

# Centrality-driven Scalable Service Migration

Panagiotis Pantazopoulos    Merkourios Karaliopoulos    and    Ioannis Stavrakakis

Department of Informatics and Telecommunications

National & Kapodistrian University of Athens

Ilissia, 157 84 Athens, Greece

Email: {ppantaz, mkaralio, ioannis}@di.uoa.gr

**Abstract**—As social networking sites provide increasingly richer context, user-centric service development is expected to explode following the example of User-Generated Content. A major challenge for this emerging paradigm is how to make these exploding in numbers, yet individually of vanishing demand, services available in a cost-effective manner; central to this task is the determination of the optimal service host location. We formulate this problem as a facility location problem and devise a distributed and highly scalable heuristic to solve it. Key to our approach is the introduction of a novel centrality metric. Wherever the service is generated, this metric helps to a) identify a small subgraph of candidate service host nodes with high service demand concentration capacity; b) project on them a reduced yet accurate view of the global demand distribution; and, ultimately, c) pave the service migration path towards the location that minimizes its aggregate access cost over the whole network. The proposed iterative service migration algorithm, called cDSMA, is extensively evaluated over both synthetic and real-world network topologies. In all cases, it achieves remarkable accuracy and robustness, clearly outperforming typical local-search heuristics for service migration. Finally, we outline a realistic cDSMA protocol implementation with complexity up to two orders of magnitude lower than that of centralized solutions.

## I. INTRODUCTION

A major change in the emerging networking landscape concerns the role of *end-user*, who is no more only content consumer but also generator. Web2.0 technologies have enabled this shift towards more user-centric paradigms, as evidenced in the *User-Generated Content (UGC)* wave in online social networks, blogs, and video distribution sites (e.g., YouTube). This trend has even motivated the rethinking of the Internet architecture [1], [2] and is more recently being spread towards the service domain [3] with the *User-Generated Service (UGS)* paradigm, whereby end-users are themselves engaged in the generation and distribution of *service facilities* [4]. Some recent efforts to enable end-user programming [5] typically involve mashup creation tools that extract data from web pages and create new applications [6].

Besides user-friendly interfaces [4], the UGS paradigm necessitates scalable distributed mechanisms for publishing, discovering, and moving service facilities within the network. In the case of next-generation networks, in particular, these mechanisms should conform to their self-organization properties. Our work addresses the problem of optimally placing

service facilities within the network to minimize the cost of accessing them. This is an instance of the *facility location* problems, typically formulated as 1- or generally  $k$ -median problems if facility replication is feasible [7]. Most of the proposed solutions to the problem are *centralized* (e.g., [8]): the optimal service location is determined by a single entity possessing global information for both the network topology and distribution of service demand across the network. Centralized approaches assume the existence of some super node that undertakes network-resources management tasks by gathering all required information and bearing the exclusive computation burden of the  $1(k)$ -median problem solution [7]. Moreover, it is willing to repeat these tasks in response to (even minor) user demand shifts or topology changes that may alter the optimal service location. Clearly, these approaches are neither scalable nor inline with the emerging self-organizing and autonomic networking paradigms.

Our approach instead is distributed and scalable: it solves *locally* much smaller-scale  $1(k)$ -median optimization problems and then *moves* the services towards their optimal location traversing an access cost-decreasing path. Moreover, it is innovative in the way it selects the nodes for the local  $1(k)$ -median problem. State-of-the-art distributed approaches (e.g., [9], [10]) recruit nodes from the immediate neighborhood of the service host. On the contrary, we invest additional effort on a more informed node selection, which promotes the “correct” directions of migration towards the optimal location. To achieve this, we devise a metric, called *weighted Conditional Betweenness Centrality (wCBC)*, that draws on *Complex Network Analysis (CNA)* [11]. Inline with what is reported elsewhere (e.g., [12]), the insights from CNA are instrumental in accelerating the service migration and directing it towards better locations.

In each service migration step, the metric serves two purposes. Firstly, it *identifies* those nodes that contribute most to the aggregate service access cost and pull the service strongly in their direction; namely, nodes holding a central position within the network topology and/or routing large service demand amounts. Secondly, it correctly *projects* the *attraction forces* these nodes exert to the service upon the current service location and facilitates a migration step towards the optimal location (fig. 1.a-b).

We detail the metric and our algorithm, called cDSMA, in Sections III and IV, respectively. We demonstrate its accuracy

This work has been partially supported by the EC IST-FET SOCIALNETS (FP7-IST-217141) project, the Marie Curie grant RETUNE (FP7-PEOPLE-2009-IEF-255409), and the National and Kapodistrian University of Athens.

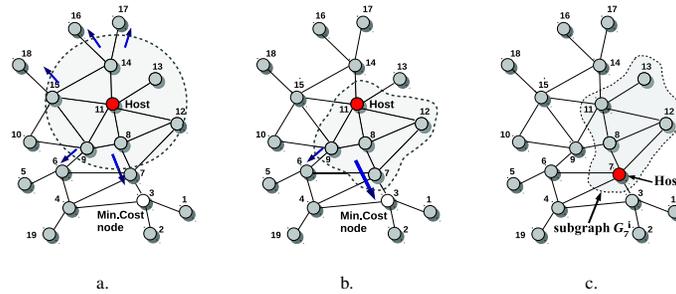


Fig. 1. a,b)Selected nodes and pulling forces for local-search heuristics(a) and cDSMA(b). c)Two non-zero demand mapping terms ( $w_{map}(8; 7)$ ,  $w_{map}(11; 7)$ ) for  $G_7^i$ .

and convergence properties in Section V. We outline a protocol implementation for cDSMA and its rough complexity analysis in Section VI and conclude our paper in Section VII.

