Proceeding of the 2004 American Control Conference
Boston, Massachusetts June 30 - July 2, 2004



# Interior Point-based Optimization for Joint Admission Control and Routing in IP Networks

Kalyan Kuppuswamy and Daniel C. Lee

***Abstract*— For MPLS-like IP networks, we propose a novel and practically implementable optimal Joint Admission Control and Routing *(JACR)* mechanism for incoming trunk traffic classes with distinct QoS requirements. We base our proposed mechanism on Interior Point-based optimization techniques to solve the admission control/routing problem in large service provider networks. The significant contributions of our proposed mechanism are a) an efficient resouce allocation mechanism that i) simultaneously satisfies both user QoS requirements and network provider's capacity and routing constraints and ii) engineers the usage of network resources efficiently by load-balancing and preventing hotspots from developing in the network, b) computationally efficient to make decisions within real-time call (flow) setup durations, c) scalability to network size and complexity and d) lends itself to a distributed implementation. The proposed mechanism runs in two distinct phases - Phase-1 is the admission control and feasible route determination step and Phase-2 is the optimal routing step. This structure provides flexibility to the service provider to implement this mechanism in its complete form or even implement it solely as an admission control/feasible route determination algorithm (i.e. use Phase-1 only). Experimental results on large-size service provider network with 56 nodes, 117 links and 14 LSPs between an ingress-egress pair demonstrate that, for incoming traffic with diverse QoS requirements, the proposed mechanism is very efficient in determining admission and allocating network resources by optimal routing, all within real-time call setup times.**

## I. Introduction

Efficient resource allocation is a key requirement for optimizing operational efficiency of MPLS-like IP networks [1]-[2]. In this paper, we propose a novel and practically implementable *Joint Admission Control and Routing (JACR)* mechanism for trunk traffic sessions that are injected into the service provider's network. Our *JACR* is an efficient network resource allocation mechanism based on Interior Point optimization techniques, and it takes into account the QoS requirements of the incoming traffic, and equally importantly, the network provider's capacity and routing constraints. This mechanism also engineers the allocation of network resources efficiently by load-balancing and preventing hotspots from developing in the network. Furthermore, if we view each 'call' as a trunk reservation request, our *JACR* mechanism could also guide a service provider in its Service Level Agreements (SLAs) with a subscriber (which could be another network provider itself) that requests for a trunk reservation.

With these objectives in mind, we propose the *Joint Admission Control and Routing* mechanism with the following key characteristics. These, we believe, are the novel and significant contributions of our work.

- Optimal resource allocation mechanism - the admission control/routing problem for large service provider networks with user and network constraints is solved based on Interior Point optimization techniques, which leads to optimal network resource allocation and steers the network towards improved operational efficiencies.
- Fast computational requirements of the algorithm - the trunk traffic requests are presented to different ingress points in the network for admission, and the admission control and routing decisions must be made within real-time call setup time durations. Interior Point-based optimization methods have been found to be very efficient (superlinear to quadratic rate of convergence) for solving large non-linear optimization problems [9]-[12].
- Scalability to large network size and complexity - since our proposed algorithm is based on Interior Point methods that exhibit good scaling property to the size of the optimization problem (number of iterations grow much more slowly as the problem dimension grows [11]), it makes them ideal candidates for large-scale networks.
- Distributed implementation - Different ingress nodes run the admission control and routing algorithm independently of each other and make decisions using the globally available topology information of the network.
- Incorporates general cost and constraint functions - the algorithm incorporates general, non-linear convex cost functions and constraint sets (incoming trunk traffic (user) QoS requirements, network routing and capacity constraints), which allows for flexible and richer set of user and network constraints to be imposed, and solved.
- Structural advantages of the algorithm - the algorithm runs in two distinct phases - Phase-1 is the admission control step (it also determines a feasible route i.e. a feasible rate on the LSP routes to the egress that satisfies incoming traffic QoS requirements and network capacity and routing constraints) and the next phase, Phase-2 is the optimal routing step that determines

K. Kuppuswamy is with Department of Electrical Engineering - Systems, University of Southern California, 3740 McClintock Ave, Los Angeles, CA 90089 and Qualcomm Inc, San Diego kalyank@qualcomm.com

D.C. Lee is with Department of Electrical Engineering - Systems, University of Southern California, 3740 McClintock Ave, Los Angeles, CA 90089 dclee@usc.edu

optimal rates on the LSP routes to egress based on minimizing a link cost function. This structure provides flexibility to the service provider to implement the algorithm in its complete form, or even implement it as an admission control/feasible route determination algorithm solely (i.e. use Phase-1 only). An example scenario could be in a lightly loaded network, the service provider might be just interested in admission control step, and may choose to skip the optimal routing step.

