

**UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO**

**Dimensionamento em Redes IP com Qualidade de Serviço:
Uma Abordagem via Metaheurística GRASP**

Dárlinton Barbosa Feres Carvalho

Orientador: Prof. Dr. Carlos Frederico M. da Cunha Cavalcanti
Co-orientador: Prof. Dr. Marccone Jamilson Freitas Souza

Relatório Técnico DECOM 000/03

Ouro Preto - MG
Dezembro de 2003

Tudo seria muito fácil, se não fossem as dificuldades.
Barão de Itararé

Resumo

Neste trabalho são apresentados alguns conceitos sobre a nova geração da Internet, a Internet com Qualidade de Serviço. Contextualizado na arquitetura do projeto Tequila, foram estudados o problema de dimensionamento de rede com sua proposta de implementação atual. A contribuição deste trabalho é um novo método baseado na metaheurística GRASP com uma função de avaliação hierárquica com indicadores de qualidade das soluções e uma nova proposta de estrutura de vizinhança que pode ser adotada analogamente para problemas complexos de várias etapas. Resultados computacionais são apresentados demonstrando que o método proposto, além de simples, é mais eficaz que o método existente.

Palavras-chave: Dimensionamento de Redes com Qualidade de Serviço, GRASP, Metaheurísticas.

Abstract

This work presents some concepts about the new generation of Internet that enhance the current one with guarantee of Quality of Service. At the context of Projeto Tequila's architecture were studied the problem of network dimensioning with the current implementation. The contribution of this work is a new approach based on the metaheuristics GRASP, making use of a hierarchical evaluation function with indicators of quality from solutions and a new proposal of neighborhood structure. Computational results are presented demonstrating that the proposed method, besides simple, it is more efficacy than the existent method.

Keywords: Network Dimensioning with Quality of Service, GRASP, Metaheuristics.

Sumário

Introdução.....	1
Motivações	2
Objetivos.....	2
Geral	2
Específicos	2
Conceitos Iniciais.....	3
Engenharia de tráfego	3
Qualidade de Serviço	3
Serviço Diferenciado	3
PHB	4
Agregado de Tráfego.....	4
Agregado de Ordem	4
MPLS	4
LSP	5
Micro-fluxo.....	5
Fluxo.....	5
Tronco de tráfego	5
Especificação em nível de serviço	5
Modelo Funil	5
O Projeto Tequila.....	7
O Módulo de Dimensionamento de Rede	8
Formulação do Problema	10
Implementação Atual	11
Construção de uma solução para o problema	12
Restaurando inviabilidades.....	13
Método Proposto	13
Construção da Solução Inicial.....	14
Fase de construção GRASP.....	14
GRASP com Filtro.....	15
Função de Avaliação.....	15
Refinamento da Solução	16
Estudos preliminares	16
Método de descida	17
Simulated Annealing	17
Resultados Computacionais	19
Conclusões	25
Trabalhos Futuros.....	26
Referências Bibliográficas	26
Apêndice	1
Exemplo de aplicação dos métodos	1

Lista de Figuras

Figura 1: Compartilhamento de enlaces entre rotas	6
Figura 2: Exemplo de uma árvore de Steiner suboptimal	7
Figura 3: Arquitetura Funcional TEQUILA	8
Figura 4: Módulos do Bloco de Engenharia de Tráfego	9
Figura 5: Interface do módulo de dimensionamento de rede	10
Figura 6: Implementação atual do módulo de dimensionamento...	12

Introdução

A proposta inicial da Internet se mostrou muito eficiente para atender aos requisitos do momento histórico em que fora criada, porém, mostra-se falha com os novos requisitos demandados por aplicações mais complexas que implementam troca de informações em tempo real. Utilizando a tecnologia disponível, vários serviços começaram a serem oferecidos como, por exemplo, voz sobre IP (VoIP), redes virtuais privadas (VPNs) e aplicações multimídias [Wang & Crowcroft, 1996]. No entanto, essas aplicações possuem necessidades que não podem ser garantidas pela arquitetura convencional da rede. Neste sentido, novas funcionalidades e projetos de redes foram propostos para estender a Internet [TJoens et al., 2002], como o projeto Tequila, objeto de estudo neste documento. A engenharia de tráfego tornou-se fundamental como uma ferramenta que propicia um uso melhor da rede, direcionando o tráfego de tal forma a usar os recursos disponíveis de uma maneira mais racional. Dentro do projeto TEQUILA, o bloco de dimensionamento de rede propõe uma alocação eficiente do tráfego na rede atendendo as necessidades políticas, financeiras e os requisitos das aplicações sendo uma das principais ferramentas para prover engenharia de tráfego.

Este relatório é uma compilação sobre o estudo do módulo de dimensionamento de rede do projeto IST-Tequila, com todos os conceitos de rede relevantes como QoS, DiffServ, MPLS e modelo funil, bem como propostas de implementação. Este módulo compõe a base da engenharia de tráfego da arquitetura, que está baseada em Serviço Diferenciado (DiffServ) e em MPLS como tecnologias para estender a arquitetura da Internet provendo Qualidade de Serviço (QoS) e engenharia de tráfego. O modelo funil (*hose model*) é uma técnica que possibilita melhorar a alocação do tráfego na rede em relação à alocação convencional (*pipe model*).

Primeiramente são apresentados alguns conceitos que definem uma série de termos utilizados no restante deste documento. O projeto Tequila e o módulo de dimensionamento de rede serão apresentados em seguida com um maior detalhamento.

Uma definição formal da funcionalidade do módulo de dimensionamento de rede está enunciada na seção Formulação do Problema. Em Implementação Atual está descrito como foi projetada uma solução para prover esta funcionalidade.

Em método proposto são apresentados os novos procedimentos para resolução do problema seguido pelos Resultados Computacionais comparando o método existente e o proposto.

Encerrando, Conclusões e Trabalhos Futuros relaciona algumas questões relevantes deste estudo.

Motivações

Para a consolidação do paradigma atual da Internet, com seus novos serviços, são necessários novos modelos de engenharia de tráfego. Neste sentido, o projeto Tequila possui um bloco de gerenciamento de rede para atender às garantias requeridas por estes serviços, e controlar a utilização dos recursos da rede.

Rabbat e Laberteauz afirmam que "É decisivo o controle de fluxo do tráfego pela rede para que qualquer garantia fim-a-fim seja sustentada. O uso de admissão de conexão, roteamento inteligente e esquemas de proteção tornam possível que garantias de qualidade de serviço fim-a-fim sejam estabelecidas. Usando a funcionalidade de Diffserv e acrescentando a esse a funcionalidade roteamento-fixo provido por MPLS, é possível satisfazer garantias de QoS quantitativas em redes como a Internet" [Rabbat & Laberteauz, 2000].

Sendo o problema de roteamento com multi-restrições em QoS (QoSr) pertencente a classe de problemas NP-completo [Garey & Johnson, 1979], [Wang & Crowcroft, 1996], a utilização de heurísticas mostra-se adequado para resolver este tipo de problemas. As metaheurísticas são técnicas que melhoram as soluções geradas, pois, ao contrário das heurísticas convencionais, possibilitam escapar de ótimos locais, além de conseguir se adaptar melhor ao grande volume de restrições e a capacidade de considerar indicadores de qualidade.

Objetivos

Geral

Propor uma nova implementação do módulo de dimensionamento de rede do projeto Tequila baseada em técnicas de metaheurística.

Específicos

Analisar os conceitos de rede relacionados ao problema como QoS, DiffServ, MPLS, modelo funil e o pacote de software *Network Simulator*, principalmente, o módulo GT-ITM.

Conceitos Iniciais

Estes conceitos são apenas as idéias e definições que calcaram o projeto da arquitetura abordada neste trabalho. Na seção Formulação do Problema está exposta sua utilização e relacionamento com o problema.

Engenharia de tráfego

Engenharia de tráfego da Internet é definida como a componente de engenharia da rede Internet que lida com questões de avaliação, desempenho e otimização de redes operacionais IP, através da aplicação de tecnologia e princípios científicos para a quantificação, caracterização, modelagem e controle do tráfego da Internet [Awduche et al., 1999].

Na abordagem do projeto Tequila, a engenharia de tráfego de rede (do termo em inglês *Network Traffic Engineering* - TE) trata de como diferenciar e tratar os diferentes tipos de tráfego em uma determinada rede. TE possui tanto objetivos orientados a usuários bem como objetivos orientados a sistemas. Os usuários esperam certo desempenho que deveria ser satisfeito pela rede. O desempenho esperado depende do tipo de tráfego que a rede transporta e é especificado por um contrato entre o cliente e o provedor de serviço Internet (ISP). O operador do sistema, por outro lado, tenta satisfazer as exigências de tráfego do usuário de um modo viável em relação a seu custo. Conseqüentemente, ele tenta acomodar, tantos quantos possíveis, os pedidos de tráfego usando os recursos de rede disponíveis da melhor forma possível.

Qualidade de Serviço

Dentre as diversas definições para Qualidade de Serviço (QoS) temos a performance de uma rede em relação às necessidades de aplicação e o conjunto de tecnologias que permitem uma rede garantir certo desempenho. As garantias de desempenho fim-a-fim são avaliadas entre os nós de ingresso e egresso dos dados na rede, portanto, é necessário que haja um controle geral do tráfego sobre a rede.

Neste documento o termo QoS é utilizado representando os requisitos das aplicações de um ponto a outro (fim-a-fim) na rede.

Serviço Diferenciado

Segundo [Goderis et al., 2001] a arquitetura DiffServ foi concebida para prover QoS, desde o princípio, de um modo escalonável ao longo da Internet. O conceito principal é manter informação de estado (*state-information*) e complexidade somente nas extremidades da rede enquanto os roteadores de trânsito são responsáveis por aplicar tratamento apropriado no repasse dos

pacotes de IP de acordo com o seu campo Serviços Diferenciados (*Differentiated Services* - DS) do cabeçalho IP.

Existem alguns conceitos em Diffserv que são relevantes para o entendimento do problema, a saber:

PHB é uma sigla do termo *Per-Hop Behavior* que foi definido por [Blake et al., 1998] como "uma descrição do comportamento observável externamente do encaminhamento de um nodo de Serviço Diferenciado aplicado a um determinado agregado de comportamento". Em outras palavras, um PHB é um conjunto de ações feitas pelos roteadores como política de escalonamento, *buffering*, etc., definindo um comportamento de um agregado na rede que implementa Serviço Diferenciado. Três importantes comportamentos, denominados EF, AF e BE, foram propostos e são, atualmente, padrões "de-facto". O PHB, chamado *Expedited Forwarding* (EF) PHB [Jacobson et al., 1999] implementa garantias quantitativas, o *Assured Forwarding* (AF) PHB Group [Heinanen et al., 1999] implementa garantias qualitativas de serviços na rede e o PHB *Best-Effort* (BE), reflete um comportamento sem nenhuma garantia, com apenas o melhor esforço para transmissão, tal qual a Internet funciona hoje.

Agregado de Tráfego (*Traffic Aggregate* - TA) é uma coleção de pacotes que recebem um mesmo código chamado DSCP (*Differentiated Service Code Point*) de tal forma a atribuí-los um PHB. Um agregado de tráfego marcado, por exemplo, para o PHB EF é chamado de "agregado de tráfego EF" ou "agregado EF". Isso generaliza o conceito de Agregado de Comportamento (*Behavior Aggregate* - BA) de um enlace para uma rede [Blake et al., 1998].

Agregado de Ordem (*Ordered Aggregate* - OA) é um conjunto de BAs que compartilha uma restrição de ordenação. O conjunto de PHBs que é aplicado a este conjunto de BAs constitui uma classe de enumeração PHB (*PHB scheduling class*) [Blake et al., 1998].

MPLS

Para prover garantias de QoS fim-a-fim em uma transmissão sobre uma rede é necessário estabelecer *a priori* a rota por onde serão transmitidos os dados.

Rabbat e Laberteaux afirmam que "*Multiprotocol Label Switching* (MPLS) é um protocolo capaz de especificar, ou 'fixar', a rota de um fluxo que provê garantias quantitativas" [Rabbat e Laberteaux, 2000]. O MPLS pode criar túneis pré-determinados entre um par de nodos. Um pacote de IP atravessando uma rota determinada, denominada LSP, é prefixado com um cabeçalho MPLS. Quando um roteador recebe um pacote com um cabeçalho MPLS, o rótulo presente no cabeçalho é usado como chave primária de consulta à tabela de encaminhamento MPLS presente no roteador, onde são recuperados a porta de destino e o novo rótulo do pacote, que foram definidos quando a rota foi estabelecida. Em

resumo, MPLS fixará uma rota particular para um fluxo determinado por um processo de Alocação de Recurso de Rede. MPLS implementa o conceito de caminho virtual e conexão no nível de IP. Assim, LSPs são criados quando uma conexão é aberta.

