. 555–568, 2017.
- [2] S. Deng, L. Huang, H. Wu, W. Tan, J. Taheri, A. Y. Zomaya, and Z. Wu, "Toward Mobile Service Computing: Opportunities and Challenges," *IEEE Cloud Computing*, vol. 3, no. 4, pp. 32–41, 2016.
- [3] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, "A survey of mobile cloud computing: architecture, applications, and approaches," *Wireless communications and mobile computing*, vol. 13, no. 18, pp. 1587–1611, 2013.
- [4] X. Hu, X. Li, E. Ngai, V. Leung, and P. Kruchten, "Multidimensional context-aware social network architecture for mobile crowdsensing," *IEEE Communications Magazine*, vol. 52, no. 6, pp. 78–87, 2014.
- [5] S. Giordano and D. Puccinelli, "The Human Element as the Key Enabler of Pervasiveness," pp. 150–156, 2011.
- [6] S. Liu and A. D. Striegel, "Exploring the potential in practice for opportunistic networks amongst smart mobile devices," in *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 2013, pp. 315–326.
- [7] A. Asadi, Q. Wang, and V. Mancuso, "A survey on device-to-device communication in cellular networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1801–1819, 2014.
- [8] C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura, "Serendipity: enabling remote computing among intermittently connected mobile devices," in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2012, pp. 145–154.
- [9] Y. Li and W. Wang, "Can mobile cloudlets support mobile applications?" in *Infocom, 2014 proceedings IEEE*. IEEE, 2014, pp. 1060–1068.
- [10] Y. Zhang, D. Niyato, and P. Wang, "Offloading in mobile cloudlet systems with intermittent connectivity," *IEEE Transactions on Mobile Computing*, vol. 14, no. 12, pp. 2516–2529, 2015.
- [11] L. Pu, X. Chen, J. Xu, and X. Fu, "Crowd Foraging : A QoS-Oriented Self-Organized Mobile Crowdsourcing Framework Over Opportunistic Networks," vol. 35, no. 4, pp. 848–862, 2017.
- [12] J. Fan, J. Chen, Y. Du, P. Wang, and Y. Sun, "Delque: A socially aware delegation query scheme in delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 5, pp. 2181–2193, 2011.
- [13] W. Chang and J. Wu, "Progressive or conservative: Rationally allocate cooperative work in mobile social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 7, pp. 2020–2035, 2015.
- [14] M. Karaliopoulos, O. Telelis, and I. Koutsopoulos, "User recruitment for mobile crowdsensing over opportunistic networks," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*. IEEE, 2015, pp. 2254–2262.
- [15] Y. Han, T. Luo, D. Li, and H. Wu, "Competition-based participant recruitment for delay-sensitive crowdsourcing applications in d2d networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 2987–2999, 2016.
- [16] G. S. Tuncay, G. Benincasa, and A. Helmy, "Participant recruitment and data collection framework for opportunistic sensing: a comparative analysis," in *Proceedings of the 8th ACM MobiCom workshop on Challenged networks*. ACM, 2013, pp. 25–30.
- [17] J. Wu, M. Xiao, and L. Huang, "Homing spread: Community home-based multi-copy routing in mobile social networks," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 2319–2327.

- [18] C. Jiang, L. Gao, L. Duan, and J. Huang, "Exploiting data reuse in mobile crowdsensing," in *Global Communications Conference (GLOBECOM), 2016 IEEE*. IEEE, 2016, pp. 1–6.
- [19] S. Deng, L. Huang, Y. Li, H. Zhou, Z. Wu, X. Cao, M. Y. Kataev, and L. Li, "Toward risk reduction for mobile service composition," *IEEE transactions on cybernetics*, vol. 46, no. 8, pp. 1807–1816, 2016.
- [20] K. Yang, A. Galis, and H.-H. Chen, "Qos-aware service selection algorithms for pervasive service composition in mobile wireless environments," *Mobile Networks and Applications*, vol. 15, no. 4, pp. 488–501, 2010.
- [21] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," *Wireless networks*, vol. 6, no. 4, pp. 307–321, 2000.