## II. FACILITY LOCATION AND RELATED WORK

The optimal placement of service facilities within a network has been typically tackled as an instance of the facility location problems [7]. Given the network topology and demand dynamics, the *k-median* problem prescribes the locations for instantiating a fixed number of service facilities so as to minimize the aggregate cost of accessing them over all network users [7]. The facility location problem has been shown to be NP-hard for general topologies [7], [13], motivating various approximate solutions to it [8], [14]. Typically, the proposed solutions are centralized and require detailed global information about the network topology and service demand, so that their applicability in large-scale networks is undermined. Distributed approaches, on the other hand, have received renewed attention [15], largely because of the wide spread of mobile networks, where the network topology varies dynamically. One recently initiated research thread relates exactly to the *approximability* of distributed approaches to the facility location problem. Moscibroda and Wattenhofer in [16] draw on a primal-dual approach earlier devised by Jain and Vazirani in [17], to derive a distributed algorithm that trades-off the approximation ratio with the communication overhead under the assumption of  $O(\log n)$  bits message size, where  $n$  the number of clients. More recently, Pandit and Pemmaraju have derived an alternative distributed algorithm that compares favorably with [16] in resolving the same trade-off [18].

Although the approximability studies can yield provable bounds for the run time and quality of the solutions, they are typically outperformed by less mathematically rigorous yet practical heuristic solutions. Common to most of the distributed facility location schemes is the iterative *migration* of facilities from the generator host towards their optimal location through a number of locally-determined hops delineating a cost-decreasing path. Oikonomou and Stavrakakis in [9] exploit the shortest-path tree structures that are induced on the network graph by the routing protocol operation to estimate upper bounds for the aggregate cost in case the service migrates to the 1-hop neighbors. Migration hops are

therefore one physical hop long and this decelerates the migration process, especially in larger networks. Closer to our work is the upcoming paper of Smaragdakis *et al.* [10]. They reduce the original *k-median* problem in multiple smaller-scale 1-median problems solved within an area of *r-hops* from the current location of each service. Our approach resolves more efficiently the accuracy vs. convergence speed trade-off, especially on real-world topologies (see V-B) and excels over typical local-search oriented schemes (see V-C).

We focus on the single service facility scenario that matches better the expectations about the User-Generated Service paradigm, *i.e.*, various services generated within the network raising small-scale interest so that replication of their facilities be less attractive. The network topology is represented by an undirected connected graph  $G(V, E)$  of  $|V|$  nodes and  $|E|$  edges. If  $w(n)$  denotes the aggregate demand generated by node  $n$  for a specific service and  $d(k, n)$  denotes the minimum cost path between nodes  $k$  and  $n$ , then the 1-median problem formulation seeks to minimize the access cost of a service located at node  $k \in V$  given by:

$$Cost(k) = \sum_{n \in V} w(n) \cdot d(k, n). \quad (1)$$

Our heuristic employs Complex Network Analysis (CNA) to cope with the high complexity and scalability requirements. We introduce the *wCBC* metric, which assesses the value of nodes as candidate service hosts. The top-*wCBC* nodes induce a small subgraph on the original network, hereafter called *1-median subgraph*, wherein a significantly smaller-scale optimization problem is efficiently solved. Besides, the metric lets us directly map the demand of the rest of the network nodes on this subgraph and, thus, single-out a more informed 1-median subgraph than what relevant local-search heuristics do. Similar CNA insights have already been exploited for significantly improving network protocols. Examples are [12] and [19] that draw on CNA to derive improved DTN routing protocols.

## III. WEIGHTED CONDITIONAL BETWEENNESS CENTRALITY

*Central* to our *distributed* approach is the Weighted Conditional Betweenness Centrality (*wCBC*) metric devised to capture *both* topological and service demand information for each node.

1) *Capturing network topology: from BC to CBC.* Betweenness Centrality (BC), a common CNA metric, reflects to what extent a node lies on the shortest paths linking other nodes. Let  $\sigma_{st}$  denote the number of shortest paths between any two nodes  $s$  and  $t$  in a connected graph  $G = (V, E)$ . If  $\sigma_{st}(u)$  is the number of shortest paths passing through the node  $u \in V$ , then the *betweenness centrality* of  $u$  is given by

$$BC(u) = \sum_{s=1}^{|V|} \sum_{t=1}^{s-1} \frac{\sigma_{st}(u)}{\sigma_{st}} \quad (2)$$

$BC(u)$  which captures the ability of a node  $u$  to control the establishment of paths between pairs of nodes, is an average value over all such node pairs. In [20] we proposed Conditional BC (*CBC*), as a way to capture *solely* the topological centrality of a random node with respect to a specific node  $t$ . It is defined as

$$CBC(u; t) = \sum_{s \in V, u \neq t} \frac{\sigma_{st}(u)}{\sigma_{st}} \quad (3)$$

with  $\sigma_{st}(s) = 0$ . The summation is over all node pairs  $(x, t) \forall x \in V$  destined at node  $t$  rather than all possible pairs, as in  $BC(u)$ . Effectively, *CBC* assesses to what extent a node  $u$  acts as a *shortest path aggregator* towards the current service location  $t$ .

2) *Capturing service demand: from CBC to wCBC.* Generally, a high number of shortest paths through  $u$  does not necessarily mean that equally high demand load stems from the sources of those paths. Naturally, we need to enhance the pure topology-aware *CBC* metric in a way that it takes into account the service demand that will be eventually served by the shortest paths routes towards the service location. To this end, we introduce *weighted conditional betweenness centrality* (*wCBC*), where the shortest path ratio is modulated by the demand load generated by each node  $s$ .

$$wCBC(u; t) = \sum_{s \in V, u \neq t} w(s) \cdot \frac{\sigma_{st}(u)}{\sigma_{st}}. \quad (4)$$

Since  $\sigma_{ut}(u) = \sigma_{ut}$ , the  $wCBC(u; t)$  value of each node  $u$  is lower bounded by its own demand  $w(u)$ . Therefore, *wCBC* assesses to what extent a node can serve as *demand load concentrator* towards a given service location. When all network nodes generate equal demand for a service (the uniform case), *wCBC* metric degenerates to the *CBC* one, within a constant factor. In [21] we derive analytical expressions for the *wCBC* metric in regular topologies.

#### IV. THE CDSM ALGORITHM DESCRIPTION

Our centrality-driven Distributed Service Migration Algorithm (cDSMA) steers the service towards its optimal location via a finite number of steps.

*Step 1: Initialization.* The algorithm execution starts at the node  $s$  that initially generates the service facility. The service placement cost at node  $s$  is assigned an infinite value (pseudocode *line 3*) to secure the first iteration (*line 11*).

