Using Stochastic Approximation to Design OSPF Routing Areas that Satisfy Multiple and Diverse End-to-End Performance Requirements

Kyriakos Manousakis
Applied Research
Telcordia Technologies
Piscataway, NJ, 08901
kyriakos@research.telcordia.com

Anthony J. McAuley
Applied Research
Telcordia Technologies
Piscataway, NJ, 08901
mcauley@research.telcordia.com

Abstract— **Dividing an Open Shortest Path First (OSPF) Autonomous System (AS) into independent routing areas allows area topology abstraction, reducing route overhead, table size, and convergence time, while providing some isolation from bad routing data. On the contrary, areas reduce connectivity, while increasing configuration complexity, routing path length, and traffic concentration. The formation of performance efficient OSPF areas subject to multiple requirements is known to be NP-complete problem; however, some simple heuristics have been used to optimize for particular routing metrics. For example, a min-cut can be used to ensure balanced number of nodes per area. However, no existing tools can optimize for actual end-to-end performance requirements or take into account the characteristics of network topology. This paper describes a fast and flexible optimization tool that automates the design of Open Shortest Path First (OSPF) routing areas to meet heterogeneous end-to-end performance requirements. The tool is based on an enhanced version of Simulated Annealing (SA) algorithm, which is a general stochastic approximation method capable of handling multiple, diverse and conflicting requirements (multi-objective optimization). The Simulated Annealing based tool can provide from highly optimized solutions for network planners designing conventional wired OSPF networks with known application flows to scalability and robust solutions in wireless networks using MANET OSPF extensions with changing application needs. This paper formulates the OSPF areas design as a weighted-sum multi-objective optimization of routing metrics to maximize user capacity, while minimizing the increased delay and lost connectivity. For diverse topologies, we show significantly reduced user delay (over 25%) and increased available bandwidth (by over 400%). Further, we show that by simply adjusting the weights, the tool can prioritize the performance requirements and adapt to heterogeneous network environments.**

Keywords- OSPF routing areas; scalability; manageability; optimization; system capacity; end-to-end performance

I. INTRODUCTION*

Any routing protocol that requires routers to know about every destination or flood the network for each new flow becomes infeasible as the network dynamics grow and the number of nodes or sessions increases [1]. Although there have been approaches to reduce this overhead (e.g., by restricting who gets updates [2]), routing hierarchy is typically needed to provide scalability and manageability.

Hierarchy works by limiting the scope of topology changes and abstracting the topology information in routing updates. A single address/label prefix, for example, can identify all nodes in a given level of the hierarchy. Thus hierarchy both reduces route update traffic and average routing table size. Furthermore, routing hierarchy provides some isolation from bad routing data, and allows faster healing of faults.

On the negative side, hierarchy can significantly increase complexity (e.g., configuring border routers), average path length (e.g., for inter-area routes), and traffic concentration (e.g., in the backbone). Moreover, hierarchy can reduce inter-area connectivity.

Despite their potential drawbacks, hierarchies are widely used. The Autonomous Systems (ASes) of the internet use BGP [16] for inter-domain routing among ASes, while using independent intra-domain routing protocols. Today, many ASes use Open Shortest Path First (OSPF) [3] [4] [5] for intra-domain routing. However, as the ASes themselves become larger and in some cases more heterogeneous (e.g., [6] [6] [8] [9]), there is an increasing need of hierarchy even within the ASes. In particular, ASes are exploiting OSPF areas to provide greater scalability.

In most conventional networks routing hierarchy design relies on manually analyzing the network topology to determine effective area boundaries and locations of border routers where summarization should be applied. Optimization, however, can provide better solutions. In a MANET environment or wherever nodes must be rapidly deployed, automated techniques become more essential. While there exist area optimization techniques for both wired and wireless networks (e.g., [20] [21] [22] [23] [24] [25] [26] [27]), none is flexible enough to be applied across all kinds of OSPF networks or meet specific end-to-end performance requirements. This paper therefore proposes a new approach

* Prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance (CTA) Program, Cooperative Agreement DAAD19-2-01-0011. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon.

WIOPT 2008, 1st – 3rd Apr 2008, Berlin, Germany.
 ISBN 978-963-9799-18-9
DOI 10.4108/ICST.WIOPT2008.3197

that can provide both: a) more optimized solutions for network planners designing conventional wired OSPF networks (analogous to traffic engineering of routing weights [10] [11]); yet also provide b) efficient and robust solutions in wireless networks using MANET OSPF extensions [6] with dynamically changing end-to-end performance requirements.

Section II gives examples of possible OSPF areas' design requirements that we mathematically formalize into constraints and objectives. Section III describes our approaches to OSPF areas design optimization. Section IV shows how the routing objectives and constraints of Section II can be synthesized to represent multiple, diverse and conflicting end-to-end performance requirements. Section V presents some indicative results from the application of the proposed approach on some sample multi-objective OSPF areas design problems.

II. OSPF Area Design Objective and Constraints

This section briefly overviews OSPF [3], then provides some indicative examples of OSPF routing areas design objectives.