Our work differs significantly from the vast array of existing literature on MPLS traffic engineering/resource allocation mechanisms; e.g. [1]- [6] and references cited therein. Our work differs from the work of [4] where they consider revenue maximization considering the traffic across all ingress-egress pairs, which is more centralized approach. The present paper takes a more decentralized approach where the different ingress nodes run the admission/ routing control independently of one another, while taking into account globally available network topology information and link residual bandwidths for making admission and routing decisions. The approach taken by the present paper has an advantage of reducing signaling overheads to disseminate information between ingress points and a central admission/routing control server (that too, within a fraction of call setup times). Additionally, the more decentralized approach taken by the present paper can bypass the issues with reliability and computational and network signaling load at the server. There are other contributions [5], [6] on constrained multipath traffic engineering schemes, but their problem formulation is based on integer programming techniques, and their objectives are quite different from ours.

The present paper is organized as follows. Section 2 provides the overview of the proposed *JACR* mechanism. Section 3 provides the mathematical model of the problem. Section 4 develops the *JACR* algorithm based on Interior Point methods, describes the Phase-1 (Admission Control) and Phase-2 (Optimal Routing) steps, and finally describes the overall *JACR* algorithm. Section 5 provides experimental results to illustrate the efficiency of our proposed mechanism on large-scale networks and we conclude in Section 6.

## II. Proposed Joint Admission Control and Routing (JACR) Mechanism

We assume that in a MPLS-supported network, the LSP path layout between different designated Ingress-Egress (IE) pairs is pre-determined and laid out based on historic traffic information. Additionally, it is assumed that multiple LSPs (order of ten, for e.g.) have been pinned down between any IE pair. The newly arriving trunk traffic sessions are presented to the network for admission and routing at different ingress points in the network. The sessions belong to different service classes and have their own distinct QoS requirements - Committed Burst Rate (CBR), Peak Data Rate (PDR), and maximum tolerable quality degradation (e.g. packet loss rate). We assume that if the incoming session is admitted at an ingress node, then it can get routed through multiple LSPs between the IE pair (i.e. the trunk traffic can be split among the LSP routes between the IE pair).

The ingress routers run the *JACR* algorithm, which is based on Interior Point optimization methods and has the following two phases - a) Phase-1 Admission Control - it determines whether the incoming trunk session can be admitted into the network by satisfying the session QoS constraints, network capacity and routing constraints. If not, the incoming session is rejected. If the trunk session is admitted, this phase also determines a feasible route (i.e. it determines a feasible rate allocation on the LSP routes to the egress). The next step b), is Phase-2 Optimal Routing - where the *JACR* mechanism determines the optimal rates on the LSP routes based on minimizing the cost of an overall objective function based on link cost metric. The optimal rates are then reserved on LSP routes between the IE pair.

The link cost metric is determined by the current residual capacity per link, which is the difference between the link capacity and the total rate reserved for previously admitted trunk sessions on LSP routes that pass through the link. The ingress routers keep getting periodic updates (order of few minutes) of the residual capacity of the links from different routers in the network through TE extensions to OSPF mechanism [7], etc.

Furthermore, we assume that once an incoming trunk traffic session is admitted into the network, the LSP flow rates determined by the *JACR* mechanism get reserved on the paths and are guaranteed throughout the length of the session. This ensures that subsequent traffic arrivals do not impact the currently admitted traffic in meeting their QoS requirements.

## III. Mathematical Formulation of the Problem

### A. Model

Consider the network with a set of some unidirectional links and predefined LSP routes setup between different designated IE pairs. Consider an IE pair with a set, $\mathcal{P}$, of LSPs available to it. Let the cardinality of set $\mathcal{P}$, i.e. the total number of LSPs be $N$. Let $\mathcal{L}$ (with cardinality $L$) to be the set of links through which the LSP routes pass through i.e., $\mathcal{L} = \{l | l \in p, p \in \mathcal{P}\}$. The incoming trunk traffic session belongs to a class $s \in \mathcal{S}$ ($\mathcal{S}$ is the set of services supported) and has distinct QoS requirements. The input rate is bounded by a minimum CBR $r_{l,s}$ and a PDR $r_{u,s}$. The ingress router routes the incoming traffic through the set $\mathcal{P}$ of LSPs, with $x_p$ denoting the rate provided through LSP $p \in \mathcal{P}$, such that $r_{l,s} \leq \sum_{p \in \mathcal{P}} x_p \leq r_{u,s}$. Note that providing the PDR may not be a hard constraint.