*Step 2: Metric computation and 1-median subgraph derivation.* Next, the computation<sup>1</sup> of  $wCBC(u; s)$  metric takes place for every node  $u$  in the network graph  $G(V, E)$ . Nodes featuring the top  $\alpha\%$  *wCBC* values together with the *Host* node currently hosting the service form the subgraph  $G_{Host}^i$  ( $i$  enumerates the iterations), over which the 1-median problem will be solved (*lines 4 – 5 and 14 – 15*). Clearly, the size of this subgraph and the complexity in the problem solution are directly affected by the choice of the parameter  $\alpha$ .

*Step 3: Mapping the demand of the remaining nodes on the subgraph.* By restricting the solution domain to the  $G_{Host}^i$  subgraph, the contribution of the “outside world” to the service provisioning cost would be totally neglected. To allow for its inclusion, the demand for service from the  $G \setminus G_{Host}^i$  nodes (*i.e.*, the non-shaded nodes in fig. 1c) is mapped on the  $G_{Host}^i$  ones. To capture the attraction forces due to the service demand of the outside nodes correctly and with no redundancy, the demand of some outside node  $z$  is credited only to the first “entry”  $G_{Host}^i$  node encountered on each shortest path from  $z$  towards the service host. Thus, the weights  $w(n)$  in the access cost calculation for the nodes  $n$  in the  $G_{Host}^i$  subgraph (see Section II) are replaced by *effective demands*:  $w_{eff}(n; Host) = w(n) + w_{map}(n; Host)$ , where (assuming that *Host* is node  $t$ ):

$$w_{map}(n; t) = \sum_{z \in \{G \setminus G_{Host}^i\}} w(z) \frac{\sigma'_{zt}(n)}{\sigma_{zt}} \quad (5)$$

$$\sigma'_{zt}(n) = \sum_{j=1}^{\sigma_{zt}} \mathbb{I}_{\{n \in SP_{zt}(j) \cap n = \underset{u \in SP_{zt}(j)}{\operatorname{argmin}} d(z, u)\}}$$

with  $SP_{zt}(j)$  standing for the  $j^{th}$  element of the shortest path set from node  $z$  to node  $t$ . For example, in fig. 1c the original service demand of node, say, 16 is not mapped on all the  $G_7^i$  nodes lying on the shortest paths from 16 to the *Host* 7 (*i.e.*, 11, 12 and 8), but only on 11.

*Step 4: 1-median problem solution and service migration to the new host node.* Any centralized technique (*e.g.*, [14]) may be used to solve this small-scale optimization problem and determine the optimal location of the new *Host* among the  $G^i$  nodes. We assign the value of this cost to  $C_{next}$ ; if it is smaller than  $C_{current}$  the service is moved to this node and the algorithm iterates through steps 2-4, steering the service, progressively, to the (globally) lowest-cost location.

We complete the description elaborating on the cDSMA convergence. Lemma 1 serves as the basis for the proof of the convergence proposition.

**Lemma 1.** A service facility following the migration process of Algorithm 1 will visit at most one network node twice.

*Proof.* Assume that the service, initially deployed at some node  $n \in G$  reaches the node  $b \in G$  twice. Right after its first placement at  $b$  upon iteration, say,  $i - 1$  we solve the 1-median in the subgraph  $G_b^i$  formed by the nodes with the top

<sup>1</sup>For the actual *wCBC* computation, which involves solving the all-pairs shortest path problem, we properly modified the scalable algorithm in [22] for *betweenness centrality* computation, with runtime  $O(|V||E|)$ .

---

**Algorithm 1** cDSMA in  $G(V,E)$ 

---

```
1. choose randomly node  $s$ 
2. place SERVICE @  $s$ 
3.  $C_{current} \leftarrow \infty$ 
4. for all  $u \in G$  do compute  $wCBC(u; s)$ 
5.  $G_s^o \leftarrow \{\alpha\% \text{ of } G \text{ with top } wCBC \text{ values}\} \cup \{s\}$ 
6. for all  $u \in G_s^o$  do
7.   compute  $w_{map}(u; s)$ 
8.    $w_{eff}(u; s) \leftarrow w_{map}(u; s) + w(u)$ 
9.  $Host \leftarrow 1\text{-median solution in } G_s^o$ 
10.  $C_{next} \leftarrow C(Host)$ ,  $i \leftarrow 1$ 
11. while  $C_{next} < C_{current}$  do
12.   move SERVICE to  $Host$ 
13.    $C_{current} \leftarrow C_{next}$ 
14.   for all  $u \in G$  do compute  $wCBC(u; Host)$ 
15.    $G_{Host}^i \leftarrow \{\alpha\% \text{ of } G \text{ with top } wCBC \text{ values}\} \cup \{Host\}$ 
16.   for all  $u \in G_{Host}^i$  do
17.     compute  $w_{map}(u; Host)$ 
18.      $w_{eff}(u; Host) \leftarrow w_{map}(u; Host) + w(u)$ 
19.    $NewHost \leftarrow 1\text{-median solution in } G_{Host}^i$ 
20.    $Host \leftarrow NewHost$ ,  $C_{next} \leftarrow C(NewHost)$ ,  $i \leftarrow i + 1$ 
21. end while
```

---

$\alpha\%$   $wCBC(u; b)$  values. Let the corresponding cost be  $C_b^i$ . When the service returns to  $b$  at iteration, say,  $j$  given that the network topology remains the same, the deterministic  $wCBC$  criterion of eq. (4) singles out the same subgraph with the one of the first visit, so we have that  $G_b^i = G_b^j$ , implying for the costs that  $C_b^i = C_b^j$ ; the cost-decreasing condition of cDSMA is then not fulfilled and, thus, the service locks at node  $n$  and the migration process halts.

**Proposition 1.** cDSMA converges at some solution in  $O(|V|)$  steps.

*Proof.* The solution derived from cDSMA is either the globally optimal or one locally anticipated as lowest-cost solution. Given the *Lemma 1* and since the number of network nodes is finite, the migrating service will visit every node at most once and only one of them, twice. This takes  $O(|V| + 1) = O(|V|)$  steps.