A. *OSPF Overview*

An OSPF router has one or more instances of OSPF running with interfaces assigned to it. Each OSPF instance assigns its interface(s) an IP address, an associated network mask, and a link weight. OSPF for IPv4 [4] and IPv6 [5] use the same fundamental mechanisms (e.g., flooding, Designated Router election, Shortest Path First calculations, and support for routing areas). The parts of the protocols that principally affect end-to-end performance are the:

- *Hello packets (Message Type 1)*: They are used to discover and maintain neighbor relationships. Bidirectional communication is indicated when the router discovers itself listed in the neighbor's Hello Packet. On broadcast and NBMA networks the Hello Protocol also elects a Designated Router to represent the network (which reduces the amount of Link State Update messages).
- *Link State Updates that exchange topology information*: OSPF can get the whole database from its neighbors (Message Type 2 and 3). However, after a restart, most link state information is gained incrementally (Message Type 4 and 5), when a node floods its information about its links throughout the AS (or area). These Link-State Advertisement (LSA) messages are acknowledged. External routing information is also flooded unaltered in LSAs throughout the AS.

OSPF uses a Shortest Path First algorithm to calculate a path to each destination with least total weight.

B. *OSPF Areas*

OSPF areas provide a two level hierarchy consisted of the OSPF backbone area ("Area 0") and the non-backbone areas (e.g., [12] [13]). The latter areas utilize the backbone area for the distribution and exchange of routing information.

Figure 2 shows how a simple flat topology shown in Figure 1 can be divided into ASes (AS 101 and AS 102), with each AS divided into OSPF areas (e.g., AS 101 has 4 areas). A hierarchical OSPF network has three main types of routers:

1. Internal Routers (e.g., R-4 in Figure 2) run a single copy of the basic routing algorithm.
2. AS Boundary Router (e.g., BRa-1 in Figure 2) advertises AS external routing information throughout the Autonomous System (e.g., from BGP).
3. Area Border Routers (e.g., BRb-7 in Figure 2) run multiple copies of the basic OSPF link-state algorithm (and separate link-state databases) for each area it is connected to. Area border routers condense the topological information: a) of their attached areas for distribution to other areas, b) its cost to all networks external to the area to its internal routers. In addition to the BR to Area 0, OSPF allows non-backbone BRs (e.g., BRr-17 in Figure 2) for direct communication among leaf areas (though they are not allowed to advertise routes outside its immediately connected areas).

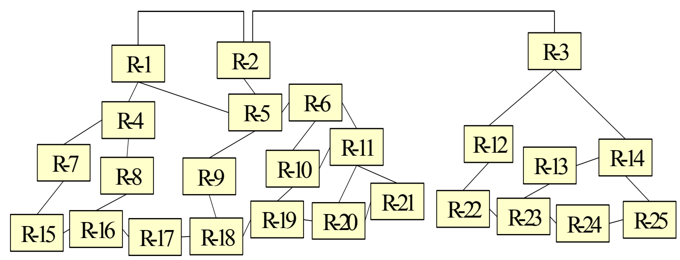

Figure 1. Example Flat Network with 25 OSPF routers, where LSAs are broadcast to all routers

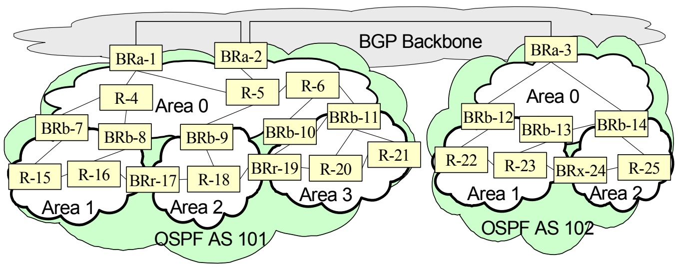

Figure 2. Division of 25 OSPF routing into a routing hierarchy, where most LSAs are broadcast only within their small OSPF areas.

There are different kinds of non-backbone areas depending on the amount of aggregation at border routers [12] [13]. The BR aggregation is a key tool in balancing the tradeoff among routing overhead, table size, convergence time and routing sub-optimality [10] [14] [11]. A BR can make the intra-area topology invisible from outside, thus greatly reducing the number and size of routing updates (as compared to treating the entire AS as a single domain) [14]. Also, routing within the area (intra-area routing) is determined only by the area's own topology, lending the area protection from bad routing data.

C. OSPF Areas Design: Constraints and Objectives

Generally, any network design requirement can be formulated as a set of objective functions (to minimize or maximize) and constraints (bounds). Specifically, for the design of OSPF areas, indicative constraints include:

- **Intra-Area Connectivity**: To isolate areas, OSPF requires a node to reach all other members of its area without passing outside the area. Thus, each OSPF area A_i, including Area 0 (A_0), must be at least 1-connected graph: i.e.:

  Path P from $node_i \in A_k$ to $node_j \in A_k$ with no $node_n \in$ P and $node_n \notin A_k$ (1)

- **Inter-Area Connectivity**: OSPF requires there is a single backbone Area 0 (A_0) that is responsible for distributing routing information between non-backbone areas (A_i); thus, all Areas must be directly attached to A_0:

  There is at least one $node_i \in A_k$, which is direct neighbor of at least one $node_j \in A_0$ (2)

Once the constraints are satisfied, we want to maximize the hierarchy benefits while minimizing its drawbacks based on some known metrics, such as:

n: Total number of nodes in the network.

b: Desired number of backbone links per area. (e.g., to provide some redundancy and load balancing, without overloading the backbone, we may set *b=2*).

g: Desired number of nodes in an Area (e.g., for larger networks we may set $g = \sqrt{n}$ for reducing routing overhead). Thus the number of areas, $q = \lceil n/g \rceil$

Table I lists indicative objectives (right column) for different routing metrics (left column). Due to the flexibility of the approach, the metrics listed on Table I are only indicative ones. We are not limited on those metrics but many other metrics are possible, such as those based on expected Link Expiration Time and relative node velocity that are not listed here.