Also associated with each service class is a quality degradation function (e.g. packet loss rate function) $p_s(\sum_{p \in \mathcal{P}} x_p, r_{u,s})$, which is a function of actual rate admitted by the network $\sum_{p \in \mathcal{P}} x_p$ and peak rates demanded

$r_{u,s}$. The function $p_s(.,.)$ is assumed to be a decreasing, continuous, twice differentiable convex function of admitted rates $\sum_{p\in\mathcal{P}} x_p$ on LSPs defined in $[0, r_{u,s}]$ (differentiable in $(0, r_{u,s})$, it is also convex in each rate $x_p, p \in \mathcal{P}$). It is assumed to be such that if the admitted rate equals the peak rate, then there is no degradation (loss) for this session. We assume that each service class imposes a maximum quality degradation threshold (e.g. packet loss rate threshold) on the network, $\rho_s$. We express this constraint as $p_s(\sum_{p\in\mathcal{P}} x_p, r_{u,s}) \leq \rho_s$.

Consider a link $l \in \mathcal{L}$ that advertises its residual capacity as $c_l$. Flow on link $l$ is due to flows from LSP routes that pass through the link $l$, and is given by $x_l = \sum_{\{p|l\in p, p\in\mathcal{P}\}} x_p$. We define a per link cost function as $C_l(x_l)$, which is assumed to be a continuous, twice differentiable, convex function in $x_l$. We define the *JACR* problem as an optimization problem, where we seek to minimize the overall link cost with respect to the optimization variables, LSP flow rates, $x_p, p \in \mathcal{P}$.

$$\min \sum_{l\in\mathcal{L}} C_l(\sum_{\{p|l\in p, p\in\mathcal{P}\}} x_p) \quad (1)$$

$$\sum_{p\in\mathcal{P}} x_p \geq r_{l,s} \quad (2)$$

$$\sum_{p\in\mathcal{P}} x_p \leq r_{u,s} \quad (3)$$

$$\sum_{\{p|l\in p, p\in\mathcal{P}\}} x_p \leq c_l \ \forall l \in \mathcal{L} \quad (4)$$

$$x_p \geq 0 \ \forall p \in \mathcal{P} \quad (5)$$

$$p_s(\sum_{p\in\mathcal{P}} x_p, r_{u,s}) \leq \rho_s \quad (6)$$

The constraints (2)-(5) are linear constraints and may be expressed in the matrix form $Ax - b \geq 0$ where $x$ is the vector $(x_1, x_2, \ldots, x_N)'$ of input flow distribution among the $N$ LSPs (we assume $'$ to be the transpose operator on a matrix). Matrix $A$ is of dimension $m \times N$, where $m = N + L + 2$, and is given by $A = \begin{pmatrix} e & -e & -R'_{L\times N} & I_{N\times N} \end{pmatrix}'$. where $e = (11\ldots1)'$ is a $N \times 1$ vector, $I_{N\times N}$ is the $N \times N$ identity matrix, $R_{L\times N}$ is a route matrix of $L \times N$ dimension, where an $i \times j$ entry of $1$ denotes that LSP $j$ passes through link $i$, else it is $0$. Vector $b = \begin{pmatrix} r_{l,s} & -r_{u,s} & -c' & 0'_N \end{pmatrix}'$ where $0_N$ is the vector $(0, 0, \ldots, 0)'$ with $N$ entries and $c$ is the vector of residual capacities $(c_1, c_2, \ldots, c_L)'$.

To simplify the usage of matrix $A$ and vector $b$, we define $A = (a_1, a_2, \ldots, a_m)'$, where the entries $a_1, \ldots, a_m$ are $N \times 1$ vectors, and $b = (b_1, b_2, \ldots, b_m)'$, where $b_1, \ldots, b_m$ are scalars. Similarly, we state the non-linear constraint (6) as $\rho_s - p_s(x, r_{u,s}) \geq 0$.

We define $\mathcal{F}$ to be the convex feasible region defined by the constraints (2)-(6). We assume it is non-empty and bounded.

### *B. Interior Point Method for Joint Admission Control and Routing*

We choose Interior Point or Barrier Methods for solving our *JACR* problem since they have been proven to efficiently (fast convergence, scalable, etc.) solve large-scale non-linear optimization problems [9]-[12]. Our *JACR* problem falls in this category - it is a non-linear convex optimization problem, and each instance of the problem has at least $N + L + 3$ constraints (2)-(6) with $N$ optimization variables (LSP flow rates between an IE pair). $N$ is assumed to be in the order of ten (in fact, this is on the higher side, in [3], they quote "two to five LSPs per IE pair is a typical setting which exists in an operational ISP network that implements MPLS"). The constraint set size is assumed to be typically in the order of hundred for a large-scale network.