## V. EVALUATION OF CDSMA

Clearly, both the network topology and service demand distribution across the network affect the performance of cDSMA. Network topological properties may give rise to or, on the contrary, suppress strong service demand poles and assist (resp. impede) the migration progress of the service facilities towards preferable locations. The *accuracy* of cDSMA is measured by the *average normalized excess cost*,  $\beta_{alg}$  (or  $\beta_{alg}(\alpha)$ ) to capture the percentage of nodes included in the 1-median subgraph, defined as the ratio of the service access cost our algorithm achieves, over the cost achieved by the optimal solution (derived by a brute-force centralized algorithm assuming availability of global topology and demand information), for some network topology  $G$  and service demand distribution  $w$ ; clearly, the error induced by our heuristic decreases with  $\alpha$ . Closely related to  $\beta_{alg}(\alpha)$  is the index  $\alpha_\epsilon$ , corresponding to the minimum value of  $\alpha$  for which the access cost achieved by cDSMA falls within  $100 \cdot \epsilon\%$  of the optimal,  $\alpha_\epsilon = \operatorname{argmin} \{\alpha | \beta_{alg}(\alpha) \leq (1 + \epsilon)\}$ . Moreover, we assess the *convergence speed* of cDSMA through the *migration hop*

*count*,  $h_m$ , reflecting how fast the algorithm converges to its (sub)optimal solution. Small  $h_m$  values imply fast service deployment and less overhead involved to transport and service set-up/shut-down tasks. For any chosen configuration of the involved parameters, we repeat 20 simulation runs, choosing each time randomly the service generation nodes, and plot their means together with their 95% confidence intervals.

### A. Experiments with synthetic topologies

Before assessing the cDSMA performance on real-world networks, we seek to draw some insights<sup>2</sup> from the inspection of two synthetic topologies with very different structural properties, namely the Barabási-Albert<sup>3</sup>(B-A) [23] and the two-dimensional ( $N \times M$ ) rectangular grid graphs. B-A graphs form probabilistically and exhibit highly skewed degree distribution. Grids, on the other hand, have strictly regular structure with constant node degree and diameter exponentially growing with the network size. These topologies let us assess the algorithm under extreme yet predictable conditions.

1) *Uniform demand.* Though non-realistic, uniform demand patterns are insightful since they can isolate the impact of network topology on cDSMA. Figures 2.a,b plot the average normalized excess cost  $\beta_{alg}$  for B-A and grid graphs of 100 nodes, respectively. The error induced by cDSMA decreases monotonically with the 1-median subgraph size in both plots. However, the behavior of cDSMA on the B-A graph is better as the aggregate service access cost increase is within 2% of the optimal, even when we include 10% of network nodes in the 1-median subgraph. Similar accuracy for the grid would require, on average, no less than 40% of the nodes. The high-degree hub nodes of B-A graphs [23] appear to ease the algorithm operation. Placing the service on, or nearby, hub nodes suffices for getting a very good solution already for small 1-median subgraphs. On the contrary, the regular structure of grids does not favor the identification of cost-effective locations. The service migration jumps within the grid are clearly shorter than in B-A graphs, often restricted to one-hop distance. Even worse, cDSMA gets more often trapped and terminates prematurely in suboptimal locations. In other words, the attraction force of the optimal location(s) is not impelling enough to pull the migrating service all the way to it.

This cDSMA performance differentiation over the two graphs is amplified when the network size and diameter grow. Table I lists the accuracy and migration hop count,  $h_m$ , against the network and 1-median subgraph size,  $N$  and  $\alpha$ , respectively. cDSMA's trend to abort early the migration process only deteriorates with the increase of network size and diameter. This is reflected in both the higher  $\beta_{alg}$  and the slightly increasing yet overly low  $h_m$  values in Table I. The significantly higher variance in the convergence speed

<sup>2</sup>A comprehensive study of cDSMA on synthetic topologies appears in [21].

<sup>3</sup>The B-A network requirement of the node degree distribution exponent being equal to 3 appears with network sizes of 1000 nodes or more. Therefore, the employed scale-free networks of a few hundred nodes' size are small to be called B-A networks; this name is only kept for the sake of clarity.

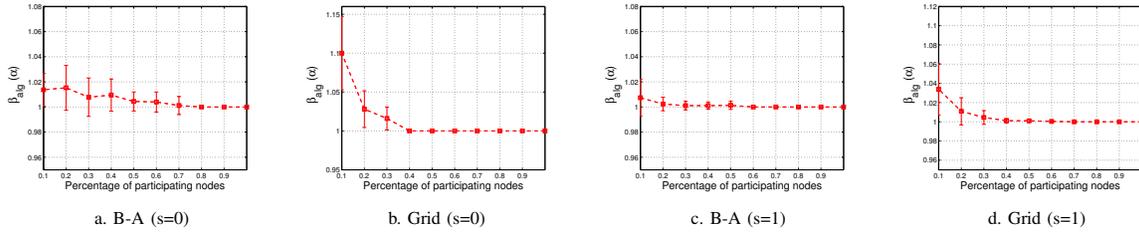


Fig. 2. Synthetic topologies of 100 nodes : cDSMA accuracy vs. 1-median subgraph size under uniform (a,b) and Zipf (c,d) demand distribution.

TABLE I  
CDSMA PERFORMANCE ON B-A AND GRID NETWORKS UNDER UNIFORM DEMAND

Network size N	B-A graph				Grid network			
	$\beta_{alg}(0.1)$	$h_m(0.1)$	$\beta_{alg}(0.4)$	$h_m(0.4)$	$\beta_{alg}(0.1)$	$h_m(0.1)$	$\beta_{alg}(0.4)$	$h_m(0.4)$
50 (25x2)	$1.0453 \pm 0.0524$	$2.25 \pm 0.31$	$1.0125 \pm 0.0186$	$1.95 \pm 0.28$	$1.0074 \pm 0.0071$	$1.40 \pm 0.35$	$1.0086 \pm 0.0058$	$1.10 \pm 0.22$
100 (25x4)	$1.0134 \pm 0.0169$	$2.00 \pm 0.32$	$1.0070 \pm 0.0164$	$2.00 \pm 0.00$	$1.0569 \pm 0.0333$	$1.30 \pm 0.33$	$1.0006 \pm 0.0012$	$1.20 \pm 0.29$
200 (40x5)	$1.0216 \pm 0.0327$	$2.00 \pm 0.00$	$1.0028 \pm 0.0061$	$1.95 \pm 0.16$	$1.0636 \pm 0.0487$	$1.60 \pm 0.71$	$1.0013 \pm 0.0043$	$2.05 \pm 0.59$
300	$1.0125 \pm 0.0147$	$2.00 \pm 0.00$	$1.0032 \pm 0.0070$	$2.00 \pm 0.00$				

of the algorithm implies dependence on the service generation location, *i.e.*, initial host. On the contrary, the cDSMA performance over B-A graphs is scalable as: a) it maintains practically the same accuracy as the network size grows; and b) its convergence speed is not affected by the network size. It reaches a near-optimal location within an average of two migration hops.