III. Optimization Approach

Table I provides some indicative and core metrics for the formulation of OSPF areas design requirements. Upon their formulation, an algorithmic framework has to utilize them for the formation of the corresponding areas. This section describes our flexible approach to OSPF areas optimization among other, less flexible approaches that have been used for the formation of network hierarchies.

A. Alternative Optimization Architectures

We consider three basic optimization alternatives:

- **Distributed optimization**. Each node decides which part of the hierarchy it is in based on local information. The most popular distributed approach is to construct a Connected Dominating Set (CDS) within the network [20] [21] [22] [23] [24]. Each node begins by randomly electing itself as "cluster-head" unless it first hears from another cluster-head. In some cases, certain nodes may have some bias towards certain nodes [22] [21] (e.g. those of or highest degree). Eventually each non-clusterhead is at most *d* hops (typically 1 hop) from a clusterhead and all the clusterheads form a dominating set. The CDS based techniques have been applied to the specifics of OSPF [8], where the members of the CDS become the backbone (OSPF Area 0 nodes). The distributed approach is simple and robust, making it ideal for the dynamic environments for which it was designed; however, it generally considers only a very small fixed subset of the possible and useful set of routing objectives and constraints that can be exploited in the design of OSPF areas. Specifically, the CDS based techniques are customized to operate only within the pre-specified subset of objectives and constraints and they do not guarantee the optimality or even "goodness" of the OSPF areas formed subject to the imposed objectives and constraints.

TABLE I EXAMPLE OSPF ROUTING OBJECTIVES

OSPF Area Metric		Example Objective
$B_{i,j,0}$	Link Bandwidth between $node_i \in A_0$ and $node_j \in A_0$	$\max \sum_{i=0}^{N_0-2} \sum_{j=i+1}^{N_0-1} B_{i,j,0}$
d_l	Delay on the active link l of the network	$\min \sum_{l=1}^{P_k^{OSPF}} d_l$
$H_{i,j,k}$	Hops from $node_i \in A_k$ to closest $node_j \in A_0$	$\min \sum_{k=1}^{q-1} \sum_{i=0}^{N_k-1} H_{i,j,k}$
$M_{i,j,0}$	Number of node/link disjoint paths from $node_i \in A_0$ to $node_j \in A_0$	$\max\left(\arg\min_{\forall i,j \in A_0}\left[M_{i,j,0}\right]\right)$
N_i	Number of nodes in A_i	$\min \sum_{i=0}^{q-1} (N_i - g)^2$
$P_{i,j}$	Number of inter-area links between A_i and A_j	$\min \sum_{i=1}^{q-2} \sum_{j=i+1}^{q-1} P_{i,j}$
$P_{i,0}$	Number of inter-area links between A_i and A_0	$\min \sum_{k=1}^{q-1} \lvert P_{i,0} - b \rvert$
$R_{i,j}$	Routing stretch (hops) between $node_i$ and $node_j$	$\min \sum_{i=0}^{n-2} \sum_{j=1}^{n-1} R_{i,j}$
$S_{i,j,k}$	Number of (flat) shortest paths using k'th link between A_i and A_j	$\min \sum_{i=1}^{q-2} \sum_{j=i+1}^{q-1} \sum_{k=1}^{P_{i,j}} S_{i,j,k}$
$T_{i,j,k}$	Total traffic on k'th link between A_i and A_j	$\min \sum_{i=1}^{q-2} \sum_{j=i+1}^{q-1} \sum_{k=1}^{P_{i,j}} T_{i,j,k}$
W_i	Network width of A_i: (Area Diameter in hops)	$\min \sum_{i=0}^{q-1} (W_i)$

- **Sequential Multi-objective Optimization**. A network can be optimized for multiple constraint or objectives one at a time. An example of centralized sequential optimization is proposed in [25], where a fast min-cut algorithm divides a topology so the number of non-backbone inter-area links ($P_{i,j}$ in Table I) is minimized, while roughly balancing the number of nodes in each non-backbone area (N_i in Table I). Next, if any areas are not connected (constraint (1)), a component merging heuristic is used. If the heuristic fails, they relax the balancing constraints and redo the min-cut. The third step is to construct the backbone Area (N_0 in Table I), so it touches all areas (constraint in Equation 2). Finally, additional objectives can be applied for satisfying additional requirements (e.g., $M_{i,j,0}$ from Table I). By using multiple objectives sequential optimization can produce much better results than the simple distributed implementations, but the sequential optimization can produce highly non-optimal results. Optimizing for one function at a time can severely limit the possible solutions space and can even prevent finding a feasible solution (satisfying the constraints) in some networks. Figure 3 shows how the initial cut at a solution (which is optimal for the first objective), might eliminate the possibility to converge to the overall optimal solution, due to the narrower view of the solution space.