LSP (*Label Switched Paths*) é um conjunto de enlaces, formando um caminho entre dois nós, ou seja, uma rota por onde os pacotes associados a um determinado LSP irão usar. Os LSP, dentro do contexto da arquitetura utilizada, são configurados na abertura da conexão usando o protocolo MPLS.

Micro-fluxo é definido por [Goderis et al., 2001] como um elemento pertencente ao conjunto dos pacotes que consistem da quintupla endereço de fonte, endereço de destino, número de protocolo, porta de origem e porta de destino, ou simplesmente entendido como um único pacote no padrão IP (*Internet Protocol*).

Fluxo é definido como um elemento pertencente ao conjunto dos pacotes que são tratados pela rede de um modo único e pré-estabelecido. Em geral, um fluxo consiste de um ou mais micro-fluxos, porém, pode-se fazer uso de outros parâmetros além das informações da quintupla do cabeçalho. Por exemplo, qualquer informação disponível no cabeçalho do datagrama IP, tal como LSP em uma arquitetura MPLS, interfaces de ingresso e egresso, rede de origem ou trânsito, etc. O modo como os micro-fluxos são agregados e desagregados e procedimentos de agregação e desagregação são vistos como uma parte consistente do projeto de TEQUILA [Goderis et al., 2001].

Tronco de tráfego (*Traffic Trunk - TT*) é uma agregação de fluxos de tráfego da mesma classe que são colocados em um mesmo LSP. Uma definição precisa para tronco de tráfego está enunciada em Formulação do Problema.

Especificação em nível de serviço

Atualmente, serviços com QoS são oferecidos através de acordos, definidos em termos e cláusulas dos contratos (*Service Level Agreements*), entre um provedor e seus consumidores. Estes termos e cláusulas acordados são convertidos em um conjunto padrão de especificações em nível de serviço (*Service Level Specification - SLS*). O conteúdo de uma especificação inclui os parâmetros essenciais relacionados a QoS, incluindo identificação de escopo e de fluxo, parâmetros de conformação de tráfego e garantias de serviço [Trimintzios, 2001]. Basicamente, um SLS é um termo especificando um serviço acordado em contrato, definido através de um conjunto padronizado de parâmetros.

Modelo Funil

O modelo tubo (*pipe model*) estabelece uma rota entre dois nós na rede, garantindo os requisitos de QoS. Entretanto, em um tronco de tráfego com o mesmo nó de origem com destinos

diversos, se representado apenas no modelo tubo, há necessidade de ser desmembrado em diversos fluxos, quantos forem os nós de destino. Ignora-se a informação de capacidade de banda na origem e seus destinos como um todo, tendo uma visão particular de cada fluxo.

O modelo funil (*hose model*) compreende que os fluxos entre o nó de ingresso e egressos devem ser convergidos em uma mesma alocação no fluxo de dados, até que seja necessária a sua divergência. Nesta abordagem permite-se uma especificação resumida dos troncos de tráfego, pois bastaria especificar o nó de entrada, e sua banda de passagem, e um conjunto de nós egressos com suas respectivas bandas de passagem. A multiplexação dos fluxos na rede possibilita uma otimização do fluxo dos dados [Kumar et al., 2001], diminuindo a banda de passagem reservada na rede, como mostrado na Figura 1.

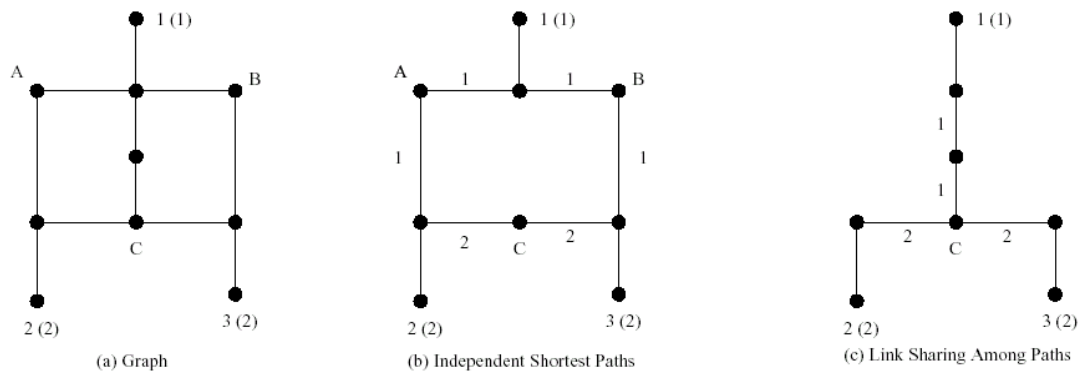


Figura 1: Compartilhamento de enlaces entre rotas

Porém, lidar com o modelo funil implica em aumentar a complexidade na resolução do problema. O problema de determinar o caminho mínimo entre alguns nós em um grafo está relacionado com um clássico da otimização combinatória conhecido como problema da árvore de Steiner [Hwang et al., 1992], classificado como NP-completo [Karp, 1972]. Entretanto o problema de Steiner ainda é insuficiente para resolução ótima deste problema [Kumar et al., 2001] como podemos observar na Figura 2.

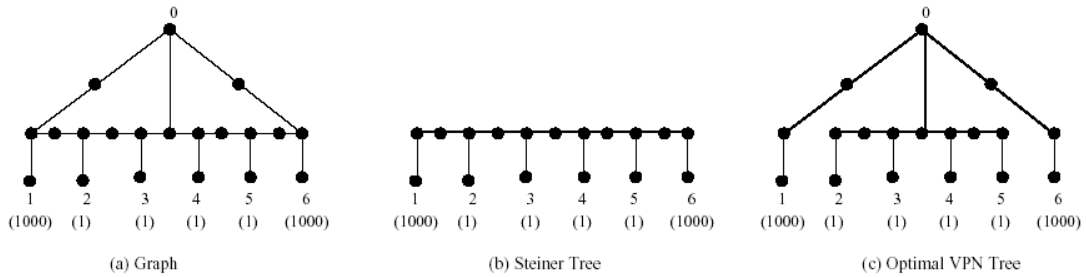


Figura 2: Exemplo de uma árvore de Steiner subóptima¹

A árvore de Steiner considera apenas a informação dos enlaces e não a capacidade requerida pelos nós de ingresso e egressos do fluxo, fundamental para alcançar a otimização proposta pelo modelo funil.

O problema de alocação de tráfego utilizando o modelo funil foi bem estudado para redes virtuais privadas (VPNs), que possui restrição apenas em relação a banda de passagem, tendo como soluções métodos exatos como demonstrado em [Kumar et al., 2001]. Entretanto para uma abordagem de QoS com multi-restrições justifica-se o uso de heurísticas devido ao elevado número de restrições e a alta complexidade do problema conforme encontrado na literatura [Cavalcanti, 2000]. Neste trabalho é utilizada a heurística proposta por [Cavalcanti, 2000], utilizada na implementação atual do módulo de dimensionamento de redes, sem realizar uma análise detalhada sobre o assunto.

O Projeto Tequila

O objetivo do projeto Tequila (*Traffic Engineering for Quality of service in the Internet at Large*) é estudar, especificar, implementar e validar um conjunto de definições de serviço e ferramentas de engenharia de tráfego para obter garantias quantitativas fim-para-fim de Qualidade de Serviço através de cuidadoso planejamento, dimensionamento e controle dinâmico de técnicas para gerenciamento de tráfego qualitativo simples e escalonável dentro da Internet (i.e., Diffserv). As áreas técnicas seguintes serão enunciadas: (a) Especificação de SLSs estático e dinâmico, intra e inter-domínio para sustentar usuários fixos e nômades. (b) Protocolos e mecanismos para transação, monitoramento e obrigações SLSs. (c) Esquemas de engenharia de tráfego para intra e interdomínio para assegurar que a rede pode contender com o SLSs contraído - dentro de domínios, e na Internet como um todo. Toda funcionalidade especificada será validada por simulação, desenvolvimento de protótipo e experimentos em rede [Goderis et al., 2001].

¹ Não ótima para o problema.

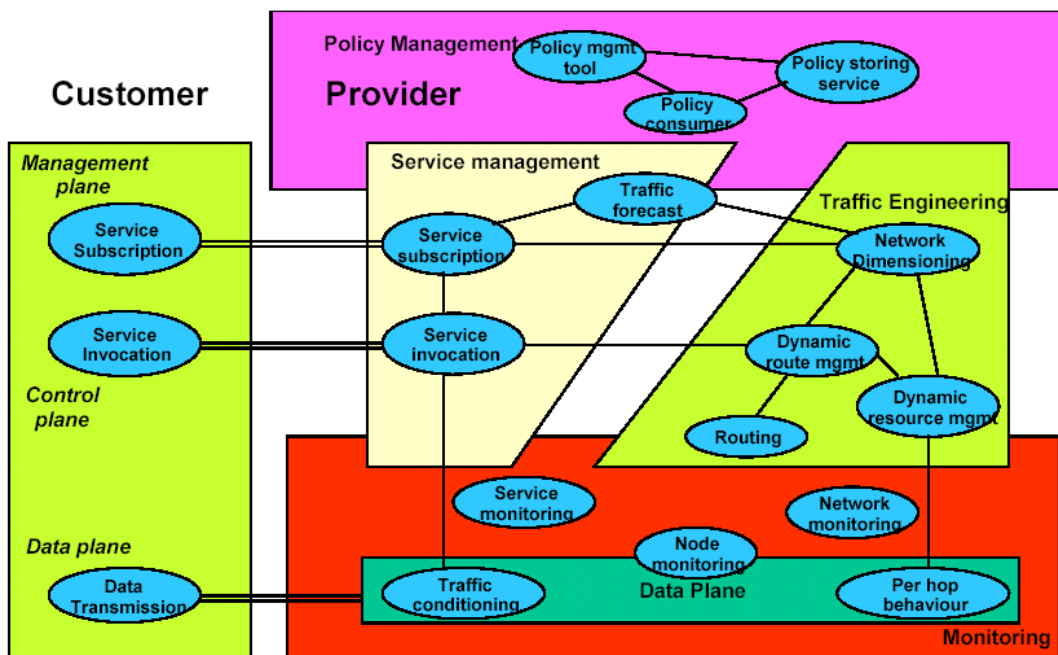


Figura 3: Arquitetura Funcional TEQUILA

A arquitetura funcional do projeto Tequila, ilustrada na Figura 3, está decomposta em 2 elementos, o cliente (*Customer*) e o provedor de serviços (*Provider*). As atividades do cliente estão estratificadas em três planos: um plano gerencial (*Management plane*) onde são contratados novos serviços, um plano de controle que aciona os serviços contratados e um plano de dados para o envio de dados na rede. No provedor estão definidos quatro blocos funcionais:

Gerenciador de Políticas (*Policy Management*), responsável por tratar das políticas de utilização da rede. Essas políticas regulam alguns requisitos de funcionamento da rede na arquitetura.

Gerenciador de Serviços (*Service Management*), onde são controlados os serviços acordados (SLS) e realizada a previsão do tráfego a ser alocado na rede.

Engenharia de Tráfego (*Traffic Engineering*), tem como finalidade definir as configurações da rede para atender aos SLSs e as políticas de rede, gerenciando os dispositivos da rede e realizar o roteamento na rede.

Monitoramento (*Monitoring*), analisa a transmissão dos dados na rede, verificando se estão sendo atendidos satisfatoriamente os SLSs e fornece uma análise sobre os dados trafegados na rede.

O Módulo de Dimensionamento de Rede

O módulo de dimensionamento de rede está inserido no bloco de engenharia de tráfego da arquitetura Tequila, como podemos ver através da Figura 4. Sua responsabilidade é prover uma efetiva solução de engenharia de tráfego, determinando a adequada alocação de recursos sujeita a tendências de tráfego, requisitos de qualidade de serviço e restrições de diretivas de políticas. O módulo de dimensionamento de rede tem como entrada a matriz de tráfego fornecida pelo módulo de previsão de tráfego e a topologia de rede que, em função desses dados e das diretivas de política correntes, calcula o conjunto de rotas explícitas (LSPs), e determina os parâmetros para a configuração da rede [Cavalcanti, 2000]².