2) *Non-uniform demand.* To assess cDSMA under the *simultaneous* influence of network topology *and* demand dynamics we introduce asymmetry in the demand distribution. Mathematically speaking, a Zipf distribution models the (spatially random) preference  $w(n; s, N)$  of nodes  $n, n \in \mathcal{N}$  to a given service:  $w(n; s, N) = \frac{1/n^s}{\sum_{i=1}^N 1/i^s}$ , practically corresponding to the normalized service request rate. The relative  $h_m$  and  $\alpha$  values under Zipf demand with  $s = 1$  are depicted in Table III. The accuracy of cDSMA for B-A graphs improves further against the respective uniform demand values (shown in fig. 2 as well). The demand asymmetry magnifies the existing attraction forces and directs cDSMA towards the globally optimal service location. On the other hand, cDSMA requires far more nodes within the 1-median subgraph to yield comparable accuracy over grids.

When spatial correlation of demands is involved (*i.e.*, the interest in the service is concentrated in a particular neighborhood, as when the service has strongly local scope), a cluster of nodes with high service demand appears in a random area within the grid. Let  $K$  cluster nodes collectively represent some percentage  $z\%$  of the total demand for the service, whereas the other  $N - K$  nodes share the remaining  $(100 - z)\%$  of the demand. We call the ratio  $z/(100 - z)$  the demand *spatial contrast*  $C_{sp}$ . In 2D grids, clusters are formed by a cluster head node together with its  $R$ -hop neighbors. The contrast can then be written as:  $C_{sp}(R, s) = \frac{\sum_{n=1}^K w(n; s, N)}{\sum_{n=K+1}^N w(n; s, N)} = \frac{\sum_{n=1}^K 1/n^s}{\sum_{n=K+1}^N 1/n^s}$  and the average normalized excess cost becomes a function of both  $\alpha$  and the contrast value. The values of  $\beta_{alg}(\alpha, C_{sp})$  under spatially random and correlated ( $R = 1$ ) distribution of demands are reported in Table II for a  $10 \times 10$  grid topology. Having the top demand values stemming from a

TABLE II  
IMPACT OF SPATIALLY CORRELATED SERVICE DEMANDS

skewness $s$	$C_{sp}(1, s)$	$\beta_{alg}(0.1)$	$\beta_{alg}(0.1, C_{sp})$
1	0.786	$1.035 \pm 0.027$	$1.016 \pm 0.023$
2	8.540	$1.003 \pm 0.006$	$1.0 \pm 0.0$

certain network neighborhood we actually “produce” a single pole of strong attraction for the migrating service. cDSMA now follows the demand gradient more effectively than before. As the percentage of the total demand held by the cluster nodes grows larger (*i.e.*, higher  $C_{sp}$ ), the pole gets even stronger driving the service firmly to the optimal location.

The above results suggest that the higher the asymmetry in either the network topology or the service demand distribution, the better the performance of our algorithm. In the sequence, we investigate manifestations of this general rule in real-world networks.

### B. Experiments with real-world network topologies

The ultimate assessment of cDSMA is carried out over real-world ISP network topologies which do not typically have the predictable properties of B-A graphs and grids; still, we show below that insightful analogies regarding the behavior of cDSMA can be drawn between real-world and synthetic topologies. The dataset we consider has been recently made available [24]. It includes numerous data files (corresponding to Tier-1, Transit and stub level networks) that represent snapshots of 14 ISP topologies. We focus on the larger Transit and Tier-1 ISP datafiles<sup>4</sup>, with sizes up to approximately 1000 nodes and show results for a representative subset (see [21] for further results) featuring adequate variance in size, diameter, and connectivity degree statistics. Table IV summarizes the performance of cDSMA over the real-world topologies. It reports the minimum number of nodes  $|G^i|$  required to achieve a solution that lies within 2.5% of the optimal and the average migration hop count  $h_m$  for different levels of service demand asymmetry.

<sup>4</sup>Several files miss some edges resulting in more than one connected components [24]. Thus, a pre-processing task using a linear-time algorithm [25], is needed to retrieve the maximal connected component  $mCC$ .

TABLE III  
CDSMA PERFORMANCE ON B-A AND GRID NETWORKS UNDER ZIPF DEMAND ( $s = 1$ )

Network size N	B-A graph				Grid network			
	$\beta_{alg}(0.1)$	$h_m(0.1)$	$\beta_{alg}(0.4)$	$h_m(0.4)$	$\beta_{alg}(0.1)$	$h_m(0.1)$	$\beta_{alg}(0.4)$	$h_m(0.4)$
50 (25x2)	1.0156±0.0205	1.60±0.48	1.0014±0.0038	1.85±0.35	1.0083±0.0068	1.50±0.37	1.0062±0.0047	1.10±0.22
100 (25x4)	1.0070±0.0143	2.15±0.35	1.0015±0.0034	1.90±0.22	1.0553±0.0319	1.35±0.35	1.0025±0.0020	1.15±0.26
200 (40x5)	1.0016±0.0031	1.90±0.22	1.0003±0.0007	2.05±0.16	1.0510±0.0346	1.47±0.73	1.0031±0.0047	1.90±0.65
300	1.0029±0.0068	2.05±0.16	1.0000±0.0000	2.00±0.00				

The  $\alpha_{0.025}$  and  $|G^i|$  values show remarkable insensitivity to both topological structure and service demand dynamics. Although the considered ISP topologies differ significantly in size and diameter, the required 1-median subgraph size does not change. Around half a dozen nodes suffices to get very good accuracy even under uniform demand distribution, the least favorable scenario for cDSMA, as discussed in the previous section. Likewise,  $\alpha_{0.025}$  and  $|G^i|$  remain practically invariable with the demand distribution skewness. Although for larger values of  $s$ , few nodes become stronger attractors for the algorithm, the added value for its accuracy is negligible. This two-way insensitivity of cDSMA has important practical implications: a) the computational complexity of the local 1-median problem can be negligible and scales well with the size and diameter of the network; b) the algorithm performance is robust to possibly inaccurate estimates of the service demand each node poses. The equally good algorithm behavior under uniform demand distribution suggests that there is already adequate topological structure in these real-world networks. As their degree distribution shows (see [21]), there are high-degree nodes and considerable variance in the nodes' connectness properties across the network. In fact, the high-degree nodes serve similarly to the hub nodes in B-A graphs; they are easily "identifiable" by cDSMA as low-cost hosts and, even for small 1-median subgraph sizes, their attraction forces are strong enough to pave a cost-effective service migration path.