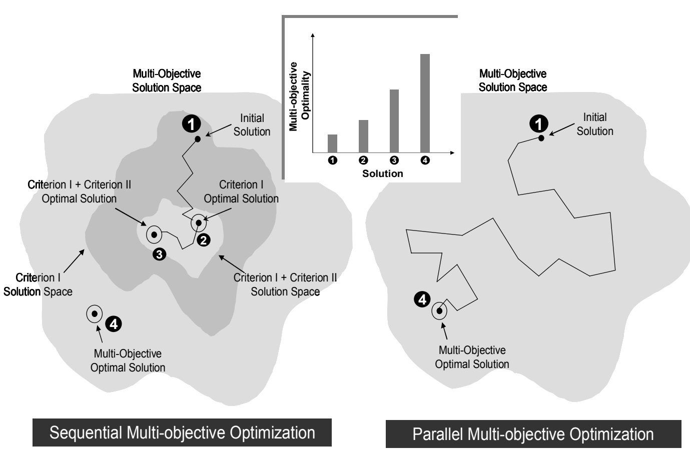

Figure 3. Visualization of the broader search space used in parallel optimization compared to sequential optimization.

- **Weighted-Sum Multi-Objective Optimization** There are many ways to combine the constraints and different objectives in parallel. One popular method is to use a weighted-sum of the y objectives (O_i), with the importance of each objective set by the weight (w_i) applied to the objective:

$$\min \sum_{i=0}^{y-1} \left(w_i . O_i \right) \tag{3}$$

where,
$$\sum_{i=0}^{y-1} w_i = 1 \tag{4}$$

In [26], for example, a weighted sum was used to simultaneously create balanced sized clusters (represented by N_i in Table I) and minimum routing stretch (represented by $R_{i,j}$ in Table I), while ensuring that intra-cluster connectivity constraint (1) is always satisfied. By simultaneously using all the information about the network, parallel optimization can produce much better results than distributed implementations. The challenge is to find efficient and robust means of solving the optimization.

B. *Multi-objective Optimization Realization*

To satisfy multiple, diverse and dynamically evolving performance requirements using parallel multi-objective optimization, we introduce a flexible optimization tool that consists of a mathematical framework responsible for the formalization of the performance requirements into objective functions and of an algorithmic framework responsible for the optimization of the corresponding objective functions. The algorithmic framework consists of two parts:

- **General stochastic approximation**. The general stochastic approximation class of algorithms includes Simulated Annealing [26], Genetic, and Kernighan-Lin graph partitioning algorithms [27]. Given its proven effectiveness we chose Simulated Annealing (SA). By searching randomly the solution space, SA can provably converge to the optimal solution at the limit. SA probabilistically (e.g. using the Metropolis criterion) accepts worse solutions to avoid local minima (maxima), which might result in the formation of low "quality" areas subject to the imposed performance requirements. SA starts from a large value of its control parameter c_0, such that almost every move gets accepted. The value of the control parameter is cooled down (decreases) with respect to a cooling schedule. In each iteration a new solution C' is generated with a small perturbation on the currently optimal one C_t^*. The difference in the solutions' costs is $\Delta E = E^* - E'$, where E^* is the cost of the currently optimal solution and E' is the cost of the new generated solution, which both are computed based on the objective function being optimized. The new solution C' is accepted ($C_{t+1}^* \leftarrow C'$) with respect to Metropolis criterion:

$$P_{c_{t+1}}\left(C_{t+1}^* \leftarrow C'\right) = \begin{cases} 1 & \text{if } \Delta E > 0 \\ \exp\left(\dfrac{\Delta E}{c_{t+1}}\right) & \text{if } \Delta E \le 0 \end{cases}$$

SA algorithm terminates when the termination condition is satisfied. The weakness of original SA is its slow convergence time. However, as we have shown in [24] by adjusting the SA characteristics (e.g., enhanced SA), a small loss in optimality can be traded for orders of magnitude improvements in convergence times. For example, some indicative convergence times for enhanced SA based multi-objective hierarchy optimization are: a) 1ms for 100 nodes, b) 20secs for 1000 nodes networks.

- **Distributed heuristics**. In the presence of network dynamics the continuous application of general stochastic approximation algorithm would require increased amounts of information transfer in small window of time. To remedy this weakness, some distributed maintenance must be performed. For this we need a distributed approach that uses the same constraints and objectives as the general stochastic approximation, but with only local information. For this we designed the Active Maintenance approach in [28]. Here Area Border Routers (ABRs) at randomly scheduled intervals, perform local optimization by locally selecting the neighboring domains to join subject to the imposed performance requirements. These local reconstructive decisions provably converge, for specific class of objective functions to the global optimal.

IV. End-to-End Performance Optimization

This section describes how we combine the multiple hierarchy objectives (from Table I) with a weighted sum that directly represents multiple and diverse end-to-end performance requirements for network connectivity, robustness, capacity and delay.

A. *Using End-to-End Performance Requirements*

The right column of Table I shows some indicative examples of mathematically formulated requirements into minimization or maximization optimization problems using the OSPF routing metrics. Unfortunately, although optimization of these metrics is desirable, the importance of each of these metrics to the end-to-end performance requirements (e.g., end-to-end delay), is not straightforward. Also, as many of these requirements are conflicting, it is always complex to know how to make the best selection or tradeoff among them.

We propose to justify the assignment of nodes into OSPF areas and the selection of Area Border Routers in terms of the real network performance and cost objectives. Thus, for example, what the user cares about is not minimizing the number of hops to the border router or the routing overhead, but the effect that these metrics have on the network connectivity, capacity and end-to-end delay.

B. *Primary Impacts of Routing Metrics on End-to-End Performance*

We found that the routing metrics (from Table I) can be classified into one of three primary performance characteristic classes. These classes are *connectivity/robustness*, *capacity* and *delay*.