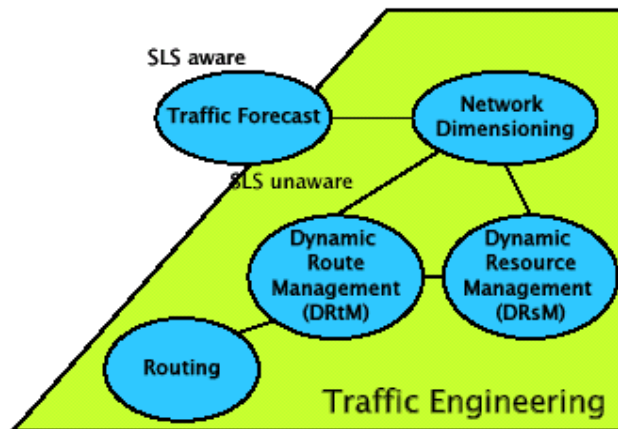


Figura 4: Módulos do Bloco de Engenharia de Tráfego

Na abordagem de engenharia de tráfego MPLS são considerados dois tipos de planejamento: TE-MPLS em longo prazo (dias – semanas) e TE-MPLS em curto prazo (minutos – horas). Esse planejamento é realizado sobre uma previsão de demanda e sobre uma configuração de rede, sujeita à falhas. Portanto, é necessário que o módulo de dimensionamento de rede seja capaz de rapidamente computar novas LSPs, quando por algum motivo as existentes não forem mais viáveis. Segundo [Goderis et al., 2001] o módulo de dimensionamento de rede é centralizado, apesar de serem possíveis implementações descentralizadas.

O módulo de previsão de tráfego (*Traffic Forecast* - TF) estima uma demanda de tráfego orientada aos SLS (*SLS aware*) e em relação a informações obtidas do bloco de monitoramento de rede (*SLS unaware*). Esta demanda é representada por uma matriz de troncos de tráfego, agrupados por agregados de ordem.

O módulo gerenciador dinâmico de rotas (*Dynamic Route Management* – DRtM) utiliza as rotas explícitas (LSPs) geradas pelo

² Na abordagem proposta com Diffserv e MPLS.

módulo de dimensionamento de rede para configurar a estrutura de rede para atender ao tráfego previsto.

O módulo gerenciador dinâmico de recursos (*Dynamic Resource Management* – DRsM) é responsável pelo controle dos recursos da rede. Sua interação com o módulo de dimensionamento de rede se faz informando sobre os recursos disponíveis para alocação dos troncos de tráfego ou notificando a necessidade de recalcular as rotas por alguma alteração na disponibilidade da rede como, por exemplo, a falha de um enlace na rede.

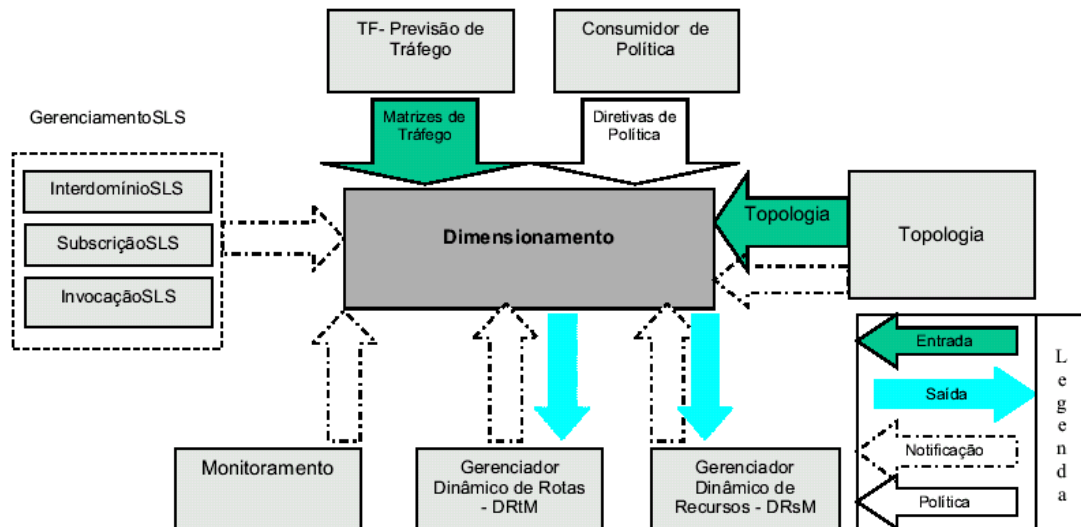


Figura 5: Interface do módulo de dimensionamento de rede

O fluxo de dados entre o módulo de dimensionamento e os outros módulos da arquitetura podem ser visualizados na Figura 5.

Formulação do Problema

Uma rede é modelada como um grafo direcionado $G(V,A)$, onde V é um conjunto de nós, representando os roteadores, e A é o conjunto das arestas, os enlaces da rede. Um elemento $a_{ij} \in A$ é especificado pelo par $l=(v_{i,ingress}, v_{j,egress})$, onde $v_{i,ingress}$ e $v_{j,egress}$ são os nós onde o tráfego entra e sai na rede respectivamente, também denominados nós de ingresso e egresso.

Sendo uma classe de qualidade de serviço h implementada pelo PHB do tipo h , temos que cada enlace l possui os seguintes recursos associados com o PHB do tipo h :

1. A capacidade do enlace C^h_l ;
2. O atraso do enlace d^h_l ;

O atraso total do enlace pode ser decomposto em quatro componentes distintos: processamento, enfileiramento, transmissão e propagação [Bertsekas & Gallager, 1992]. Dependendo das características associadas a um PHB, o atraso assumido pode ser determinístico ou probabilístico, máximo ou médio. Por exemplo, considere os serviços implementados pelos comportamentos dos tipos EF e AF. Um serviço implementado pelo comportamento do tipo EF, o atraso máximo deve ser determinístico e limitado a um valor máximo. Diferentemente, um serviço implementado pelo comportamento do tipo AF, o atraso pode assumir um valor esperado, isto é, com a maior probabilidade de acontecer. Geralmente, a capacidade C^h_i é expressa em Mbps e o atraso d^h_i em milisegundos.

Um tronco de tráfego $TT^h_i \in OA^h$, com nó de ingresso $v_{TT,ingress}$ e o conjunto de nós de egresso $V_{TT,egress}$, sendo $V_{TT,egress} \subseteq (V \setminus V_{TT,ingress})$, é associado a requisitos de banda de passagem de ingresso mínima e máxima, denotadas respectivamente por $b_{TT,ingress}$, $\overline{b_{TT,ingress}}$ e com um atraso fim-a-fim máximo $d_{TT,i}$. O atraso máximo pode ser determinístico ou probabilístico, dependendo do tipo do PHB h . Para cada nó de egresso $v_{TT,y} \in V_{TT,egress}$, a banda de passagem requisitada de egresso é definida como $b_{TT,y}$.

Deve-se determinar no grafo G os caminhos para cada par de nós ingresso-egresso $(v_{TT,ingress}, v_{TT,egress})$ dos troncos de tráfego TT^h_i , atendendo aos requisitos, objetivando minimizar a utilização da banda de passagem em todos os enlaces.

Um caminho é um conjunto ER (rota explícita), composto por enlaces $l=(v_{i,ingress}, v_{j,egress})$, portanto $ER \subseteq A$. Um conjunto de caminhos com o mesmo nó inicial de ingresso $(v_{TT,ingress})$, para um PHB h de qualidade de serviço, define uma árvore T^h_{TT} , cuja raiz é o nó de ingresso.

A abordagem do modelo funil é utilizada para determinar de modo eficiente os caminhos para os troncos de tráfego, proporcionando uma melhor utilização dos enlaces do grafo.

Portanto, defini-se o problema como: determinar uma árvore T^h_{TT} para cada tronco de tráfego $TT^h_i \in OA^h$, de modo que estas árvores atendam aos requisitos de cada TT^h_i e requisitos de largura de banda disponível em cada enlace C^h_i objetivando minimizar a utilização dos enlaces em todo o grafo.

Implementação Atual

Considera-se como a implementação atual do Módulo de Dimensionamento de Rede proposta por Cavalcanti em [Cavalcanti, 2000]. A implementação está dividida em duas partes, como ilustrado na Figura 6. Uma responsável por prover a interface com os outros módulos da arquitetura, tratamento das diretivas de política e preparar os dados para o núcleo do módulo. O núcleo

realiza a alocação dos troncos de tráfego, para um determinado agregado de ordem, na topologia de rede disponível e retorna as respectivas rotas explícitas, os LSPs.

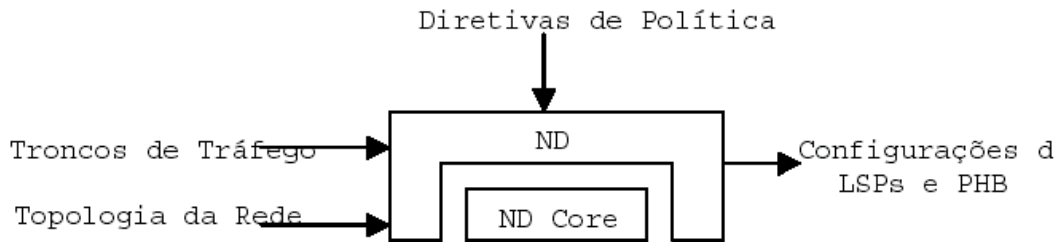


Figura 6: Implementação atual do módulo de dimensionamento

O núcleo utiliza um algoritmo de duas fases. Uma fase de alocação com uma heurística construtiva e uma fase de realocação para eliminar possíveis inviabilidades geradas na fase de alocação. É considerada como uma inviabilidade alguma sobrecarga na utilização de um enlace, ou seja, a soma de todos os troncos de tráfegos alocados em um enlace superar a capacidade deste enlace.

Construção de uma solução para o problema

Na fase de alocação é gerada a solução inicial a partir da alocação individual de cada tronco de tráfego na rede, obedecendo a uma ordenação decrescente em relação ao somatório das bandas de passagem dos nós egressos. A alocação de cada tronco de tráfego é um procedimento heurístico baseado em caminhos mínimos entre os nós de ingresso e egresso do fluxo.

Os caminhos de um tronco de tráfego entre os nós de ingresso e egresso são caminhos mínimos na rede e são gerados individualmente para cada dupla. Um pré-processamento na rede é realizado para os arcos que não possuem banda de passagem disponível para atender aos requisitos não sejam utilizados. Na inexistência de um caminho que atenda aos requisitos entre os nós de ingresso e egresso são consideradas relaxações no estado presente da rede, como considerar os recursos originais da topologia, para se determinar uma nova alocação.

No caso da topologia original de rede não comportar algum tronco de tráfego por causa dos requisitos de QoS do agregado de ordem, é gerada uma exceção, sinalizando a inviabilidade na alocação da demanda de tráfego naquela topologia de rede.

No modelo tubo esses caminhos já são a solução. O uso do modelo funil nesta abordagem consiste em considerar fluxos de um tronco de tráfego em um mesmo enlace sendo apenas um, alocando o menor valor entre a capacidade do nó de ingresso e o somatório das capacidades dos nós egressos dos fluxos analisados.

Restaurando inviabilidades

Caso exista algum arco com alocação maior que sua capacidade então é realizado um procedimento para tentar realocar os troncos de tráfego em outros arcos. Os arcos com as maiores inviabilidades são tratados primeiro sendo apenas uma otimização para cada um a fim de evitar situações de computação infinita como ficar realocando troncos de tráfego entre dois arcos sobrecarregados, cada momento em um arco.

A otimização de um arco consiste em determinar o menor tronco de tráfego, que se realocado, elimina a inviabilidade do arco. Se a sobrecarga for maior do que a maior capacidade alocada no arco ou a realocação deste tronco de tráfego não for possível, é aplicado outro método onde se tenta realocar todos os troncos de tráfego que utilizem o arco. Uma realocação é aceita caso a nova banda de passagem utilizada pelo tronco de tráfego for menor do que a alocação prévia na rede.

Ainda é considerado que o algoritmo de dimensionamento poderá indicar a necessidade de alocar mais recursos que a capacidade corrente do enlace, sendo sempre uma decisão de política tratar esta exceção, que é rejeitar o(s) tronco(s) de tráfego que causaram esta demanda ou aumentar os recursos dos enlaces [Cavalcanti, 2000].

Método Proposto

Conforme descrito nas notas de aula do Prof. Marcone [Souza, 2002], as metaheurísticas são métodos de busca local destinados a encontrar uma boa solução, eventualmente a ótima, consistindo na aplicação, em cada passo, de uma heurística subordinada, a qual tem que ser modelada para cada problema específico.

Contrariamente às heurísticas convencionais, as metaheurísticas são de caráter geral e têm condições de escapar de ótimos locais.

As metaheurísticas, assim como os métodos de busca local tradicionais, diferenciam-se entre si basicamente pelas seguintes características:

- a) critério de escolha de uma solução inicial;
- b) definição da vizinhança $N(s)$ de uma solução s ;
- c) critério de seleção de uma solução vizinha dentro de $N(s)$;
- d) critério de término;

A metaheurística GRASP (*Greedy Randomized Adaptive Search Procedure*) é um método iterativo proposto por [Feo & Resende, 1995] que consiste de duas fases: uma fase de construção na qual uma solução é gerada elemento a elemento e de uma fase de busca local, na qual um ótimo local na vizinhança da solução

construída é pesquisado. O método proposto é baseado no método atual e utiliza muito dos algoritmos implementados.