We formulate our problem *JACR* using Logarithmic Barrier function as follows.

$$B(x,\mu) = \sum_{l\in\mathcal{L}} C_l(\sum_{\{p|l\in p, p\in\mathcal{P}\}} x_p) - \mu \sum_{i=1}^{m} log(a'_i x - b_i) \quad (7)$$
$$-\mu\{log(\rho_s - p_s(x, r_{u,s}))\}$$

where $\mu$ is called *barrier parameter* and is strictly positive.

We seek the unconstrained minimizer of (7) as $\mu$ approached $0$, and this solves the constrained minimization problem *JACR* (1) [9]-[12]. The iterative procedure for solving (7) is detailed in the next section.

### *C. KKT Conditions for Optimality*

We formulate the Karush-Kuhn-Tucker (KKT) conditions for finding the unconstrained minimizer of $B(x,\mu)$. The gradient $\nabla B(x,\mu)$ is given by $\nabla B(x,\mu) = g(x) - \mu \sum_{i=1}^{m} \frac{a_i}{a'_i x - b_i} - \mu \frac{\nabla p_s(x, r_{u,s})}{p_s(x, r_{u,s}) - \rho_s}$, where $g(x)$ is a $N \times 1$ vector with $k^{th}$ element given by $g(x_k) = \sum_{\{p|l\in p, p\in\mathcal{P}\}} \frac{\partial C_l(\sum_{\{p|l\in p, p\in\mathcal{P}\}} x_p)}{\partial x_k}$. Assume $x^*(\mu)$ is the unconstrained minimizer for $B(x,\mu)$ for a given $\mu$. The first order KKT condition is -

$$\nabla B(x^*(\mu), \mu) = 0 \quad (8)$$

The second order KKT condition for optimality is that the Hessian $\nabla^2 B(x(\mu), \mu)$ must be positive definite at $x^*(\mu)$.

The Hessian $\nabla^2 B(x(\mu), \mu)$ is given by $\nabla^2 B(x(\mu), \mu) = H(x(\mu)) + \sum_{i=1}^{m} \frac{\mu}{(a'_i x(\mu) - b_i)^2} a_i a'_i + \mu \frac{\nabla p_s(x(\mu), r_{u,s}) \nabla p_s(x(\mu), r_{u,s})'}{(\rho_s - p_s(x(\mu), r_{u,s}))^2} + \mu \frac{\nabla^2 p_s(x(\mu), r_{u,s})}{\rho_s - p_s(x(\mu), r_{u,s})}$

where $H(x(\mu))$ is the Hessian of the cost function of the objective function. Evaluated at $x^*(\mu)$, each of the terms in the RHS of the Hessian is positive definite due to the convex properties of the cost function and the constraint functions. Hence $\nabla^2 B(x(\mu), \mu)$ is also positive definite and the second-order KKT optimality condition is satisfied.

## IV. Algorithm for Joint Admission Control and Routing

In this section, we develop an iterative interior point algorithm to solve for *JACR* problem. For each iteration

$k \geq 1$, starting with a given barrier parameter $\mu_0$, we reduce the barrier parameter $\mu_k$ by a cut factor $\gamma \in (0, 1)$ and solve for the minimizer $x^*(\mu_k)$ for $B(x, \mu_k)$ as the barrier parameter $\mu_k$ converges to $0$. The next subsection outlines the Newton's Method for solving the KKT equation (8) for $x^*(\mu_k)$. We subsequently develop the *JACR* algorithm.

### A. Newton's Method for Solving KKT Equation

Consider any iteration $k \geq 1$, and the barrier parameter to be $\mu_k$. We seek to solve for unconstrained minimizer $x^*(\mu_k)$ for $B(x, \mu_k)$. We call $k$ as an index of *major* iteration. Define $m \geq 1$ to be *minor* iteration index, which denote the iterations carried out to solve for $x^*(\mu_k)$. We shall denote the iterates to be $x_{k,m}$. Define initial feasible point $x_{k,0} = x^*(\mu_{k-1})$ for $k > 1$ (the minimizer obtained in previous major iteration $k-1$) and for $k = 1$, $x_{k,0}$ is set to the chosen initial strict feasible point. Furthermore, define $\Delta x_{k,m} = x_{k,m} - x_{k,m-1}$. Newton's method for solving the KKT equation (8) is given by

$$\nabla^2 B(x_{k,m}, \mu_k)\Delta x_{k,m} = -\nabla B(x_{k,m}, \mu_k) \tag{9}$$

The equation (9) can be solved iteratively in minor index $m$ (by standard line search methods) to obtain the unconstrained minimizer $x^*(\mu_k)$. The sequence of minimizers for each iteration $k$, $x^*(\mu_k)$ converge to the constrained minimizer $x^*$ of *JACR* problem as $\mu_k \rightarrow 0$ (we have rigorously proved this convergence, but the proof is long and has been omitted in this paper due to lack of space). However, we prove an important proposition that is a key to the convergence proof.