Those metrics that primarily affect *connectivity* and *robustness* include:

- $\mathbf{M}_{i,j,0}$: Most paths go through the backbone, thus having maximizing *k*-connectivity the network connectivity and robustness is expected to be significantly improved.
- $P_{i,j}$: Creating OSPF areas eliminates inter-area paths between nodes that use a non-backbone area as a transit network. Thus, minimizing the number of links cut by area divisions can improve connectivity and robustness.
- $P_{i,0}$: Increasing the number of links in the backbone area (Area 0) improves connectivity and robustness.

Those metrics that primarily affect *capacity* include:

- $B_{i,j,0}$: OSPF hierarchy eliminates paths that transit non-backbone areas, forcing more traffic into the backbone (A_0). LSA overhead in large networks is also roughly two orders of magnitude larger than any other area [15]. Thus to improve network capacity, we want to maximize the available backbone bandwidth
- N_i and q: As a first order approximation, we can estimate the OSPF routing overhead, which reduces the available network capacity that will grow as $\Theta(n^2)$ in a flat network, where n is the number of nodes. However, with a two level area hierarchy with $N_i=n^{0.5}$ nodes per area and $q=n^{0.5}$ areas, the routing overhead only grows as $\Theta(n^{1.5})$. Thus to improve network capacity, we can form $n^{0.5}$ balanced sized areas.
- $S_{i,j,k}$, $T_{i,j,k}$: The OSPF hierarchy forces more traffic load into the backbone (A_0). Even paths between directly connected areas will typically not travel by the optimal paths but through the backbone area. For improving the optimality of the routing paths and subsequently the network capacity, we want to minimize the amount of shortest paths or traffic that is cut by non-backbone areas division.

Those metrics that primarily affect *delay* include:

- $H_{i,j,k}$: OSPF routing stretch is proportional to the average distance of nodes from their area's corresponding Border Router(s). As OSPF directs most of the inter-area traffic through the Area-0 Border Routers, it is particularly desirable to minimize the distance of non-backbone areas' nodes from any node in A_0:
- W_i: Minimizing the number of hops across areas reduces the latency of routing updates. The less hops can also reduce the routing stretch among areas and this can subsequently benefit the end-to-end delay.
- d_l: The delay on each active link of the end-to-end path selected by OSPF affects the end-to-end delay. The selection of the end-to-end path and the corresponding end-to-end delay are significantly affected by the design of OSPF areas.

We also note that the objectives are not independent. For example, increasing the number of backbone links ($P_{i,0}$) directly improves connectivity and robustness, but will also increase the backbone routing overhead (by increasing N_0) and reduce latency (by reducing $H_{i,j,k}$).

C. Weighted Sum Objectives for OSPF

Rather than having a fixed objective, we formulated a weighted sum of objectives with flexible weights that depend on:

- The relative importance of connectivity (w_c), capacity or rate (w_r) and delay (w_d).
- The network environment (e.g., network size, node density, dynamics). For example, in larger more dynamic networks the importance of routing overhead (and the parameters N_i and q) on overall capacity can be highlighted.

The general form of the weighted sum objectives is given by the equation below:

$$w_c\left(w_{c1}f_m(M_{i,j,0})+w_{c2}f_p(P_{i,j})+w_{c3}f_q(P_{i,0})\right)+$$
$$w_d\left(w_{d1}f_h(H_{i,j,k})+w_{d2}f_w(W_i)\right)+$$
$$w_r\left(w_{r1}f_b(B_{i,j,0})+w_{r2}f_n(N_i,q)+w_{r3}f_s(S_{i,j,k},T_{i,j,k})\right)$$

where,

$$w_c+w_d+w_r=1$$
$$w_{c1}+w_{c2}+w_{c3}=1$$
$$w_{d1}+w_{d2}=1$$
$$w_{r1}+w_{r2}+w_{r3}=1$$

The values of w_c, w_d, and w_r reflect the relative importance of network connectivity, delay and capacity, while the value of w_{c1}, w_{c2}, w_{c3}, w_{d1}, w_{d2}, w_{r1}, w_{r2}, and w_{r3} reflect the importance of the specific OSPF areas' characteristics In order to simplify the optimization, some weights may be set to 0 when their relative importance is low for the corresponding design.

D. Example Objectives for OSPF Areas Design

We have designed many objective functions that emphasize various diverse end-to-end performance requirements. Representative examples are described by the following objective functions:

- **Maximizing Capacity when routing overhead is the most important factor.** If we assume that we have a limited bandwidth dynamic network, where routing overhead can be significant factor to the total available bandwidth, then we require an OSPF configuration that minimizes routing overhead by creating balanced size areas. Our objective function that corresponds to this requirement (see Table I) is formalized as:

$$J(A_{OSPF})=\min_{A_{OSPF}}\sum_{i=1}^{q}\left(N_i-g_i^*\right)^2 \tag{5}$$

where,

A_{OSPF} : OSPF areas configuration
q : number of OSPF areas
N_i : number of nodes per area
g_i^* optimal size per OSPF area for reducing routing overhead

- **Maximizing Capacity when path stretch is the most important factor**. If we assume that we have a guaranteed high bandwidth network, where routing overhead represents a small fraction of the total bandwidth, then an important factor affecting capacity can be routing stretch (increased path lengths due to the hierarchal routing). The stretch requires more network resource usage, reducing overall network capacity. Our objective function that formalizes the minimization of routing stretch requirement is:

$$J(A_{OSPF})=\min_{A_{OSPF}}\sum_{s=1}^{P}R_s^{OSPF} \tag{6}$$

where:

A_{OSPF} : OSPF areas configuration
P : set of source-destination pairs
R_s^{OSPF} : OSPF routing path length (stretch) for source destination pair s

- **Maximizing Capacity when both path stretch and routing overhead are important.** The multi-objective function and the corresponding weights that best combine routing overhead and stretch for improving the capacity of network is:

$$J(A_{OSPF})=\min_{A_{OSPF}}\left[0.1*\sum_{i=1}^{q}\left(N_i-g_i^*\right)^2+0.9*\sum_{s=1}^{P}R_s^{OSPF}\right] \tag{7}$$

- **Maximizing Capacity and Minimizing Delay**. In this indicative scenario we combine factors related to overhead (e.g. we assume again that the main overhead source is the routing overhead) and added delay due to the hierarchal routing since it is not using the lowest delay paths through the network. The optimal weights have been obtained experimentally and are provided in the following objective function that formalizes the combined capacity and delay requirements:

$$J(A_{OSPF})=\min_{A_{OSPF}}\left[2\times10^{-2}*\sum_{i=1}^{q}\left(N_i-g_i^*\right)^2+0.98*\sum_{s=1}^{P}\sum_{l=1}^{P_s^{OSPF}}d_l\right] \tag{8}$$

where,

P : set of source-destination pairs
P_s^{OSPF} : active links on the end-to-end OSPF path for source-destination pair s
d_l : Delay on active link l

V. Results

This section presents quantitative results from the application of our enhanced optimization framework on the design of OSPF areas in various networks subject to diverse end-to-end performance objectives. Specifically, we provide results, related to the ability of the proposed mechanism and objective functions to form OSPF routing areas that satisfy multiple and diverse end-to-end performance requirements.

A. Experimental Setup

The experimental set up of this simulation analysis is based on the generation of 100 diverse networks (e.g. different node densities) of 100 nodes each. Each active link l_z, which is defined by the nodes i_z, j_z of network z is characterized by a delay $d_{l_z}^{i_z,j_z}$, which follows the uniform distribution $d_{l_z}^{i_z,j_z} \sim U(10ms)$. This characterization virtually injects link diversity into the network and does not affect the generality of the conclusions drawn.

B. Improving Network Capacity and Delay

The results shown on Figure 4, represent the average improvement on end-to-end delay when different number of OSPF areas (e.g. 3 to 8 areas, where the backbone Area-0 is also included) are designed subject to optimizing only for capacity ((Objective 1),equation (5)) or for both capacity and delay ((Objective 1+Objective 2), equation (8)).

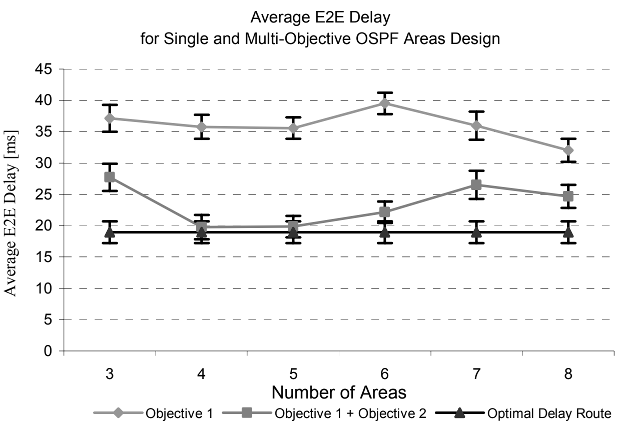

Figure 4. Multi-objective optimization effect on end-to-end delay performance

In each simulation scenario, the delay has been computed based on the end-to-end path obtained by emulating OSPF routing protocol on the designed areas for a set of ten randomly selected source destination pairs. Each bar per data point corresponds to confidence interval of 95%.

Figure 4 shows the importance of the multi-objective optimization on the design of OSPF areas. The end-to-end delay improvement with multi-objective optimization is significant compared to the end-to-end delay measurements in the case of single objective optimization (e.g. routing overhead), where the delay was not considered. Furthermore, after the multi-objective optimization of OSPF areas the end-to-end delay is very close to the optimal one, which has been computed by applying Dijkstra's shortest path algorithm on the weighted (e.g. link delay) graphs that represent the flat network topologies. We have to highlight also the fact that the routing overhead remains low and undisturbed, even though we have added the extra requirement on end-to-end delay performance.

An interesting observation from the analysis of results, is that the average end-to-end delay improvement is lower for small (e.g. 3 areas) or large (e.g. 7 and 8) number of designed areas compared to the intermediate cases, where 5, 6 and 7 number of areas have been formed. Even though the effect is the same the explanation is different for the lower and higher number of formed areas. Specifically, for the lower number of formed areas the effect of single objective is less significant due to small number of OSPF areas, so the end-to-end delay suboptimality is not exploited much. The latter is because the suboptimality mainly arises due to inter-area routing, since the intra-area routing is supposed to be optimal. In the case of larger number of formed areas the suboptimality is due to the large solution space. The enhanced optimization algorithm, which has been configured towards fast convergence rather than optimality, is unable to sufficiently explore the solution space in order to obtain an optimal solution. The latter can be fixed by allowing the optimization algorithm to run for more time (e.g. configure it for optimality), instead of configuring it towards fast convergence.