Construção da Solução Inicial

Foram implementadas duas metodologias onde a segunda é um método complementar para refinar as soluções geradas pela fase de construção GRASP.

Fase de construção GRASP

A heurística utilizada na fase de construção é a mesma da implementação atual com uma adaptação que permite gerar diferentes alocações. A cada iteração da fase de construção, os troncos de tráfego ainda não alocados de um mesmo agregado de ordem, são colocados em uma lista C de candidatos, seguindo um critério de ordenação relativo à soma das bandas de passagem dos nós egressos. A componente probabilística do procedimento reside no fato de que cada elemento é selecionado de forma aleatória a partir de um subconjunto restrito formado pelos melhores elementos da lista de candidatos, a lista de candidatos restrita (LRC). O tamanho da LRC é definido segundo o fator α onde temos a relação $||LRC|| = \alpha * ||LC||$. O procedimento de alocação de um tronco de tráfego é o mesmo apresentado em Implementação Atual. Essa técnica de escolha permite que diferentes soluções sejam geradas em cada execução deste procedimento de construção GRASP.

Para valores do fator α próximos a um temos escolhas aleatórias dos candidatos na composição da solução. Conseqüentemente para valores próximos a zero produzem soluções mais refinadas propiciando uma convergência rápida das soluções geradas pelo método.

Pseudocódigo do procedimento:

```

Procedimento First_Allocate_Grasp ( $\alpha$ ,  $g(\cdot)$ ,  $\oplus$ )
1    $s \leftarrow \emptyset$ ;
2   Inicialize o conjunto de candidatos LC;
3   enquanto (  $LC \neq \emptyset$  ) faça
4        $t_{\min} \leftarrow \min\{g(t) \mid t \in LC\}$ ;
5        $t_{\max} \leftarrow \max\{g(t) \mid t \in LC\}$ ;
6        $LRC \leftarrow \{ t \in LC \mid g(t) \leq t_{\min} + \alpha(t_{\max} - t_{\min}) \}$ ;
7       Selecione, aleatoriamente, um elemento  $t \in LRC$ ;
8        $s \leftarrow s \oplus t$ ;
9        $LC \leftarrow LC / \{t\}$ ;
10  fim enquanto;
11  Retorne  $s$ ;
fim First_Allocate_Grasp;

```

GRASP com Filtro

A eficiência da busca local depende da qualidade da solução construída. O procedimento de construção da solução inicial tem então um papel importante na busca local, uma vez que as soluções construídas constituem bons pontos de partida para busca local, permitindo assim não induzir a ótimos locais e podendo acelerá-la. Com esta premissa propõe-se aplicar o método de construção da solução inicial diversas vezes e considerar apenas a melhor dessas soluções iniciais, operando como um filtro de soluções.

Nesta abordagem é difícil determinar o custo-benefício entre o número de soluções a considerar e a variabilidade do critério de aleatoriedade da solução. Como a fase de construção da solução inicial deve ser eficiente e rápida, não foram consideradas muitas iterações para determinar uma solução para a fase de refinamento que é mais demorada. Existem duas abordagens para a escolha do fator α : a primeira empírica baseada na experiência do operador do sistema e a segunda computacional baseada no método conhecimento como GRASP Reativo.

Pseudocódigo do procedimento:

```

Procedimento Grasp_Filter ( $\alpha$ , maxFilter, fa(.))
1    $f^* \leftarrow \infty$ ;
2   para  $i$  de 1 até maxFilter faça
3      $s \leftarrow \text{First\_Allocate\_Grasp}(\alpha_{\text{grasp}})$ ;
4     se  $fa(s) < f^*$  então
5        $s^* \leftarrow s$ ;
6        $f^* \leftarrow fa(s)$ ;
7     fim se;
8   fim para;
9   Retorne  $s^*$ ;
fim Grasp_Filter;

```

Função de Avaliação

A fim de analisar as soluções existentes utilizamos uma função de avaliação hierárquica de dois níveis. No primeiro nível é computada a taxa média de utilização nos enlaces sobrecarregados e no segundo nível a taxa média de utilização em todos os enlaces da rede.

Uma função de avaliação hierárquica é composta de diversos níveis que para comparação entre duas soluções é considerada melhor a solução que apresentar melhor valor no nível superior e em caso de empate recorre-se aos níveis inferiores. Sendo assim não é gerado um único valor resultado de um agregado a partir das diferentes análises, podendo causar distorções nas comparações, mas compreendem-se diferentes valores analisados separadamente segundo uma ordem de importância. Com esta função de avaliação pode-se acrescentar critérios de qualidade sem perder o foco principal no refinamento da solução.

Refinamento da Solução

Estudos preliminares

O método de refinamento da solução foi concebido de modo a utilizar ao máximo das estruturas e algoritmos existentes na implementação atual. Um primeiro método de refinamento foi proposto utilizando a heurística conforme em implementação atual, adaptando-a para aceitar apenas a realocação que obteve a maior melhora para um tt_i de um enlace sobrecarregado. Podendo ser realizadas várias otimizações sobre o mesmo enlace sobrecarregado, mas respeitando que um mesmo enlace não pode ser otimizado outra vez antes que todos os enlaces sobrecarregados tenham sido avaliados, sendo o critério de parada um tempo pré-determinado. Nos testes, os valores da função de avaliação para as soluções obtidas pioraram em relação aos da solução inicial, portanto este critério foi desconsiderado. Isso deve ao fato de que o critério de escolha era a demanda utilizada pelo tt_i , que tinha uma tendência para diminuir em cada interação, porém, a avaliação da solução é calculada principalmente a partir dos enlaces sobrecarregados na solução.

Para se determinar a melhor realocação passou-se a considerar a que causou maior melhora no resultado da função de avaliação. Resumidamente a idéia é selecionar os arcos sobrecarregados em ordem decrescente em relação à taxa de utilização e tentar realocar cada tt_i que utiliza este arco, alterando o grafo de modo a desprestigiar sua utilização nas novas alocações. Aceitar a melhor realocação encontrada, medida pela função de avaliação, caso ela melhore o resultado anterior ao movimento. Iterativamente selecionando um novo arco pelo mesmo critério, exceto, por considerar que para um arco sobrecarregado ser escolhido novamente, os demais devem ter sido escolhidos pelo menos o mesmo número de vezes. A parada deste procedimento é determinada por tempo ou quando não houver mais sobrecargas.

Outras estratégias foram concebidas analisando os troncos de tráfego como, por exemplo, selecionar os troncos de tráfego que utilizam os arcos mais sobrecarregados e tentar realocá-los, sem utilizar estes arcos, aceitando apenas melhoras na função de avaliação. Pode-se selecionar cada tronco de tráfego mais de uma vez, sendo o critério de parada o tempo ou quando não houver mais sobrecargas e garantindo justiça para que todos possam ser escolhidos. Essa nova abordagem deu origem à heurística de descida proposta e que se demonstrou de maior eficácia, dentre os métodos analisados, para refinamento da solução.

Método de descida

É um método de busca local que se caracteriza por analisar todos os possíveis vizinhos de uma solução s em sua vizinhança $N(s)$, escolhendo, a cada passo, aquele que tem o menor valor para a função objetivo. Nesse método, o vizinho candidato somente é aceito se ele melhorar estritamente o valor da melhor solução até então obtida. Dessa forma, o método pára tão logo um mínimo local seja encontrado. [Souza, 2002]

Na estrutura de vizinhança considerada para busca local um vizinho de uma solução é obtido com uma realocação de um tronco de tráfego tt_i através do procedimento de alocação considerando o estado momentâneo da rede. Uma realocação consiste em desalocar e alocar novamente na rede, sendo esta alocação realizada pelo método descrito em Implementação Atual, e por isto é adaptativa ao estado momentâneo da rede durante a busca.

$$N(s) \xrightarrow{\text{realocar}TT_i} s'$$

O refinamento da solução é realizado realocando cada tronco de tráfego e aceitando o vizinho de maior melhora segundo a função de avaliação. Quando não houver vizinhos que melhorem a solução está encerrada a busca.

Pseudocódigo do procedimento:

```

Procedimento MétodoDeDescida (fa(.), N(.), s)
1  repita
2    s' ← Selecione a melhor solução em N(s);
3    se ( fa(s') < fa(s) ) então
4      s ← s';
5  até s ≠ s' ;
6  Retorne s;
fim;

```

A paralelização deste algoritmo pode ser feita considerando um particionamento do conjunto de troncos de tráfegos a serem analisados. Deste modo podemos construir uma versão onde possa utilizar processamento distribuído para um grande conjunto de troncos de tráfego obtendo uma melhora linear. Um outro meio de melhorar o desempenho do método é considerar vizinhança parcial como, por exemplo, considerar apenas os troncos de tráfego que utilizem enlaces sobrecarregados.

Simulated Annealing

Trata-se de uma técnica de busca local probabilística, proposta originalmente por [Kirkpatrick et al., 1983], que se fundamenta em uma analogia com a termodinâmica, ao simular o resfriamento de um conjunto de átomos aquecidos, operação conhecida como recozimento.

A solução inicial considerada é gerada pela fase de construção GRASP. O procedimento principal consiste de uma repetição que gera aleatoriamente, em cada iteração, um único vizinho s' da solução corrente s .

Chamando de Δ a variação de valor da função objetivo ao mover-se para uma solução vizinha candidata, isto é, $\Delta=f(s')-f(s)$, o método aceita o movimento, e a solução vizinha passa a ser a nova solução corrente, se $\Delta < 0$. Caso $\Delta = 0$ a solução vizinha candidata também poderá ser aceita, mas neste caso, com uma probabilidade $e^{-\Delta/T}$, onde T é um parâmetro do método, chamado de temperatura e que regula a probabilidade de se aceitar soluções de pior custo. A temperatura T assume, inicialmente, um valor elevado T_0 . Após um número fixo de iterações (o qual representa o número de iterações necessárias para o sistema atingir o equilíbrio térmico em uma dada temperatura), a temperatura é gradativamente diminuída por uma razão de resfriamento α , tal que $T_n \leftarrow \alpha \times T_{n-1}$, sendo $0 < \alpha < 1$. Com esse procedimento, dá-se, no início uma chance maior para escapar de mínimos locais e, à medida que T aproxima-se de zero, o algoritmo comporta-se como o método de descida, uma vez que diminui a probabilidade de se aceitar movimentos de piora ($T \rightarrow 0 \rightarrow e^{-\Delta/T} \rightarrow 0$).

O procedimento pára quando a temperatura chega a um valor próximo de zero e nenhuma solução que piore o valor da melhor solução é mais aceita, isto é, quando o sistema está estável. A solução obtida quando o sistema encontra-se nesta situação evidencia o encontro de um mínimo local.

Os parâmetros de controle do procedimento são a razão de resfriamento α , o número de iterações para cada temperatura (SMax) e a temperatura inicial T_0 .

Com a estrutura de vizinhança definida anteriormente, a função de avaliação e utilizando as mesmas funções de realocação definiu-se um Simulated Annealing para refinamento de uma solução para o problema. Nos testes iniciais para calibrar os parâmetros verificou-se que após poucos movimentos aleatórios a solução melhorava um pouco. Entretanto após as realocações aceitas pela função probabilística por serem de piora da solução, determinaram ótimos locais com solução pior do que a solução inicial. Acredita-se que pelo fato da estrutura de vizinhança se basear em um movimento baseado em uma heurística o método se tornou instável e os resultados finais obtidos não foram satisfatórios. Pelo tipo de convergência das soluções e instabilidade na busca conclui-se que a estrutura de vizinhança não é adequada para o método.

Resultados Computacionais

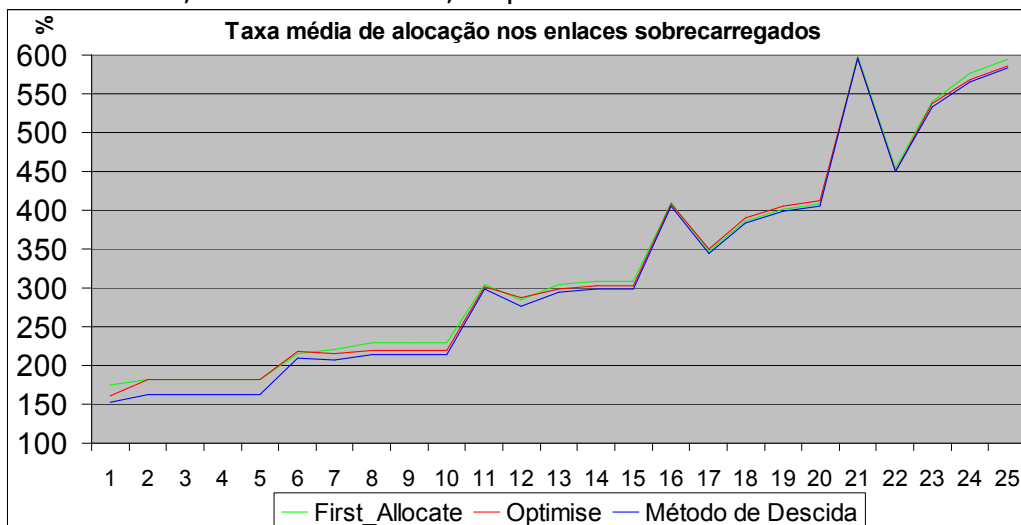
Os dois métodos foram submetidos a um conjunto de testes construídos conforme a metodologia descrita em [Cavalcanti, 2000]. Foram consideradas redes com 16, 32 e 64 vértices com diferentes graus de conectividades. A geração dos troncos de tráfego é realizada por um procedimento que considera uma série de valores e nos modelos tubo e funil. Com este processo pode-se analisar o desempenho de algoritmos em diversas situações.