**Proposition :** The cost function for *JACR* is monotonically non-increasing with increasing iterations $k$, i.e. all $k$, we have the following relations that hold good -

$$\sum_{l \in \mathcal{L}} C_l \left( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_{k+1}) \right) \leq \sum_{l \in \mathcal{L}} C_l \left( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_k) \right) \tag{10}$$

where $x_p^*(.)$ is the $p^{th}$ component of vector $x^*(.)$.

**Proof:** By definition, we have,

$$\sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_k)) - \mu_k \sum_{i=1}^{m} log(a_i' x^*(\mu_k) - b_i) -$$
$$\mu_k log(\rho_s - p_s(x^*(\mu_k), r_{u,s})) \leq \sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_{k+1})) -$$
$$\mu_k \sum_{i=1}^{m} log(a_i' x^*(\mu_{k+1}) - b_i) - \mu_k log(\rho_s - p_s(x^*(\mu_{k+1}), r_{u,s}))$$

and,

$$\sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_{k+1})) - \mu_{k+1} \sum_{i=1}^{m} log(a_i' x^*(\mu_{k+1}) - b_i) -$$
$$\mu_{k+1} log(\rho_s - p_s(x^*(\mu_{k+1}), r_{u,s})) \leq \sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_k)) -$$
$$\mu_{k+1} \sum_{i=1}^{m} log(a_i' x^*(\mu_k) - b_i) - \mu_{k+1} log(\rho_s - p_s(x^*(\mu_k), r_{u,s}))$$

Multiplying the first inequality by $\frac{\mu_{k+1}}{\mu_k}$, and adding to the second, we get the following -

$$\sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_{k+1})) \left(1 - \frac{\mu_{k+1}}{\mu_k}\right) \leq \sum_{l \in \mathcal{L}} C_l( \sum_{l \in p, p \in \mathcal{P}} x_p^*(\mu_k)) \left(1 - \frac{\mu_{k+1}}{\mu_k}\right) \tag{11}$$

Since $0 < \mu_{k+1} < \mu_k$, (11) implies (10) as desired •

This proposition suggests that the minimizer in each iteration $k$, $x^*(\mu_k)$ successively improves (or retains) the previous cost function for *JACR* and the cost function progresses towards the minimum (i.e. $x^*(\mu_k)$ tends towards $x^*$) with increasing $k$.

Next, we use the above method for developing the *JACR* algorithm. The algorithm developed in two phases. Phase-1 (Admission Control) determines whether a feasible solution exists to the *JACR* problem, and if so, Phase-2 (Optimal Routing) then determines the optimal solution to the problem. Together, they constitute the *Joint Admission Control and Routing* algorithm.

### B. Phase-1 - Admission Control

The admission control attempts to find a feasible point in the constraint set, the convex region $\mathcal{F}$ (which satisfies the incoming trunk traffic session QoS requirements and the network capacity and routing constraints). If it fails to find a feasible point, then the admission criteria fails and the incoming session is rejected. Otherwise, if a feasible point in $\mathcal{F}$, then the *JACR* algorithm proceeds to Phase-2, optimal routing step. The admission control problem is a challenging one considering that for a large-scale network, the constraint set size is large (typically of the order of hundred).

We may express the constraint set by the set of constraint equations $c_i(x) \geq 0$, where $c_i(x) = a_i' x - b_i$ for $i \in \{1, \ldots, m\}$ and $c_{m+1}(x) = \rho_s - p_s(x, r_{u,s})$. This follows from equations (2)-(6). We define $v(x)$ as the vector of constraint violations at $x$ where the $i^{th}$ component is defined as $v_i(x) = \max(-c_i(x), 0)$ for $i \in \{1, \ldots, m\}$. Given an initial infeasible point $x_{init}$, we wish to find a feasible point in $\mathcal{F}$ to satisfy the admission control criteria. The constraint violation $v_i(x_{init})$ is non-zero for at least one $i$, and we wish to find a feasible point $x_0 \in \mathcal{F}$ such that $v(x_0) = 0$. We can setup this problem as minimizing the maximum violation in order to find a feasible point in $\mathcal{F}$. Define the *infinity-norm* of the violation vector $v(x)$ as $\|v(x)\|_\infty = \max_{1 \leq i \leq m+1} |v_i(x)|$. Then the problem can be cast as the following convex unconstrained problem -

$$\min_{x \in R^N} \|v(x)\|_\infty \tag{12}$$

To solve this problem, we first define $c_i(x) = r_i(x) - v_i(x)$, $r_i(x) \geq 0$, $v_i(x) \geq 0$ where $r_i(x)$ and $v_i(x)$ are the magnitude of the positive and negative parts of $c_i(x)$ for $1 \leq i \leq m+1$. The original unconstrained problem may be written as the following constrained problem:

$\min_{x \in R^N} \max_{1 \leq i \leq m+1} v_i(x)$ such that $c_i(x) = r_i(x) - v_i(x), 1 \leq i \leq m+1$, $v(x) \geq 0$, $r(x) \geq 0$.

We further introduce a non-negative variable $s$ such that $v_i(x) \leq s$ for all $1 \leq i \leq m+1$. Minimizing $s$ is then equivalent to minimizing the maximum element of $v(x)$. The equality constraint $c_i(x) = r_i(x) - v_i(x)$ can be replaced by $r_i(x) - c_i(x) = v_i(x) \leq s$, or equivalently, $-c_i(x) \leq s$ since $r_i(x) \geq 0$, or $s + c_i(x) \geq 0$. The problem may be restated as the following -

$$\min_{x \in R^N, s \in R} s \tag{13}$$

$$\text{s.t. } c_i(x) + s \geq 0, 1 \leq i \leq m+1 \tag{14}$$

$$s \geq 0 \tag{15}$$

Note that the initial infeasible point in $\mathcal{F}$, $x_{init}$ and $s_{init} = max_{1 \leq i \leq m+1} v_i(x_{init})$ is feasible in the constraint region defined by (14)-(15). We call this the *Phase-1* optimization problem. Note that this is again a convex optimization problem and can be solved using interior point methods as above. For the special case when the constraint set $c_i(x)$ for $1 \leq i \leq m+1$ is a linear set, then the *Phase-1* problem reduces to a simpler Linear Programming problem. To find a feasible solution in $\mathcal{F}$, the optimal solution of the *Phase-1* problem must have $s_{opt} = 0$. However, if the solution to the *Phase-1* optimization problem (13) is $s_{opt} > 0$, then the *JACR* problem does not have a feasible solution (i.e. admission criteria fails).

### C. Phase-2 - Optimal Routing

Let $x_0$ be a feasible solution [1] after solving *Phase-1* optimization problem. Phase-2 is an iterative process where an initial barrier parameter $\mu_0$ is chosen. For each iteration $k \geq 1$, $\mu_k$ is decreased by a cut factor $\gamma \in (0, 1)$ and the unconstrained minimum of $B(x, \mu_k)$ (7), $x^*(\mu_k)$ is found using Newton's method for solving KKT equation (9). Then, for next iteration $k+1$, the barrier parameter $\mu_k$ is reduced by cut factor $\gamma$ and $x^*(\mu_k)$, the previous iteration optimal solution is used as the initial point for solving for the unconstrained minimum $x^*(\mu_{k+1})$ of $B(x, \mu_{k+1})$. This process is repeated till the barrier parameter $\mu_k \rightarrow 0$ (in practice, a very small value $\tau$, like $10^{-8}$), where in the unconstrained minimum of $B(x, \mu_k)$ closely approximates the optimal solution for the constrained optimization problem *JACR* (1) (the sequence of minimizers $x^*(\mu_k)$ of $B(x, \mu_k)$ converges to the constrained minimizer $x^*$ of *JACR* as $\mu_k \rightarrow 0$). $x^*$ corresponds to the optimal routing of the incoming trunk traffic through the LSP routes between the IE pair. The overall *JACR* algorithm is stated next.

### D. Overall Joint Admission Control and Routing Algorithm

1) **Phase-1 Admission Control Step:** Choose an initial point $x_{init}$ (may be infeasible in $\mathcal{F}$) and solve for *Phase-1* optimization problem (13). If no feasible solution exists (i.e. $s_{opt} > 0$ for (13), then Admission Control fails. EXIT.
2) **Phase-2 Optimal Route Computation Steps:** Let initial strictly feasible point from *Phase-1* solution be $x_0$. Set $\tau$. Set $x_{1,0} \leftarrow x_0$. Choose $\mu_0$ and cut factor $\gamma \in (0, 1)$. Set $\mu_1 = \gamma\mu_0$. Set iteration $k \leftarrow 1$.
3) while not converged // major iterations
   a) Compute unconstrained minimizer $x^*(\mu_k)$ of $B(x, \mu_k)$ (7) using Newton's method for solving KKT equation (9). (Minor iterations are performed while solving for $x^*(\mu_k)$).
   b) $x_{k+1,0} \leftarrow x^*(\mu_k)$, $\mu_{k+1} \leftarrow \gamma\mu_k$, $k \leftarrow k+1$.