As expected, when the areas formation objectives simultaneously emphasize routing overhead reduction and end-to-end delay performance, the OSPF paths are not established only with lower overhead but they are also characterized by smaller end-to-end delay. On the contrary, when the objectives for the formation of domains do not consider the end-to-end delay, the routing overhead may be low but the paths are characterized by large end-to-end delay, which may affect considerably the performance of many delay sensitive (e.g., real-time) applications.

C. OSPF Areas Design: Impact on Capacity

Previously, the selected design objectives aimed on the design of areas that reduce both routing overhead and end-to-end delay. In this section we present the ability of the OSPF areas design mechanism to improve multiple factors that affect capacity. Specifically, we compare the available capacity of the network, when OSPF areas are designed merely for reducing the path length suboptimality caused by hierarchical routing (6) and when the areas are formed so that both OSPF routing path length suboptimality and routing overhead (7) are simultaneously reduced.

Having formed OSPF areas under both simulation analysis scenarios, we measured the capacity consumed for signaling by OSPF protocol. Based on the latter measurements and taking into consideration (9) we computed the available capacity remained on the network for data delivery.

Definition (Available Capacity): If the flat capacity of the network G is C_G and the bandwidth consumed by routing overhead and routing stretch is O_r^G, then the available network capacity C_G^a is evaluated as:

$$C_G^a = C_G - O_r^G \qquad (9)$$

In the specific experiments we assume that the networks are characterized by limited resources (e.g. their capacity is 10^4

Kbps), so that routing overhead has substantial impact on their available resources and performance.

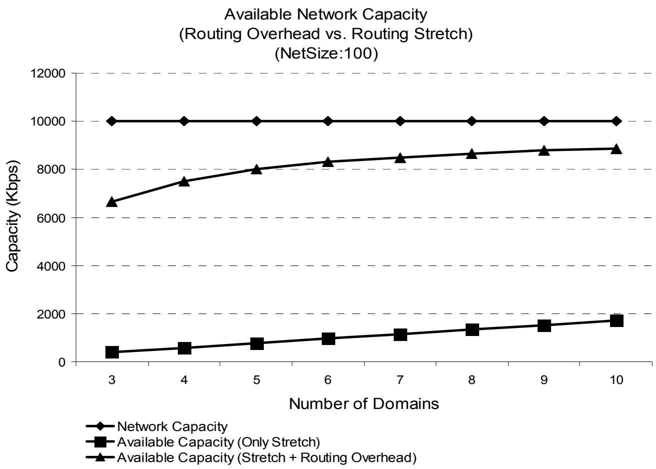

Figure 5. Multi-objective optimization effect on available network capacity

Figure 5 shows the dramatic impact of routing overhead on capacity. The top line of figure 5 reflects the end-to-end network capacity (C_G) before taking into account the routing overhead. Using only on the routing path length suboptimality given by (6), the available capacity is low (e.g., only 20% of the network capacity, since 80% is consumed by routing overhead). In contrast, when the OSPF areas are formed subject to both objectives (7), routing overhead has been reduced significantly and utilizes only 20% of network capacity, allowing the rest 80% available for data delivery.

During the simulation we observed that when the formation of OSPF areas is subject only to routing path suboptimality, the tool assigns a large number of nodes to the backbone (Area-0). Intuitively, this is expected since all inter-area traffic utilizes the backbone resulting in suboptimal paths, as opposed to the intra-area routing that has been assumed optimal with respect to path length. The membership size imbalance of the formed areas becomes the main source of the significant and overwhelming routing overhead.

VI. CONCLUSION

This paper presents the significance of using multiple and diverse end-to-end performance objectives rather than routing objectives for the design of areas (e.g. clusters). Although, the proposed stochastic approximation based mechanism (Simulated Annealing) is independent of the imposed areas design requirements, we selected two indicative sets of multiple objectives:

- Design of OSPF areas, which simultaneously improve capacity and end-to-end delay.
- Design of OSPF areas, which minimize multiple sources that limit the available network capacity. Specifically, end-to-end path suboptimality (stretch) and routing overhead.

By weighting appropriately the routing objectives in accordance to the network environment and their importance on the end-to-end performance, we show that the network performance improves significantly, compared to the scenarios, where the areas formation is unaware of the performance objectives of interest. Specifically the results show that for multiple different topologies, we could significantly reduce user delay (over 25%) and increases available bandwidth (by over 400%).

More generally, we identified the importance of the routing metrics on the formation of hierarchical structures to meet capacity, delay and connectivity requirements. This work also emphasized the importance of parallel multi-objective approaches such as Simulated Annealing, compared to the inherent weakness of sequential multi-objective mechanisms for simultaneously satisfying and trading off multiple end-to-end performance objectives. Even though the latter mechanisms are more realizable in dynamic network environments, they cannot provide the flexibility and quality of solutions obtained by the parallel multi-objective optimization approaches.

While the results indicate the greater flexibility and optimality of the new optimization framework over existing approaches, additional research is required to further improve various aspects of end-to-end network performance. Specifically, we need to better understand the importance of different metrics, how they impact each other and how this knowledge will lead to better multi-objective optimization equations. Also, research is needed to understand how to set the weights on the various objectives subject to the type of network environment and the relative importance of the imposed requirements.