### C. cDSMA vs. locality-oriented service migration

The way cDSMA determines the service migration path clearly differentiates from typical "local-search" approaches such as the  $r$ -ball heuristic [10], which restricts *a-priori* their search for a better service host to the neighborhood of the current service location. On the contrary, cDSMA focuses its search for the next service host in certain directions. The resulting 1-median subgraph is spatially stretched across that path -consisting of highly "central" nodes- and therefore oversteps the local neighborhood "barriers". This is clearly illustrated in fig. 3 showing the hopcount distribution between all 1-median subgraph nodes and the host node, for five executions of cDSMA under Zipf demand with  $s=2$ .

To compare the two approaches, we have implemented a Locality-Oriented Migration heuristic (LOM). With LOM we solve the 1-median problem within the direct neighborhood of  $r = 2$  hops around the current host and apply the same demand mapping mechanism (Section IV) to capture the demand from nodes lying further than  $r$  hops away from the current service host. The comparison between LOM and cDSMA, illustrated in Table V, proceeds as follows. We first generate asymmetric service demand (Zipf distribution with  $s = 1$ ) across the

network. We compute the globally optimal service host node and we select a fixed set of service generation nodes, at  $D_{gen}$  hops away from the optimal location. We then calculate the values of  $h_m$  and  $\beta_{alg}$  metrics<sup>5</sup> for the two approaches. For cDSMA, we have set the parameter  $\alpha = 3\%$ , yielding 1-median subgraphs of 6 to 12 nodes' size (Table V).

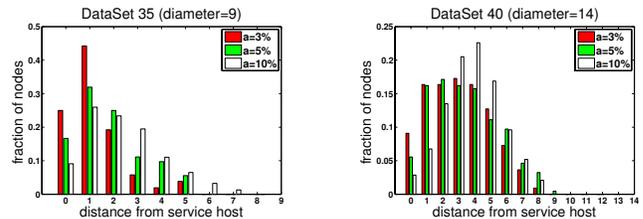


Fig. 3. Hopcount distribution of 1-median nodes from *Host* under cDSMA.

Since migration hops under LOM are spatially restricted (*i.e.*,  $D_{gen}/r$  hops to reach the optimal location), one would expect that the latter approach needs more migration hops than cDSMA. Interestingly enough, LOM combines high excess costs with fewer migration hops, irrespective of the service generation location or topology. Selecting "blindly" the  $r$ -hop neighbors of the current service host as future candidate hosts, LOM effectively introduces noise to the detection of the cost-effective service migration direction; the 1-median subgraph nodes are spread unidirectionally around the service host and the mapping process projects more uniformly the demand of the remaining nodes on them (see fig. 1.a-b). Consequently, the migrating service gets easily trapped in local minima and the process stops prematurely, resembling the behavior of the cDSMA in grids under uniform demand. There it was the topology of the network that induced a more local 1-median subgraph and attenuated the attraction force towards the optimal; with LOM, this locality is inherently imposed by the method *a-priori*.

On the other hand, the cDSMA seeks to choose the most "appropriate" candidate hosts to lead the service fast to cost-minimizing locations, no matter what the shape/radius of the emerging  $G^i$  neighborhood would be. Even when both approaches are trapped in a suboptimal place, LOM needs more hops to reach it than cDSMA, *e.g.*, three *vs.* one hops for Dataset 33 and  $D_{gen}=7$ . Moreover, facilitating longer migration hops, cDSMA does not favor nodes according to their proximity to the service demand and/or network topology hot spots, inducing a notion of fairness in the performance that different users get.

<sup>5</sup>The void entries are due to the fact that the most distant node to the global minimum location, lies at some distance smaller than the  $D_{gen}$  value.

TABLE IV  
MEAN VALUE OF  $\alpha_\epsilon$  FOR VARIOUS DATASETS UNDER DIFFERENT DEMAND DISTRIBUTIONS

ISP	Dataset id/AS#	mCC nodes	Diameter	<Degree>	s=0		s=1		s=2	
					$\alpha_{0.025}$	$ G^* $	$\alpha_{0.025}$	$ G^* $	$\alpha_{0.025}$	$ G^* $
<i>type: Tier-1</i>										
Global Crossing	36/3549	76	10	3.71	0.047±0.001	4	0.047±0.002	4	0.046±0.001	4
-/-	35/3549	100	9	3.78	0.045±0.002	5	0.045±0.001	5	0.043±0.001	5
NTTC-Gin	33/2914	180	11	3.53	0.024±0.002	5	0.022±0.002	4	0.019±0.002	4
Sprint	23/1239	184	13	3.06	0.019±0.002	4	0.018±0.002	4	0.017±0.002	4
-/-	21/1239	216	12	3.07	0.016±0.002	4	0.016±0.002	4	0.014±0.003	4
Level-3	27/3356	339	24	3.98	0.018±0.002	7	0.017±0.002	6	0.014±0.003	5
-/-	13/3356	378	25	4.49	0.012±0.002	5	0.012±0.002	5	0.011±0.002	5
<i>type: Transit</i>										
TDC	46/3292	71	9	3.30	0.033±0.003	3	0.027±0.004	2	0.026±0.003	2
DFN-IPX-Win	41/680	253	14	2.62	0.019±0.003	5	0.015±0.003	4	0.015±0.003	4
JanetUK	40/786	336	14	2.69	0.012±0.003	5	0.012±0.002	5	0.013±0.002	5