A fim de avaliar os dois métodos de refinamento a construção da solução inicial foi realizada pelo mesmo procedimento denominado *First_Allocate*, conforme descrito em Implementação Atual. O método representando a implementação atual é denominado *Optimise* e o método proposto de *Método de Descida*.

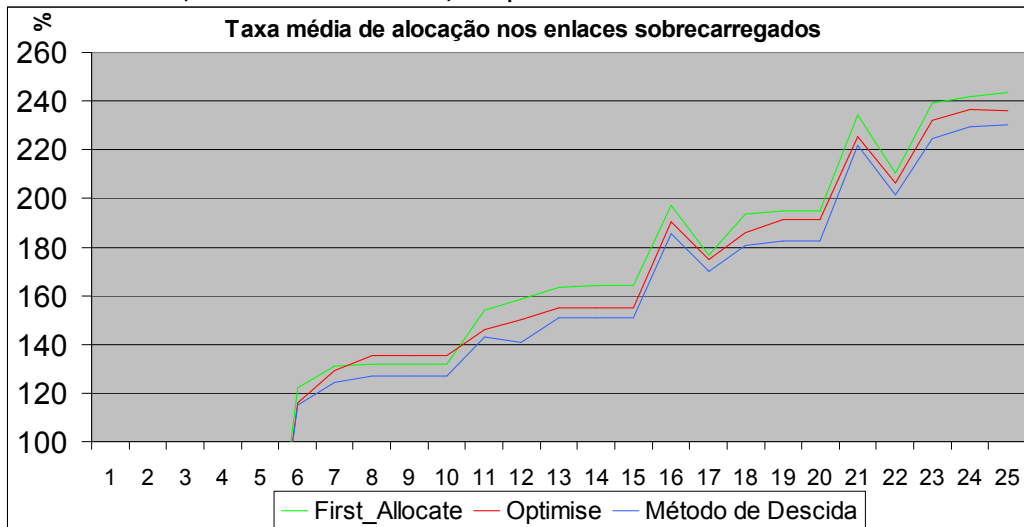
No teste com 16 vértices, das 100 instâncias testadas, 75 apresentaram enlaces sobrecarregados. Comparando os resultados obtidos nas fases de refinamento, segundo a função de avaliação, temos que o método proposto foi melhor em 98% dos casos. Nas instâncias com inviabilidade obteve-se uma melhora média da taxa de ocupação dos enlaces sobrecarregados de 7,88% com desvio padrão de 7,10 e moda 20,44% em relação ao método de refinamento da implementação atual.

Os gráficos representam os valores da função de avaliação em relação às soluções obtidas pelos métodos. Os gráficos são relativos as quatro topologias geradas, que possuem diferentes graus de conectividade. São considerados 16 troncos de tráfego no modelo funil e 240 no modelo tubo.

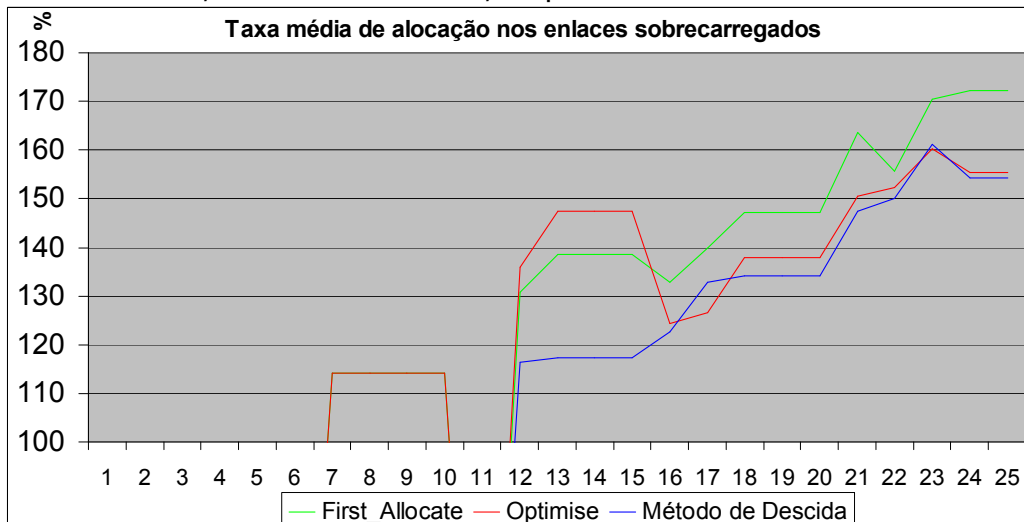
#Arcos = 70, Grau médio = 4, Capacidade total = 552 MB



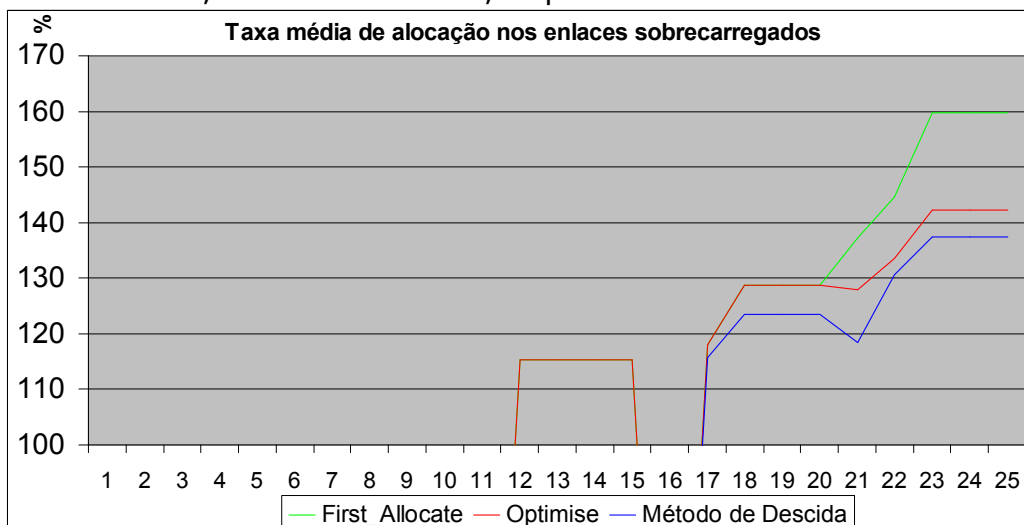
#Arcos = 98, Grau médio = 6, Capacidade total = 994 MB

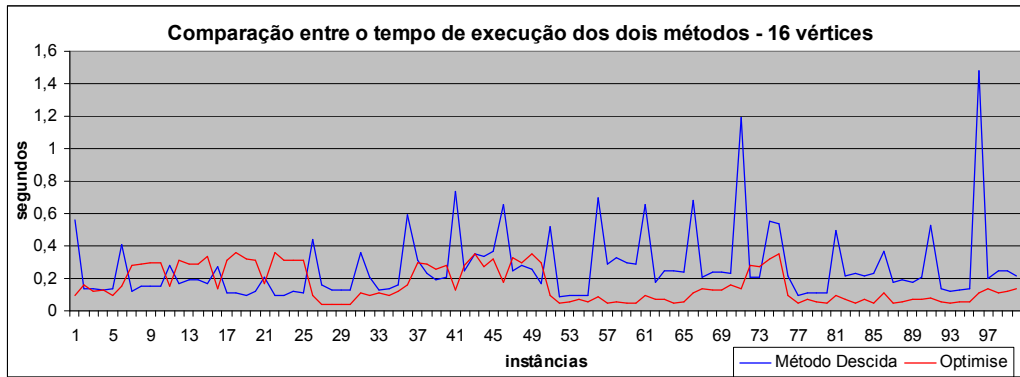


#Arcos = 172, Grau médio = 10, Capacidade total = 1488 MB

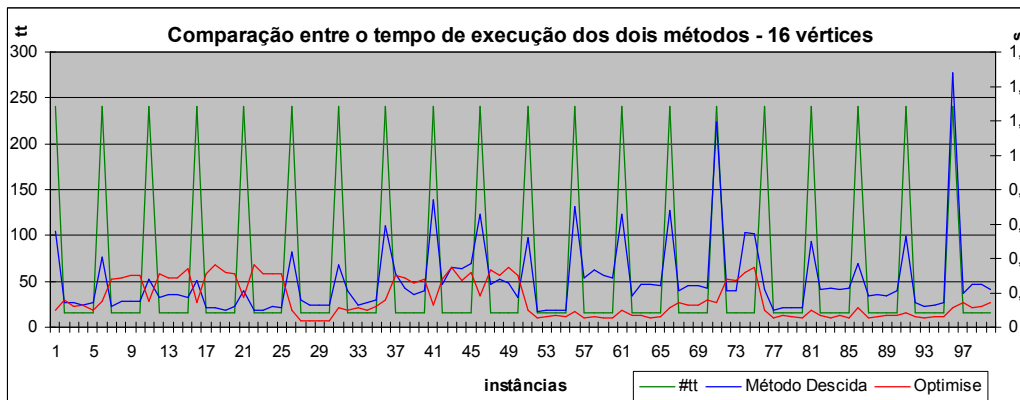


#Arcos = 218, Grau médio = 13, Capacidade total = 1916 MB





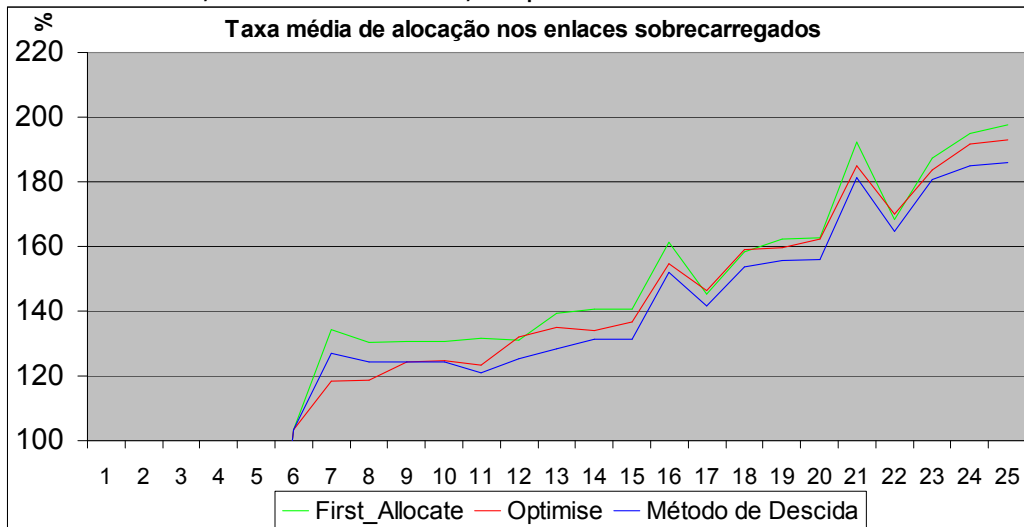
Os picos nos tempos de execução são exatamente nas instâncias no modelo tubo, visto que a estrutura de vizinhança cresce consideravelmente. O tempo de execução no *Método de Descida* tende a aumentar nas instâncias com maior grau de conectividade, fator relacionado com a realocação dos troncos de tráfego. O *Optimise* obteve uma execução mais rápida nas últimas instâncias, pois nestas instâncias tiveram poucos enlaces sobrecarregados.