REFERENCES

[1] L. Kleinrock, F. Kamoun, “Hierarchical Routing for Large Networks: Performance Evaluation and Optimization,” Computer Networks, Vol. 1, pp. 155-174, January 1977.

[2] C. Santiváñez, R. Ramanathan, I. Stavrakakis, “Making Link-State Routing Scale for Ad Hoc Networks,” MOBIHOC 2001

[3] J. T. Moy, “OSPF: Anatomy of an Internet Routing Protocol,” Reading, MA: Addison-Wesley, 1998.

[4] J. Moy, “OSPF Version 2,” *IETF RFC*, No. 2328, April 1998.

[5] R. Coltun, D. Ferguson, J. Moy, OSPF for IPv6,” Requests for Comments: 2740, December 1999.

[6] T. Henderson, P. Spagnolo, and J. Kim, “A Wireless Interface Type for OSPF,” Proceedings of IEEE MILCOM, pp. 1256-1261, Oct. 2003.

[7] P. Spagnolo, T. Henderson, “Comparison of Proposed OSPF MANET Extensions,” Proceedings of IEEE MILCOM, Oct. 2006.

[8] R. Ogier, P. Spagnolo, “MANET Extension of OSPF using CDS Flooding,” <draft-ogier-manet-ospf-extension-08.txt>, October 2006

[9] G. Apostolopoulos, R. Guerin and S. Kamat, “Implementation and performance measurements of QoS routing extensions to OSPF,” Proc. IEEE Infocom, New York, NY, March 21-25, 1999.

[10] B. Fortz and M. Thorup, “Internet traffic engineering by optimizing OSPF weights,” in Proc. IEEE INFOCOM, Tel Aviv, Israel, pp. 519–528, March 2000.

[11] T. Ye, H. T. Kaur, S. Kalyanaraman, K. S. Vastola and S. Yadav, “Dynamic optimization of OSPF weights using online simulation,” proceedings of IEEE INFOCOM, 2002.

[12] Zinin, Lindem, D. Yeung, “Alternative Implementations of OSPF Area Border Routers,” Request for Comments: 3509, April 2003

[13] P. Murphy, “The OSPF Not-So-Stubby Area (NSSA) Option,” Request for Comments: 3101 (Obsoletes: 1587), January 2003.
[14] R. Rastogi, Y. Breitbart, M. Garofalakis, and A. Kumar “Optimal Configuration for OSPF Aggregates”, IEEE/ACM Transactions on Networking, Vol. 11, No. 2, April 2003.
[15] S. Isett, A. Greenberg, M. Roughan and J. Gottlieb, “A case study of OSPF behavior in a large enterprise network” Proc. of the ACM Internet Measurement Workshop, Marseille, 2002.
[16] Y. Rekhter, T. Li, “A Border Gateway Protocol 4 (BGP-4),” RFC 1771, March 1995.
[17] M. Garey, D. Johnson, and L. Stockmeyer, “Some simplified NP-complete graph problems,” Theoretical Computer Science, 1, pp. 237-267, 1976
[18] R. K. Ahuja, T. L. Magnanti and J. B. Orlin, “Network Flows: Theory, Algorithms and Applications,” Prentice Hall, Englewood Cliffs, NJ, USA, 1993.
[19] Y. Collette, P. Siarry, “Multiobjective Optimization,” Publisher: Springer, ISBN 3540401822, 2004.
[20] V. Ramamoorthy, A. Bhide, and J. Srivastava, “Reliable Clustering Techniques for Large, Mobile Packet Radio Networks,” Proc. IEEE INFOCOM ’87, pp. 218–226, 1987.
[21] S. Basagni, “Distributed and Mobility-Adaptive Clustering for Multimedia Support in Multi-Hop Wireless Networks,” Proceedings of IEEE Vehicular Technology Conference, VTC 1999-Fall, pp. 889-893, November 1999.
[22] M. Chatterjee, S. K. Das, D. Turgut, “WCA: A Weighted Clustering Algorithm for Mobile Ad-hoc Networks, “Journal of Cluster Computing (Special Issue on Mobile Ad-hoc Networks), Vol. 5, No. 2, pp. 193-204, April 2002.
[23] T. Hou, Tzu-Jane Tsai, “An access-based clustering protocol for multihop wireless ad-hoc networks”, IEEE Journal on Selected Areas in Communications, no. 7, pp. 1201-1210, July 2001.
[24] S. Banerjee, S. Khuller, “A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks”, IEEE Infocom 2001, Anchorage, Alaska, April 2001.
[25] S. Galli, H. Luss, J. Sucec, A. McAuley, S. Samtani, D. Dubois, K. DeTerra, R. Stewart, B. Kelley, "A Novel Approach to OSPF-Area Design for Large Wireless Ad-Hoc Networks", IEEE International Conference on Communications (ICC), Seoul, Korea, May 2005.
[26] K. Manousakis, A. McAuley, R. Morera, J. Baras, “Using Multi-objective Domain Optimization for Routing in Hierarchical Network,” WIRELESSCOM 2005, Maui, HI, June 2005
[27] R. Ramanathan, M. Steenstrup, “Hierarchically Organized, Multihop Mobile Networks for Quality of Service Support,” ACM/Baltzer Mobile Networks and Applications, Vol.3, No.1, pp.101-19, January 1998.
[28] K. Manousakis, A. McAuley, ”Distributed Active Hierarchy Maintenance in MANETs,” IEEE International Conference on Communications (ICC), Istanbul, Turkey, June 2006