TABLE V  
CONVERGENCE SPEED AND ACCURACY OF LOM AND cDSMA ON REAL-WORLD TOPOLOGIES

$D_{gen}$	Dataset 23				Dataset 33				Dataset 27				Dataset 13			
	LOM		cDSMA		LOM		cDSMA		LOM		cDSMA		LOM		cDSMA	
	$h_m$	$\beta_{alg}$	$h_m$	$\beta_{alg}(3\%)$												
3	1	1.1050	2	1	1	1.0308	2	1	1	1.1109	1	1.0057	1	1.1054	1	1
4	1	1.1275	3	1	1	1.3206	2	1	1	1.2523	1	1.0057	1	1.2312	1	1
5	1	1.1632	2	1	1	1.2800	1	1.2800	2	1.1109	1	1	1	1.0434	2	1
7	1	1.6060	2	1	3	1.0308	1	1.0308	3	1.1763	1	1	1	1.4202	1	1
10	-	-	-	-	-	-	-	-	1	1.7094	2	1	1	1.4604	2	1
13	-	-	-	-	-	-	-	-	2	1.8579	1	1.0057	3	1.6887	1	1.1054

## VI. A PROTOCOL IMPLEMENTATION FOR cDSMA

We now sketch a distributed protocol implementation for cDSMA. Our implementation takes into account the constraints of self-organizing networks in that it draws on information locally available at network nodes and distributes the computational load and decision-making task among the nodes on the service migration path. In the same time, it keeps to a minimum information dissemination overhead through messages propagating in the network. We present the proposed implementation in a step-by-step fashion together with a rough analysis of time and message complexity:

*Service host advertisement:* Every time the service carries out a cDSMA-driven migration hop, the new host initiates a service advertisement phase (fig. 4.a) to inform all network nodes about the current service location. This task may be carried out by any efficient flooding scheme requiring  $O(|E|)$  messages and  $O(D)$  time, where  $D$  is the network diameter.

*Computation of nodes' wCBC metrics:* Our experiments with real-world ISP topologies suggest high rank correlation of centrality metrics (Section III) with their egocentric counterparts, as measured in the nodes' ego-networks<sup>6</sup> [27]. Table VI, for example, reports the correlation values between node rankings produced on the basis of the BC and ego-BC metrics, respectively, on these topologies. The high rank correlation suggests that the selection of the nodes with the highest centrality metrics can be made with sufficient accuracy according to their egocentric counterparts. Likewise, ego-wCBC can be computed locally (fig. 4.b) by individual nodes in  $O(d_{max}^2)$ , (where  $d_{max}$  is network's maximum node degree), assuming they are aware of their neighbors' service demands. These

<sup>6</sup>The ego-network structure of social studies comprises a person of interest referred to as the "ego", along with those having an affiliation with it, known as "alters". Alters may as well share relations with each other. The counterpart of the ego network in graph-theoretic terms (shaded in fig. 4.b) is called centered graph [26] and includes a given node plus its 1-hop neighbor nodes.

TABLE VI  
CORRELATION STUDY BETWEEN BC-EGCBC ON ISP TOPOLOGIES

DataSet	ISP (AS number)	Spearman Correlation Coefficient
36	Global Crossing (3549)	0.9648
35	-/-	0.9690
33	NTTC-Gin (2914)	0.9209
21	Sprint (1239)	0.9718
41	DFN-IPX-Win (680)	0.9662
46	TDC (3292)	0.9934
40	JanetUK (786)	0.9714

values along with the individual service demands are then communicated (in  $O(D)$  time) to the current service host (alternatively, demands could be directly measured by the current service host if the estimation delays can be tolerated). These dedicated  $O(|V|)$  ego-info reporting messages should include a field where all intermediate nodes on the shortest path to the service host will be recorded.

*Identification of top-wCBC nodes and mapping the residual demand on them:* The current service host collects the ego-info-reporting messages and identifies the top-wCBC nodes that form the  $G_{Host}^i$  (fig. 4.c). Moreover, the host can carry out the residual demand mapping task. For each outside node  $z$ , the host parses its report and adds its demand to the firstly encountered top-wCBC node on the path from  $z$  to itself. This demand mapping is clearly an approximation of the nominal algorithm process, as described in step 3 of section IV, in that the demand of each outside node is fully credited to only one node in  $G_{Host}^i$ , even if node  $z$  has more than one shortest paths to the host node going through other  $G_{Host}^i$  nodes. The demand mapping step for each outside node involves  $O(D)$  binary searches within the  $O(\alpha|V|)$ -size vector of  $G_{Host}^i$  nodes (shown in the subplot of fig. 4.c). Therefore, the mapping of the demands of all outside nodes takes  $O((1-\alpha)|V|D \log(\alpha|V|))$  steps.

*Solution of the 1-median problem within the  $G_{Host}^i$  sub-graph:* The service host notifies each of the the top-wCBC

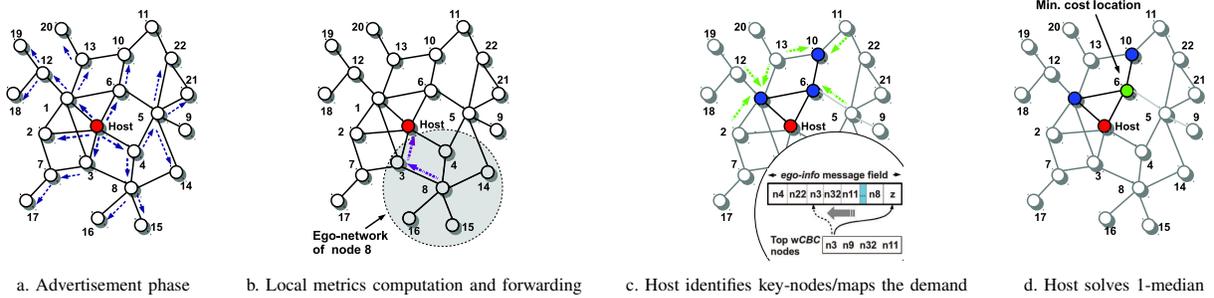


Fig. 4. cDSMA protocol implementation phases

nodes with unicast messages of which other nodes (co-players) are included in the  $G_{Host}^i$  and queries them for their pairwise distances. Each node determines its distance to the other co-players via a mechanism such as the ping utility ( $O(\alpha^2|V|^2)$  steps,  $O(\alpha^2|V|^2)$  messages), and communicates them (with  $O(\alpha|V|)$  messages) to the host. With the pairwise distances of  $G_{Host}^i$  nodes and the demand mapping terms at hand, the host is capable to solve the reduced 1-median problem and determine the next-best service host.