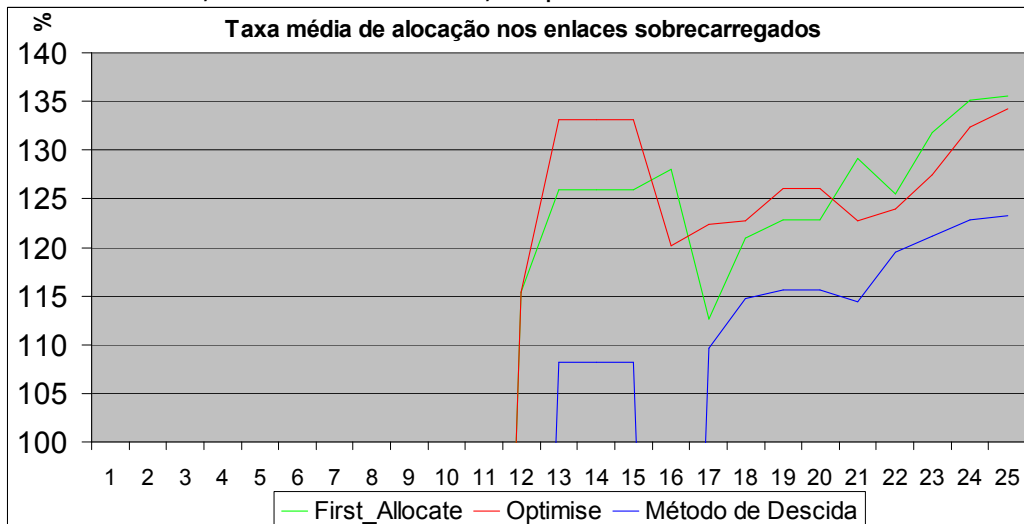
No teste com 32 vértices, das 100 instâncias testadas, 43 apresentaram enlaces sobrecarregados. Comparando os resultados obtidos nas fases de refinamento, segundo a função de avaliação, observa-se que o método proposto foi melhor em 98% dos casos. Nas instâncias com inviabilidade obteve-se uma melhora média da taxa de ocupação dos enlaces sobrecarregados de 7,99% com desvio padrão de 6,99 e moda 24,91% em relação ao método de refinamento da implementação atual.

Os gráficos representam os valores da função de avaliação em relação às soluções obtidas pelos métodos. Os gráficos são relativos às quatro topologias geradas, que possuem diferentes graus de conectividade. São considerados 32 troncos de tráfego no modelo funil e 992 no modelo tubo.

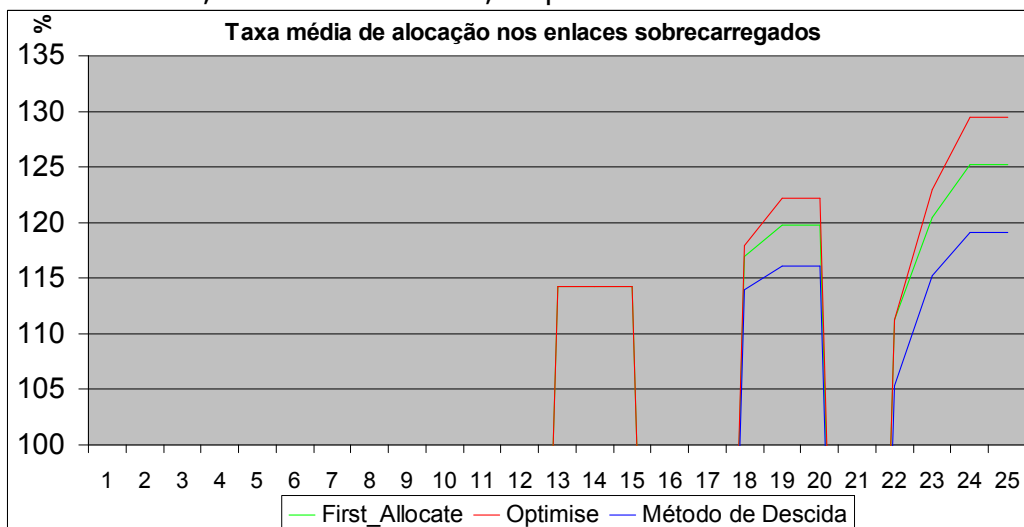
#Arcos = 302, Grau médio = 9, Capacidade total = 5022 MB



#Arcos = 488, Grau médio = 15, Capacidade total = 9008 MB



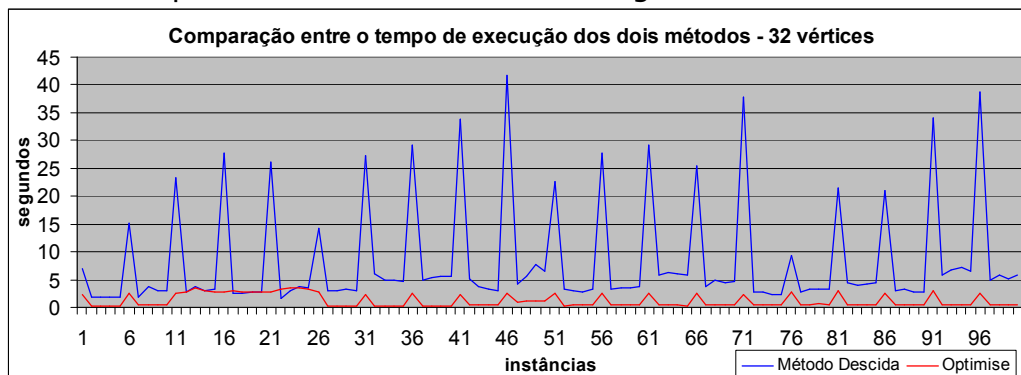
#Arcos = 688, Grau médio = 21, Capacidade total = 12392 MB



#Arcos = 894, Grau médio = 27, Capacidade total = 16064 MB

Nenhuma instância apresentou enlaces sobrecarregados.

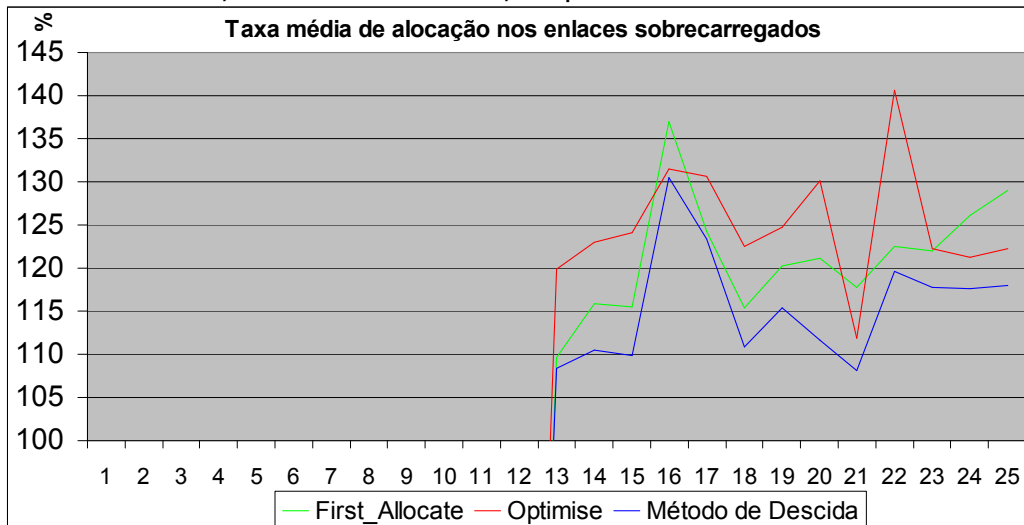
Uma importante relação entre os métodos é que o *Método de Descida* somente pára quando todos os critérios de qualidade não puderem ser melhorados, e a cada busca é feita a realocação de todos os troncos de tráfego, enquanto o *Optimise* só é executado se existir enlaces sobrecarregados e somente analisa os troncos de tráfego que utilizem este enlace. O *Método de Descida* nunca irá piorar a solução inicial enquanto o *Optimise*, pelo seu critério de aceite em uma realocação, pode piorar na média as sobrecargas na rede como podemos observar em muitos gráficos.



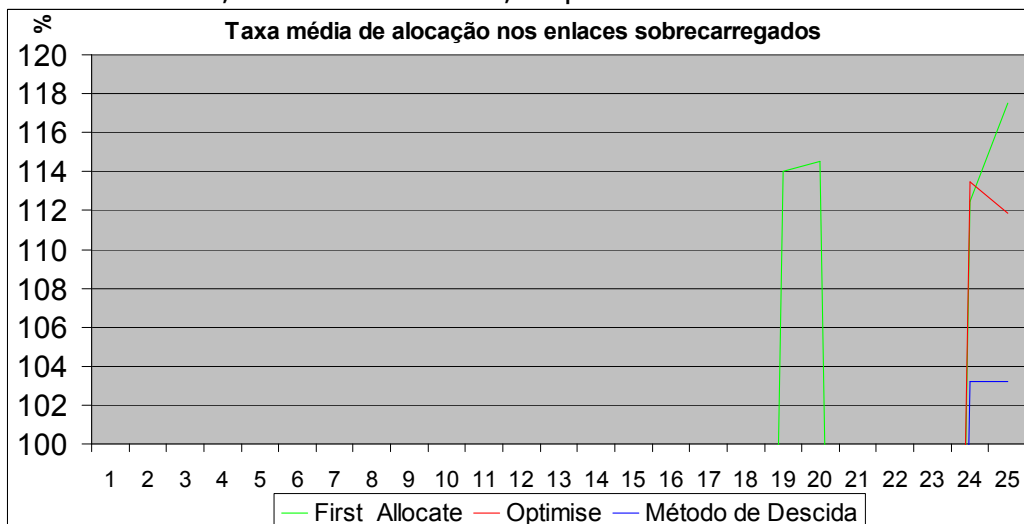
No teste com 64 vértices, das 100 instâncias testadas, 19 apresentaram enlaces sobrecarregados. Comparando os resultados obtidos nas fases de refinamento, segundo a função de avaliação, observa-se que o método proposto também foi melhor em 98% dos casos. Nas instâncias com inviabilidade obteve-se uma melhora média da taxa de ocupação dos enlaces sobrecarregados de 9,17% com desvio padrão de 8,26 e sem moda em relação ao método de refinamento da implementação atual.

Os gráficos representam os valores da função de avaliação em relação às soluções obtidas pelos métodos. Os gráficos são relativos as quatro topologias geradas, que possuem diferentes graus de conectividade. São considerados 64 troncos de tráfego no modelo funil e 4032 no modelo tubo.

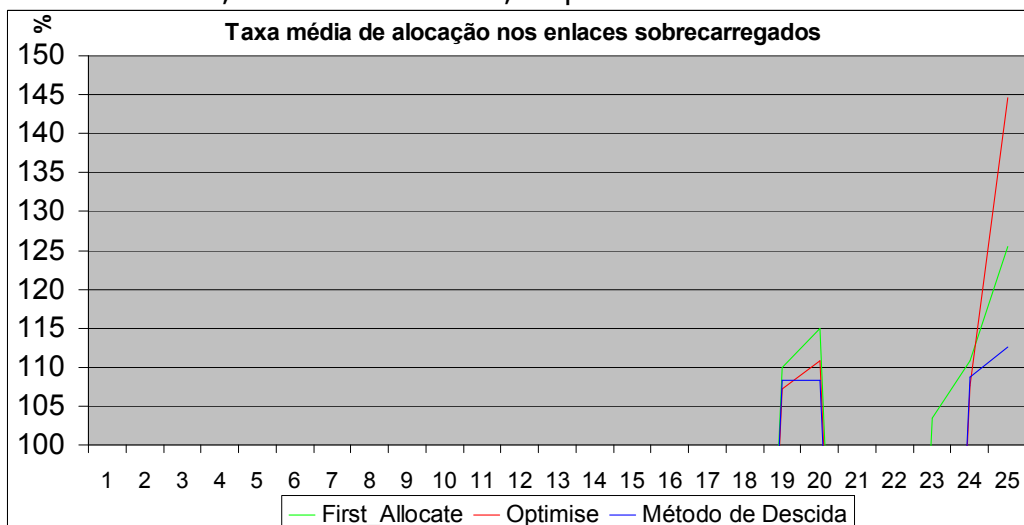
#Arcos = 1208, Grau médio = 18, Capacidade total = 37936 MB



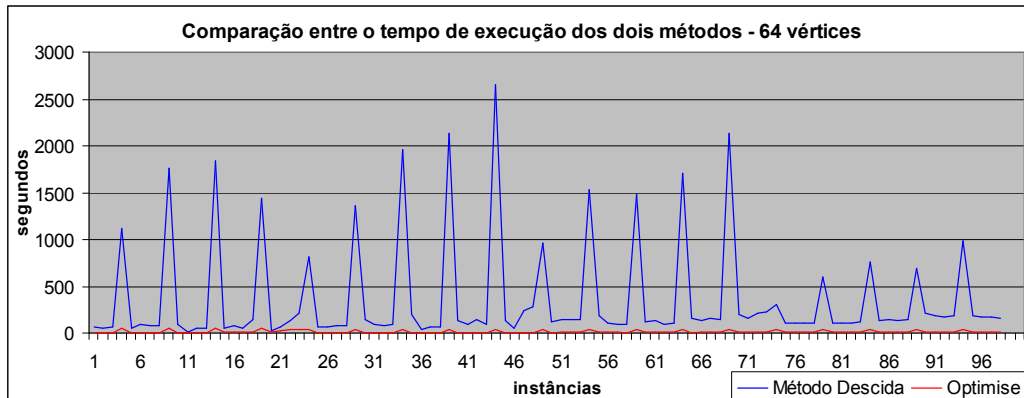
#Arcos = 2034, Grau médio = 31, Capacidade total = 69764 MB



#Arcos = 2872, Grau médio = 44, Capacidade total = 89008 MB



#Arcos = 3674, Grau médio = 57, Capacidade total = 113040 MB
Nenhuma instância apresentou enlaces sobrecarregados.