- [22] J. Wu, L. Chen, Y. Feng, Z. Zheng, M. C. Zhou, and Z. Wu, "Predicting quality of service for selection by neighborhood-based collaborative filtering," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 2, pp. 428–439, 2013.
- [23] X. Luo, M. Zhou, Y. Xia, and Q. Zhu, "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1273–1284, 2014.
- [24] X. Luo, M. Zhou, Y. Xia, Q. Zhu, A. C. Ammari, and A. Alabdulwahab, "Generating highly accurate predictions for missing qos data via aggregating nonnegative latent factor models," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 3, pp. 524–537, 2016.
- [25] Q. Wu, F. Ishikawa, Q. Zhu, and D.-H. Shin, "QoS-Aware Multi-granularity Service Composition: Modeling and Optimization," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 11, pp. 1565–1577, 2016.
- [26] M. C. Jaeger, G. Rojec-Goldmann, and G. Muhl, "Qos aggregation for web service composition using workflow patterns," in *Enterprise distributed object computing conference, 2004. EDOC 2004. Proceedings. Eighth IEEE International*. IEEE, 2004, pp. 149–159.
- [27] H. Zheng, W. Zhao, J. Yang, and A. Bouguettaya, "Qos analysis for web service compositions with complex structures," *IEEE Transactions on Services Computing*, vol. 6, no. 3, pp. 373–386, 2013.
- [28] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers & operations research*, vol. 13, no. 5, pp. 533–549, 1986.
- [29] A. H. Gandomi and A. H. Alavi, "Krill herd: a new bio-inspired optimization algorithm," *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 12, pp. 4831–4845, 2012.
- [30] M. Gen and R. Cheng, "A survey of penalty techniques in genetic algorithms," in *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*. IEEE, 1996, pp. 804–809.
- [31] W. Tan, Y. Fan, M. Zhou, and Z. Tian, "Data-driven service composition in enterprise soa solutions: A petri net approach," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 686–694, 2010.
- [32] P. Wang, Z. Ding, C. Jiang, and M. Zhou, "Constraint-aware approach to web service composition," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 6, pp. 770–784, 2014.
- [33] S. Deng, H. Wu, D. Hu, and J. L. Zhao, "Service selection for composition with qos correlations," *IEEE Transactions on Services Computing*, vol. 9, no. 2, pp. 291–303, 2016.
- [34] S. X. Sun and J. Zhao, "A decomposition-based approach for service composition with global qos guarantees," *Information Sciences*, vol. 199, pp. 138–153, 2012.
- [35] S. Kalasapur, M. Kumar, and B. A. Shirazi, "Dynamic service composition in pervasive computing," *IEEE Transactions on parallel and distributed systems*, vol. 18, no. 7, 2007.
- [36] C. Groba and S. Clarke, "Opportunistic service composition in dynamic ad hoc environments," *IEEE Transactions on Services Computing*, vol. 7, no. 4, pp. 642–653, 2014.
- [37] A. Passarella, M. Conti, E. Borgia, and M. Kumar, "Performance evaluation of service execution in opportunistic computing," in *Proceedings of the 13th ACM international conference on Modeling, analysis, and simulation of wireless and mobile systems*. ACM, 2010, pp. 291–298.
- [38] S. Han and Y. Zhang, "Design and implementation of service composition protocol based on dsr," in *Parallel and Distributed Computing, Applications and Technologies (PDCAT), 2010 International Conference on*. IEEE, 2010, pp. 323–328.
- [39] J. Wang, "Exploiting mobility prediction for dependable service composition in wireless mobile ad hoc networks," *IEEE Transactions on Services Computing*, vol. 4, no. 1, pp. 44–55, 2011.
- [40] U. G. Acer, P. Drineas, and A. A. Abouzeid, "Connectivity in time-graphs," *Pervasive and Mobile Computing*, vol. 7, no. 2, pp. 160–171, 2011.
- [41] S. Ferretti, "Shaping opportunistic networks," *Computer Communications*, vol. 36, no. 5, pp. 481–503, 2013.
- [42] T. Melodia, H. Kulhandjian, L. Kuo, and E. Demirors, "Mobile ad hoc networking: cutting edge directions," *Advances in Underwater Acoustic Networking*, 2013.