The centralized brute-force solution of the problem by a single super-node entity would require: a) collecting information about the global network topology and demand distribution; b) solving an instance of the all-pairs shortest paths problem, which requires  $O(|V|^3)$  time (e.g., using the Floyd-Warshall algorithm); c) exhaustively enumerating all  $|V|$  possible service locations and comparing the costs related to each one of them, involving  $O(|V|)$  multiplications, additions, and comparisons. Therefore, the overall complexity would be  $O(|V|^3)$ . Through the four cDSMA steps each host totally takes  $O(\alpha^2|V|^2)$  time to determine the optimal location in the respective subgraph (of size  $\alpha|V|$ ). The full migration process includes  $h_m(\alpha)$  such computations resulting in  $O(h_m(\alpha)\alpha^2|V|^2)$  time complexity, where  $a \ll 1$ . However, for typical network sizes the constants  $h_m(\alpha)$  and  $\alpha$  drastically affect the emerging cost; our experiments (Section V) suggest that the typically involved values (i.e.,  $\alpha = 3\%$  and  $h_m(\alpha) \approx 3$ ) lead to an apparently beneficial cost reduction of one or more orders of magnitude.

## VII. CONCLUSIONS

We have developed a scalable heuristic for placing service facilities in distributed self-organizing environments. We have introduced a CNA-driven metric ( $wCBC$ ) and an algorithm (cDSMA), which appropriately convert the original complex global optimization problem into a few smaller ones; their iterative solution directs the service migration towards cost-effective locations. Experiments on real-world topologies have shown that cDSMA provides excellent accuracy and scalability, even when it runs its iterations with less than a dozen nodes. Finally, we have outlined a realistic cDSMA protocol implementation with drastically reduced complexity compared to the typical centralized solution.

## REFERENCES

- [1] T. Plogemann *et al.*, "From content distribution networks to content networks," *Comput. Commun.*, vol. 29, no. 5, pp. 551–562, 2006.
- [2] V. Jacobson *et al.*, "Networking named content," in *5th ACM CoNEXT 2009*, Rome, Italy, December 2009, pp. 1–12.
- [3] A. Galis *et al.*, "Management architecture and systems for future internet networks," in *FIA Book : "Towards the Future Internet"*, May 2009.
- [4] E. Silva *et al.*, "Supporting dynamic service composition at runtime based on end-user requirements," in *Proc. 1st International Workshop on User-generated Services*, Stockholm, Sweden, November 2009.
- [5] Z. Zhao *et al.*, "A survey of user generated service," in *IEEE Network Infrastructure and Digital Content*, Beijing, China, November 2009.
- [6] Yahoo Pipes, <http://pipes.yahoo.com/pipes/>.
- [7] P. Mirchandani and R. Francis, *Discrete location theory*, Wiley, 1990.
- [8] D. B. Shmoys *et al.*, "Approximation algorithms for facility location problems," in *29th ACM STOC*, 1997, pp. 265–274.
- [9] K. Oikonomou and I. Stavrakakis, "Scalable service migration in automatic network environments," *IEEE JSAC*, vol. 28, no. 1, 2010.
- [10] G. Smaragdakis *et al.*, "Distributed Server Migration for Scalable Internet Service Deployment," *IEEE/ACM Trans. on Netw.*, (To Appear).
- [11] M. E. J. Newman, "The Structure and Function of Complex Networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.
- [12] E. M. Daly and M. Haahr, "Social network analysis for information flow in disconnected delay-tolerant manets," *IEEE Trans. Mob. Comput.*, vol. 8, no. 5, pp. 606–621, 2009.
- [13] O. Kariv and S. Hakimi, "An algorithmic approach to network location problems. II: The p-medians," *SIAM J. App. Maths.*, vol. 37, no. 3, 1979.
- [14] R. Solis-Oba, "Approximation algorithms for the k-median problem," ser. Springer Lecture Notes in Comp. Science, vol. 3484, 2006.
- [15] G. Wittenburg and J. Schiller, "A survey of current directions in service placement in mobile ad-hoc networks," in *IEEE PERCOM '08*, Hong Kong, March 17-21 2008, pp. 548–553.
- [16] T. Moscibroda and R. Wattenhofer, "Facility location: distributed approximation," in *PODC '05*, 2005, pp. 108–117.
- [17] K. Jain and V. V. Vazirani, "Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation," *J. of ACM*, vol. 48, no. 2, pp. 274–296, 2001.
- [18] S. Pandit and S. Pemmaraju, "Return of the primal-dual: distributed metric facility location," in *ACM PODC '09*, 2009, pp. 180–189.
- [19] P. Hui *et al.*, "Bubble rap: social-based forwarding in delay tolerant networks," in *ACM MobiHoc '08*, Hong Kong, China, 2008.
- [20] P. Pantazopoulos *et al.*, "Efficient social-aware content placement for opportunistic networks," in *IFIP/IEEE WONS*, Slovenia, 2010.
- [21] —, "Scalable distributed service migration via complex networks analysis," Tech. Rep. [Online]. Available: <http://cgi.di.uoa.gr/~ppantaz/>
- [22] U. Brandes, "A faster algorithm for betweenness centrality," *Journal of Mathematical Sociology*, vol. 25, pp. 163–177, 2001.
- [23] A. L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, Oct 1999.
- [24] J.-J. Pansiot *et al.*, "Extracting intra-domain topology from mrinfo probing," in *Proc. PAM*, April 2010.
- [25] R. M. Karp and R. E. Tarjan, "Linear expected-time algorithms for connectivity problems," in *ACM STOC '80*, Los Angeles, 1980.
- [26] L. C. Freeman, "Centered graphs and the structure of ego networks," *Mathematical Social Sciences*, no. 3, pp. 291–234, 1982.
- [27] M. Everett and S. P. Borgatti, "Ego network betweenness," *Social Networks*, vol. 27, no. 1, pp. 31–38, 2005.