Conclusões

Neste trabalho foram analisados diversos aspectos da nova geração da Internet, sendo um dos maiores desafios garantir qualidade no serviço prestado. Foi visto que há necessidade de uma arquitetura com controle sobre a rede e tráfego para viabilizar uma solução com garantias. Contextualizado na arquitetura Tequila, este problema é tratado primordialmente no módulo de dimensionamento de redes onde foi realizado um estudo detalhado sobre sua implementação atual, e proposto um novo método baseado na metaheurística GRASP.

Este trabalho aborda um problema combinatorial complexo e que pode ser entendido como dois problemas distintos. O problema de roteamento com multi-restrições em QoS (QoS_R), que é o problema de alocar um único tronco de tráfego na rede, e o segundo sendo o problema do dimensionamento de redes em si, que lida com as diversas possibilidades de alocação dos troncos de tráfego na rede de modo que não haja sobrecargas nos enlaces. Abordagens heurísticas viabilizam a resolução prática deste problema devido a sua alta complexidade, mesmo sem oferecerem garantias de otimalidade.

Através da metaheurística GRASP foi possível propor um método simples e eficaz em relação à implementação atual. Nos resultados computacionais constatamos que há necessidade de uma pesquisa no sentido de diminuir o tempo de processamento gasto, principalmente em instâncias no modelo tubo. As soluções produzidas pelo método proposto foram melhores em 98% das instâncias analisadas, sendo que em 13 delas conseguiu remover as sobrecargas onde pelo refinamento através da implementação atual não foi possível.

Apesar da implementação atual tender a diminuir o número de enlaces sobrecarregados temos que se a inviabilidade for pequena

existem outras técnicas de redes, como filas, que podem contornar estas inviabilidades possibilitando o funcionamento da rede. Por isso o critério de avaliar a média da taxa de alocação dos enlaces sobrecarregados é mais significativo, e de melhor resultado na prática, do que o número de enlaces sobrecarregados. A utilização da função de avaliação hierárquica possibilita alterar os critérios de qualidade na avaliação das soluções e permite aos administradores do sistema configurar seus interesses diretamente na busca. Dentro da arquitetura Tequila pode-se até considerar a função de avaliação como sendo uma política.

Finalmente concluímos que o trabalho foi desenvolvido com êxito e ficam abertas diversas possibilidades de integração deste problema da área de redes de computadores tendo uma abordagem fundamentada em inteligência computacional.

Trabalhos Futuros

Implementação da análise de vizinhança parcial bem como outras técnicas que viabilizem uma execução mais veloz. Melhorar a formulação do problema permitindo uma formalização dos procedimentos. Definir novas instâncias para teste. Desenvolver novas métricas para avaliar as soluções. Novas abordagens heurísticas. Desenvolvimento de um modelo de programação matemática para avaliar as soluções geradas pelos métodos heurísticos.

Referências Bibliográficas

Awduche D., Malcolm J., Agogbua J., O'Dell M. e McManus J., "*Requirements for Traffic Engineering over MPLS*", IETF RFC 2702, Setembro 1999.

Bertsekas D., Gallager R., "*Data Networks*", Prentice Hall, 1992.

Blake S., Black D., Carlson M., Davies E., Wang Z., and Weiss W., "*An Architecture for Differentiated Services*", IETF Informational RFC 2475, Dezembro 1998.

Cavalcanti C.F.M.C., "Algoritmo de Dimensionamento de Rede", Tese de Doutorado, DCC/UFMG, 2000

Feo T.A. and Resende M.G.C.. "*Greedy randomized adaptive search procedures*". *Journal of Global Optimization*, 6:109-133, 1995

Garey M.S., Johnson D.S., "*Computers and Intractability: A Guide to the Theory of NP-Completeness*", W. H. Freeman, New York, 1979.

Goderis D. et al., "Service Level Specification Semantics, Parameters, and Negotiation Requirements", IETF Internet Draft: draft-tequila-sls-01.txt, Dec. 2001. Veja: <http://www.ist-tequila.org>. Acessado em Julho 2003

Heinanen J., Baker F., Weiss W., and Wroclawski J., "Assured Forwarding PHB Group", IETF Standards Track RFC 2597, Junho 1999.

Hwang F.K., Richards D.S., and Winter P., "The Steiner tree problem", North-Holland, Amsterdam, 1992.

Jacobson V., Nichols K., and Poduri K., "An Expedited Forwarding PHB", IETF Standards Track RFC 2598, Junho 1999.

Karp R.M., "Reducibility among combinatorial problems", Complexity of Computer Computations (E. Miller and J.W. Thatcher, eds.), 85-103, Plenum Press, New York, 1972.

Kirkpatrick S., Gellat D.C., e Vecchi M.P.. "Optimization by Simulated Annealing", Science, 220:671-680, 1983.

Kumar A., Rastogi R., Silberschatz A., e Yener B., "Algorithms for provisioning virtual private networks in the hose model", ACM Sigcomm, Cambridge, Massachusetts, USA, Agosto 2001.

Rabbat R., Laberteaux K., Modi N., and Kenney J., "Traffic Engineering Algorithms Using MPLS for Service Differentiation". International Conference on Communications IEEE ICC'2000, New Orleans, Junho 2000.

Rosen E., A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", IETF Standards Track RFC 3031, Janeiro 2001

Souza M. J. F. S., "Notas de aula da disciplina Inteligência Computacional para Otimização Segundo semestre 2002" Veja: www.decom.ufop.br/prof/marcone, Acessado em Novembro 2003.

Tjoens Y. et al., "Overview of external liason within (Clustering) and outside IST", Abril de 2000, Veja: <http://www.ist-tequila.org>, Acessado em Julho 2003

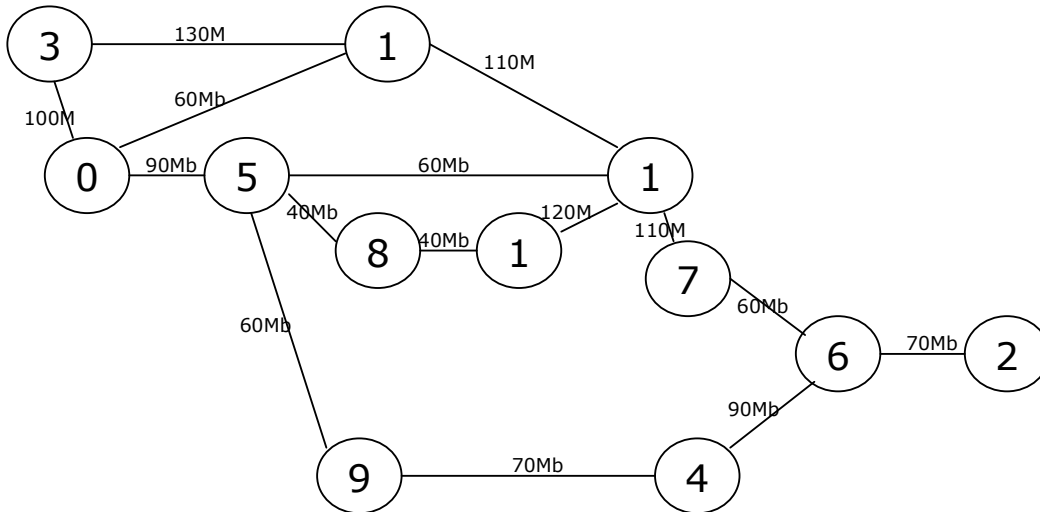
Trimintzios P., et al., "A Management and Control Architecture for Providing IP Differentiated Services in MPLS-based Networks", IEEE Communications Magazine, vol. 39, no. 5, Maio 2001

Wang Z. e Crowcroft J.. "*Quality-of-service routing for supporting multimedia applications*". IEEE Journal on Selected Areas in Communications, vol. 14, no. 7, pp. 1228–1234, Settembre 1996.

Apêndice

Exemplo de aplicação dos métodos

Topologia original:

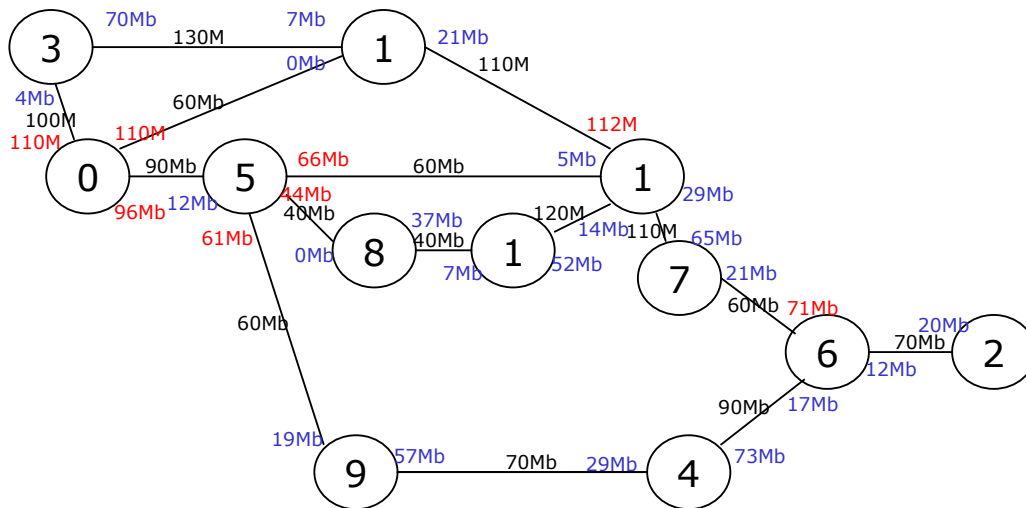


Troncos de Tráfego:

#	#id	in	eg	OA	in min	bw max	eg min	bw max	delay	loss	start	end
0	0	0	11	EF	20	20	20	20	10.0	0.1	0	0
1	0	11	EF	30	30	30	30	30	10.0	0.1	0	0
2	0	11	EF	40	40	40	40	40	10.0	0.1	0	0
3	0	11	EF	50	50	50	50	50	10.0	0.1	0	0
4	0	3	EF	20	20	20	20	20	10.0	0.1	0	0
5	3	11	EF	54	54	54	54	54	10.0	0.1	0	0
6	0	3	EF	53	53	33	33	33	10.0	0.1	0	0
6	0	11	EF	53	53	23	23	23	10.0	0.1	0	0
7	0	3	EF	52	52	52	52	52	10.0	0.1	0	0
7	0	11	EF	52	52	22	22	22	10.0	0.1	0	0
8	5	10	EF	55	55	35	35	35	10.0	0.1	0	0
8	5	11	EF	55	55	25	25	25	10.0	0.1	0	0
9	0	1	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	2	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	3	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	4	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	5	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	6	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	7	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	8	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	9	EF	20	20	1	1	1	10.0	0.1	0	0
9	0	10	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	0	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	2	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	3	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	4	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	5	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	6	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	7	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	8	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	9	EF	20	20	1	1	1	10.0	0.1	0	0
10	1	10	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	0	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	1	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	3	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	4	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	5	EF	20	20	1	1	1	10.0	0.1	0	0
11	2	6	EF	20	20	1	1	1	10.0	0.1	0	0