[1] for initiating the barrier method, we need $x_0$ to be strictly feasible inside $\mathcal{F}$. So, in practice, in Phase-1, we solve for $c_i(x) \geq \delta > 0$, where $\delta$ is a very small perturbation. Equation (14) changes to $c_i(x) + s \geq \delta, 1 \leq i \leq m+1$, everything else remains the same for Phase-1 optimization problem (13)

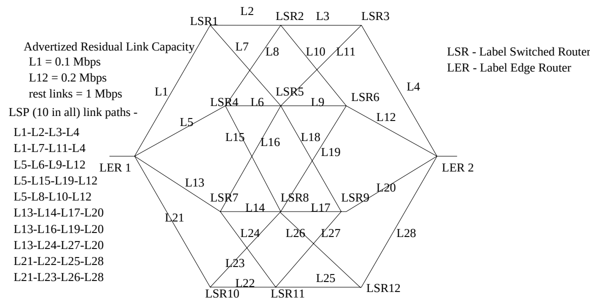


Fig. 1. Network Topology 1 for Admission Control/Routing Problem

## V. Experimental Results

### A. Simulation Setup

We consider two networks topologies for our experimental analysis. Network Topology 1 (NT1) (Figure 1) consists of a 14-node, 28-link and 10 distinct LSPs (routes shown in Figure 1) between a chosen IE pair. Network Topology 2 is more complex (figure not shown) - we use NT1 as subnet connecting to three other such NT1 subnets and generating a 56-node, 117-link overall network (with a maximum depth of 11 hops). There is a set of 14 LSPs between the chosen IE pair. This configuration is closer to real-world service provider networks in terms of size and complexity of network.

In our simulations, we choose a certain IE pair in the provider network (*JACR* runs concurrently on all ingress nodes of the network). We also assume a 'snapshot' of the current network, i.e. the network topology shows the residual capacity in each link (refer Figure 1). The incoming trunk traffic session into the ingress node has the following traffic characterization for the two network topologies. The session demands a CBR of 1.2 Mbps and PDR of 1.5 Mbps as it is injected into the network. The quality degradation function is of the form $p_s(x, r_{u,s}) = \frac{10}{9}(r_{u,s} - \sum_{p \in \mathcal{P}} x_p)^2$ in $[0, r_{u,s}]$, and $\rho_s = 10^{-3}$. We fix the cost metric per link $l$, $C_l(x_l) = x_l/(c_l - x_l)$, a commonly considered convex, twice differentiable cost function based on $M/M/1$ queue approximation [8].

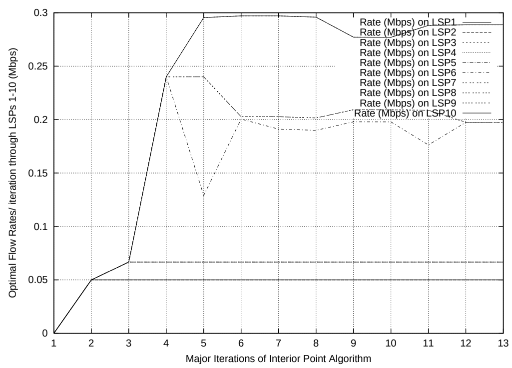


Fig. 2. Convergence of Optimal Flow Rates/iterations for Network Topology 1, LSPs 1-10, Major Iterations 1 to 4 correspond to finding a feasible solution (Admission Control) and iterations 5-13 is the Optimal Routing stage

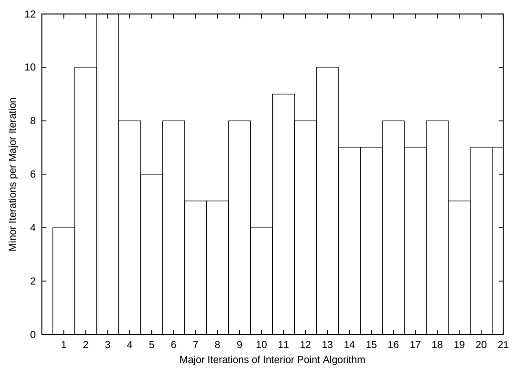


Fig. 3. Computational Load: Minor Iterations per Major Iteration for Network Topology 2