19	3	8	EF	20	20	1	1	10.0	0.1	0	0
19	3	9	EF	20	20	1	1	10.0	0.1	0	0
19	3	10	EF	20	20	1	1	10.0	0.1	0	0
20	4	1	EF	20	20	1	1	10.0	0.1	0	0
20	4	2	EF	20	20	1	1	10.0	0.1	0	0
20	4	3	EF	20	20	1	1	10.0	0.1	0	0
20	4	5	EF	20	20	1	1	10.0	0.1	0	0
20	4	6	EF	20	20	1	1	10.0	0.1	0	0
20	4	7	EF	20	20	1	1	10.0	0.1	0	0
20	4	8	EF	20	20	1	1	10.0	0.1	0	0
20	4	9	EF	20	20	1	1	10.0	0.1	0	0
20	4	10	EF	20	20	1	1	10.0	0.1	0	0
21	4	1	EF	20	20	1	1	10.0	0.1	0	0
21	4	2	EF	20	20	1	1	10.0	0.1	0	0
21	4	3	EF	20	20	1	1	10.0	0.1	0	0
21	4	5	EF	20	20	1	1	10.0	0.1	0	0
21	4	6	EF	20	20	1	1	10.0	0.1	0	0
21	4	7	EF	20	20	1	1	10.0	0.1	0	0
21	4	8	EF	20	20	1	1	10.0	0.1	0	0
21	4	9	EF	20	20	1	1	10.0	0.1	0	0
21	4	10	EF	20	20	1	1	10.0	0.1	0	0
22	5	0	EF	20	20	1	1	10.0	0.1	0	0
22	5	1	EF	20	20	1	1	10.0	0.1	0	0
22	5	2	EF	20	20	1	1	10.0	0.1	0	0
22	5	3	EF	20	20	1	1	10.0	0.1	0	0
22	5	4	EF	20	20	1	1	10.0	0.1	0	0
22	5	6	EF	20	20	1	1	10.0	0.1	0	0
22	5	7	EF	20	20	1	1	10.0	0.1	0	0
22	5	8	EF	20	20	1	1	10.0	0.1	0	0
22	5	9	EF	20	20	1	1	10.0	0.1	0	0
22	5	10	EF	20	20	1	1	10.0	0.1	0	0

First_Allocate



#Arcos: 30 #Sobrecarregados: 8

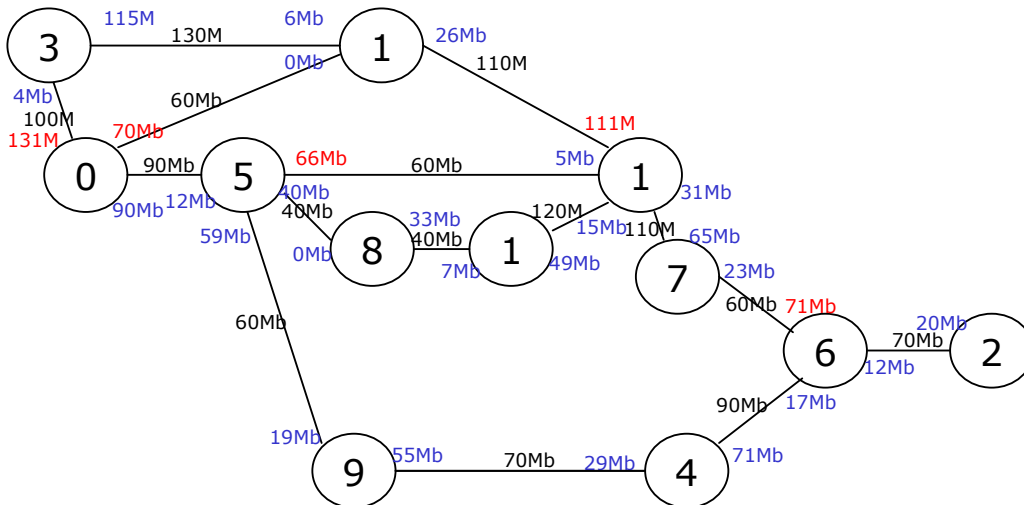
Taxa média de alocação nos enlaces sobrecarregados: 117,727%

Desvio Padrão: 27,037

Taxa média de alocação em todos enlaces: 54,383%

Desvio Padrão: 47,164

Optimise



#Arcos: 30 #Sobrecarregados: 5

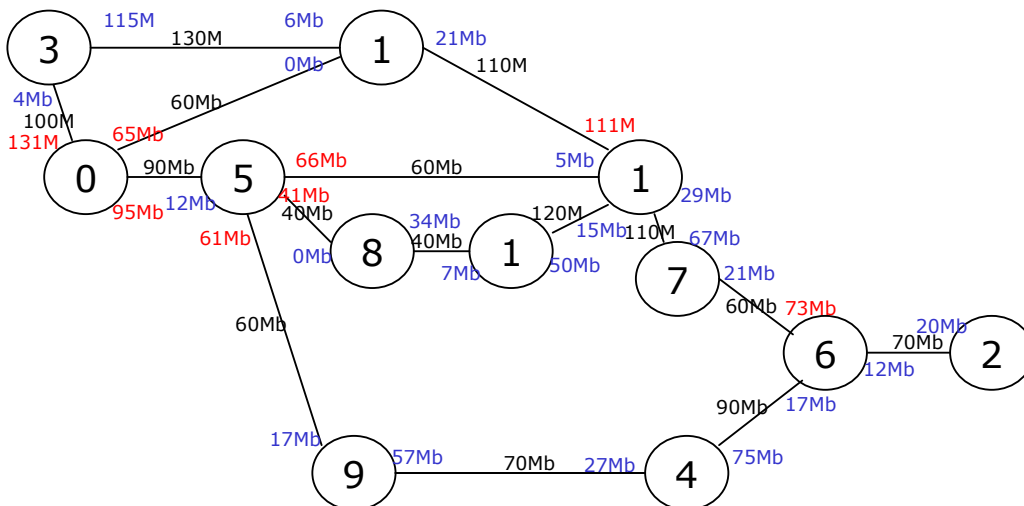
Taxa média de alocação nos enlaces sobrecarregados: 115,382%

Desvio Padrão: 11,099

Taxa média de alocação em todos enlaces: 53,057%

Desvio Padrão: 42,434

Método de Descida



#Arcos: 30 #Sobrecarregados: 8

Taxa média de alocação nos enlaces sobrecarregados: 110,204%

Desvio Padrão: 10,741

Taxa média de alocação em todos enlaces: 53,156%

Desvio Padrão: 43,176

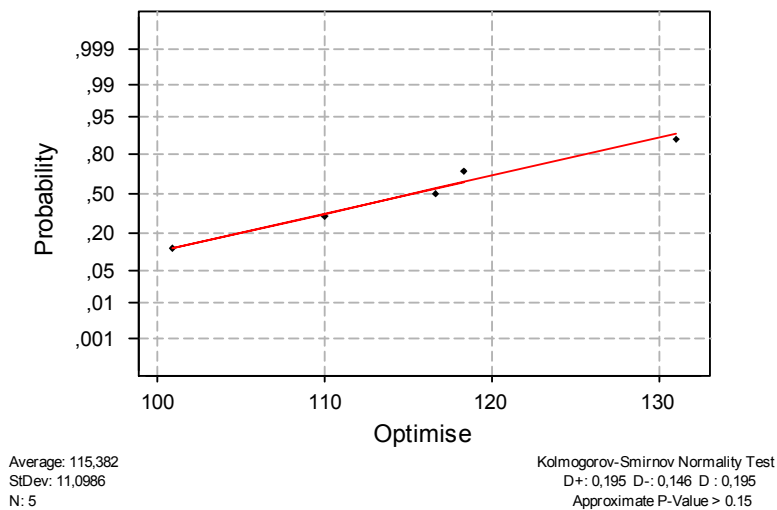
Este exemplo mostra o funcionamento dos dois métodos de refinamento em uma instância visualizável. Considerando o local de ocorrência dos enlaces sobrecarregados verifica-se que a maioria

deles representam gargalos de tráfego na rede, por exemplo, todos os arcos que saem do vértice 0 estão sobrecarregados. Uma diferença positiva a princípio para o método *Optimise* é a diminuição dos números de enlaces sobrecarregados, apenas 5. Por sua vez o *Método de Descida* não removeu as 8 inviabilidades mas espalhou o tráfego extra. Verificando os enlaces 0-1 e 0-5 nota-se que o *Optimise* consegue eliminar a sobrecarga de 0-5 transferindo o tráfego extra para 0-1, aumentando ainda mais a inviabilidade. Esta estratégia não é adequada, pois inviabilidades pequenas podem ser tratadas com filas, mas sem garantir viabilidade para as soluções. Um outro ponto a considerar é que nenhuma dessas soluções representa a melhor conhecida. Aplicando o método de Grasp com filtro com 100 iterações e o fator alfa constante em 0.25 consegue-se obter soluções em torno de 108%.

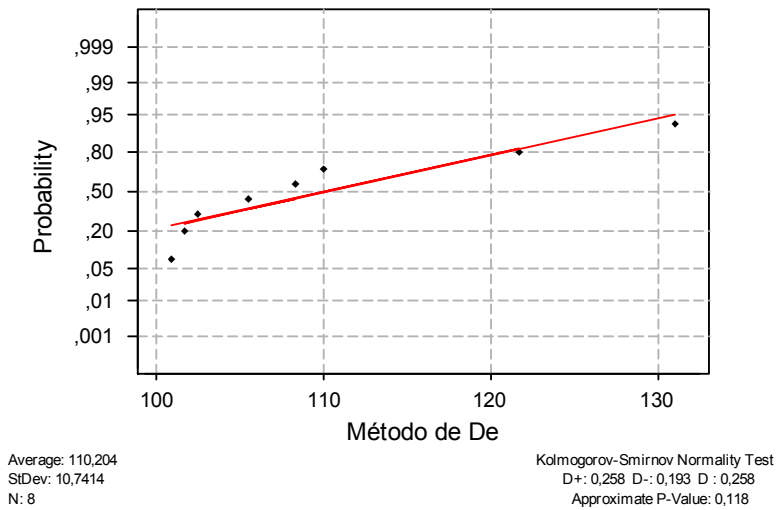
A seguir está descrita uma análise estatística realizada na tentativa de se comprovar que o método proposto é realmente melhor que o método atual. A metodologia adotada foi um Teste de Médias e o software estatístico o Minitab® versão 13.

As amostras consideradas foram em relação à taxa de utilização dos enlaces sobrecarregados. Devido ao reduzido tamanho das amostras foi realizado um teste de normalidade em ambas obtendo um razoável valor de prova máximo de 15%.

Normal Probability Plot



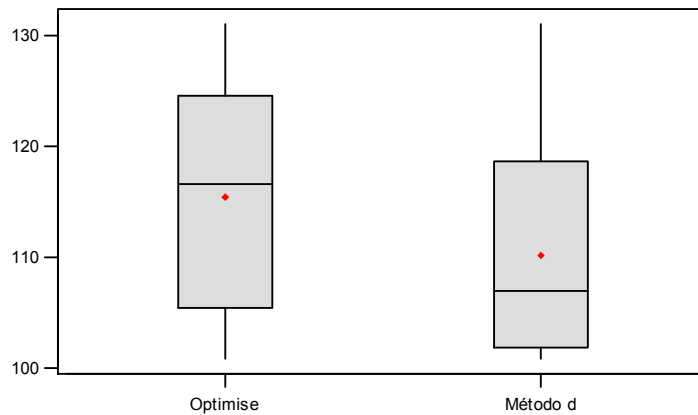
Normal Probability Plot



O Boxplot e o Dotplot nos permite uma melhor visualização das soluções geradas pelo método. Podemos observar pelo Dotplot que apesar do *Método de Descida* apresentar mais sobrecargas essas são pequenas e próximas ao limite do enlace.

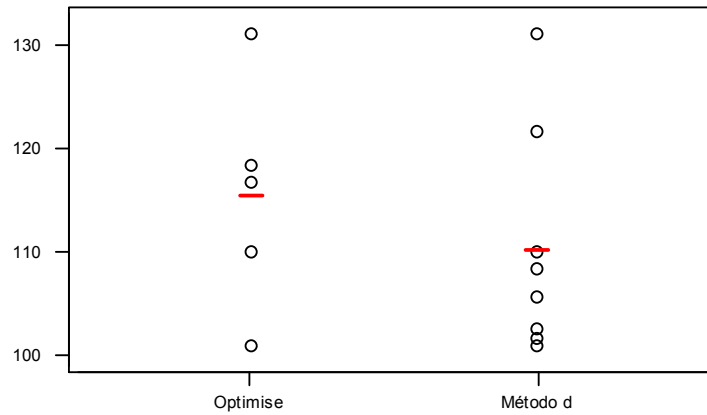
O resultado da análise de diferença entre médias está apresentado a seguir.

Boxplots of Optimise and Método d
(means are indicated by solid circles)



Dotplots of Optimise and Método d

(means are indicated by lines)



Two-sample T for Optimise vs Método de Descida

	N	Mean	StDev	SE Mean
Optimise	5	115,4	11,1	5,0
MétodoDes	8	110,2	10,7	3,8

Difference = mu Optimise - mu Método de Descida

Estimate for difference: 5,18

95% CI for difference: (-9,23; 19,59)

T-Test of difference = 0 (vs !=): T-Value = 0,83 P-Value = 0,431 DF = 8

A diferença estimada entre os métodos é de 5,15% e o teste de diferença entre as médias não é conclusivo, pois não foi possível rejeitar a H_0 . Esta análise fica descrita como apenas uma das atividades desenvolvidas neste trabalho.