### *B. Simulation Results and Discussion*

We demonstrate the efficient resource allocation and performance (good convergence properties) of the *JACR* algorithm. Figure 2 demonstrates how the final routing allocation (optimal LSP flow rates) adjust themselves to avoid bottleneck links (here L1 and L12 links) and load balance among the other links with equal residual capacities, thus preventing hot spots from developing in the network. Interestingly, the LSP flow rates through bottleneck routes (L1 and L12) are identified very quickly (in $3$ major iterations) by the algorithm. Figure 2 also demonstrates the fast convergence of flow rates in LSPs 1-10 for network topology 1 and the optimal flow rates are found within $\sim 13$ major iterations of the algorithm. The initial point $x_{init}$ is chosen to be an infeasible rate (we chose all-zero vector for the LSP rates). 4 Major iterations were required to find a feasible solution (Phase 1), so if the service provider wishes to run the algorithm solely as an admission control/feasible route determination algorithm, then the algorithm terminates here. The remaining 9 major iterations are spent on finding the optimal route through the LSPs. A feature of the convergence is that successive iterations *improve* upon previous iterations towards the optimal solution (in Figure 2, in $\ell_2$-norm, the sequence of $\|x^*(\mu_k) - x^*\|$ is monotonically decreasing with iterations $k$, validating the proposition proved previously), so every iteration produces minimizer 'closer' towards optimal solution.

Figure 3 demonstrates the performance efficiency of *JACR*, i.e. computational load in terms of total iterations (major and minor) required to converge to optimal solution for the more complex network topology 2. The size of the optimization problem is large - LSP flows, $N = 14$ (order of ten) and total number of linear and non-linear constraints are $N+L+3 = 104$ (order of hundred). The initial Phase-1 Admission Control step to find a strict feasible point took 6 major iterations (with a total of $48$ minor iterations) starting with an initial all-zero vector of infeasible LSP rates. The convergence in Phase-2 Optimal routing took $\sim 15$ major iterations (with a total of $105$ minor iterations). In terms of running time, on Sun Sparc workstation with a modest $750$ MHz CPU speed, our algorithm (Phase-1 and Phase-2 total) ran in $\sim 10.8$ seconds to converge to optimal solution, which is within the real-time call setup duration, thus demonstrating the fast convergence of our algorithm in practice. The performance result demonstrates that our algorithm scales well to size and complexity of the network (and hence the size of the non-linear optimization problem), which enhances its practical applicability even further.

## VI. Conclusions

Our goal in this paper was to effectively apply optimization principles towards developing an efficient network resource allocation *JACR* mechanism. Experimental results demonstrate the efficiency of our mechanism and it scales well with the size and complexity of the network. We feel that our proposed mechanism should be of great interest to service providers who are looking out for efficient implementation techniques to improve the overall operational efficiency of their networks.

## References

[1] D.O. Awduche, J. Malcolm, A. Agogbua, M. O'Dell, and J. McManus, Requirements for Traffic Engineering over MPLS, *RFC 2702*, Sep. 1999.

[2] D.O. Awduche, A. Chui, A. Elwalid, I. Widjaja, and X. Xiao, A framework for Internet Traffic Engineering, *Internet draft, draft-ietf-tewg-framework-01.txt*, Mar. 2000.

[3] A. Elwalid, C. Jin, S. Low, and I. Widjaja, MATE: MPLS Adaptive Traffic Engineering *IEEE INFOCOM 2001*, 1300-1309.

[4] D. Mitra and K.G. Ramakrishnan, A Case Study of Multiservice, Multipriority Traffic Engineering Design for Data Networks, *Proc. Globecomm'99*, Dec 1999.

[5] Y. Lee, Y. Seok, Y. Choi, and C. Kim, A Constrained Multipath Traffic Engineering Scheme for MPLS Networks, *IEEE ICC*, May 2002.

[6] M.K. Girish, B. Zhou, and J.Q. Hu, Formulation of the Traffic Engineering Problems in MPLS Based IP Networks, *ISCC*, 2000.

[7] G. Apostolopoulos, D. Williams, S. Kamat, R. Guerin, A. Orda, and T. Przygienda, QoS Routing Mechanisms and OSPF Extensions, *RFC 2676*, Aug. 1999.

[8] D. Bertsekas and R. Gallagher, *Data Networks*, Englewood Cliffs, NJ: Prentice Hall, 2nd ed. 1992.

[9] D. Bertsekas, *Nonlinear Programming*, Athena Scientific, 1995.

[10] P.E. Gill, *MA 271A-B-C Numerical Optimization Class Notes*, University of California, San Diego, 2002.

[11] Y. Ye, *Interior Point Algorithms*, John Wiley and Sons, 1997.

[12] M. Wright, *Interior Methods for Constrained Optimization*, Acta Numerica, 1992.