Genetic Local Search for Multicast Routing

[Extended Abstract]

Mohammed S. Zahrani
University of Hertfordshire
School of Computer Science
Hatfield, AL10 9AB, UK
M.Zahrani@herts.ac.uk

Martin J. Loomes
Middlesex University
School of Computer Science
London NW4 4BT, UK
M.Loomes@mdx.ac.uk

James A. Malcolm
University of Hertfordshire
School of Computer Science
Hatfield, AL10 9AB, UK
J.A.Malcolm@herts.ac.uk

Andreas A. Albrecht
University of Hertfordshire
School of Computer Science
Hatfield, AL10 9AB, UK
A.Albrecht@herts.ac.uk

ABSTRACT

We describe a population-based search algorithm for cost minimization of multicast routing. The algorithm utilizes the partially mixed crossover operation (PMX) and a landscape analysis in a pre-processing step. The aim of the landscape analysis is to estimate the depth Γ of the deepest local minima in the landscape generated by the routing tasks and the objective function. The analysis employs simulated annealing with a logarithmic cooling schedule (LSA). The local search performs alternating sequences of descending and ascending steps for each individual of the population, where the length of a sequence with uniform direction is controlled by the estimated value of Γ. We present results from computational experiments on a synthetic routing tasks, and we provide experimental evidence that our genetic local search procedure, that combines LSA and PMX, performs better than algorithms using either LSA or PMX only.

Categories and Subject Descriptors

I.2 [**Artificial Intelligence**]: Problem Solving and Search

General Terms

Algorithms

Keywords

Multicast routing, genetic local search, simulated annealing.

1. INTRODUCTION

The problem of minimizing the tree costs of single requests under the constraint that all path capacities are within a user-specified capacity bound, i.e. the requests are executed simultaneously, is referred to as the capacity constrained multicast routing problem (CCMRP) [4].

The CCMRP can be formalised as a constrained Steiner tree problem, which is known to be NP-complete. We note that in applications like video conferencing, media broadcasting, and distance-learning the routing procedure is updated only from time to time, e.g. when new customers register to use one of the services. In such cases, off-line routing algorithms are an appropriate way to solve the routing problem. Since we are dealing with an NP-complete problem, local search methods are a natural choice to approach the problem.

We propose a genetic local search algorithm to solve multicast routing problems. The general structure of the algorithm follows the approach presented by Merz and Freisleben [7]: a classical local search is executed for each individual of a population and combined with a crossover operation, where a key element constitutes a thorough analysis of the underlying landscape.

GECCO'06, July 8–12, 2006, Seattle, Washington, USA.
ACM 1-59593-186-4/06/0007.

2. LSA PRE-PROCESSING

We employ the following formal definition of multicast routing: Given a graph $G = (V, E)$ that represents a communication network with node set V and edges E, we define two non-negative weight functions $Co : E \rightarrow \Re$ and $Ca : E \rightarrow \Re$, where Co is the cost function and Ca is the capacity function on E, respectively.

Each of the point-to-multipoint requests has a source node $s \in V$ and a set of destination nodes $D \subseteq V$. We define a multicast request R by setting
$R = [v_s \Rightarrow (v_1, v_2, ..., v_n); C]$, where
$v_s =$ the source node of R;
$D = \{v_1, ..., v_n\}$ = the destination nodes;
$C =$ the capacity required by each $v_s \Rightarrow v_i$.
The multicast problem P is then defined by $\mathsf{P} = [G; Co; Ca; R_1, ..., R_n]$.

In a pre-processing step, we utilize simulated annealing [1] for the analysis of multicast routing. The configuration space consists of all feasible solutions for a given multicast problem $\mathsf{P} = [G; Co; Ca; R_1, ..., R_n]$:

$$\mathsf{M} = \big\{S | S = [R_{i_1}, ..., R_{i_n}]; R_{i_1}, ..., R_{i_n} \text{ are KMB-routed}\big\}. \quad (1)$$

By N_S we denote the neighbourhood of S, and $Z(S)$ denotes the underlying objective function. Given $S \in \mathsf{M}$ by $S = [R_{i_1}, ..., R_{i_n}]$, the neighbourhood N_S includes S itself

and is defined by the following procedure:
(1) Two integers a and b, $1 \leq a < b \leq n$, are randomly chosen, and the order of all requests from number i_a to number i_b is reversed; a new potential configuration S' is generated.
(2) The potential configuration S' is validated for feasibility, i.e. we try to simultaneously schedule all the requests from R_{i_b} upwards by using an implementation of the KMB algorithm [6]. If a conflict occurs, a new pair (a, b) is generated.
(3) If S' indeed belongs to M, the objective function $Z(S)$ is calculated.

The objective function represents a combined measure of transmission costs and capacity constraints: let $T(R)$ denote the set of edges of the tree associated with the request R from configuration $S \in \mathsf{M}$. We first define $W(R) := C \cdot \sum_{e \in T(R)} Co(e)$. The value of the objective function $Z(S)$, $S \in \mathsf{M}$, is then simply given by $Z(S) := \sum_{R \text{ from } S} W(R)$.

We consider a special type of simulated annealing that is based on Hajek's theorem [5] for the cooling schedule $c(k) = \Gamma / \ln(k+2)$, $k = 0, 1, ...$: if $F_{\min}$ denotes the set of optimum solutions and $\mathbf{a}_f(k)$ is the probability to be in f after k steps, the asymptotic convergence $\sum_{f \in \mathsf{F}_{min}} \mathbf{a}_f(k) \underset{k \to \infty}{\longrightarrow} 1$ of logarithmic simulated annealing (LSA) is guaranteed if and only if Γ is lower bounded by the maximum value of the minimum escape height from local minima.

In our approach, the landscape analysis basically estimates Γ. For a pre-defined number of transitions T, we proceed as follows in order to find an estimation of Γ: first, the procedure tries to estimate the intermediate increase G_{est} of the objective function between two successive improvements of the best value $Z(S)$ found so far. Then, we establish a conjecture about Γ_{est} that is based on G_{est}. The initial estimate is from two randomly generated initial solutions: $G^0_{est} :=| Z(S^1_0) - Z(S^2_0) |$.

The LSA algorithm has been implemented in Java. The underlying graphs are the instances steinb10, steinb11, and steinb18 from the OR library [3] (here, we report results for steinb18 only). The graphs have $75 - 100$ nodes and $150 - 200$ edges. Each edge was randomly assigned a cost value $Co(e) \in \{1, 2, ..., 10\}$; the capacity of edges was set by $Ca(e) = 12$. For the steinb-instances, 20 requests were generated randomly. From the 20 requests, we derived 12 multicast routing problems P_i, with P_i defined by $\{R_1, R_2, ..., R_i\}$, $i = 9, ..., 20$. For each P_i, we performed 12 computational experiments: for each of the two values of $T = 10^4, 2 \cdot 10^4$ the experiments were executed for 6 different values $\Gamma := G_{\text{est}}/c$ for $c = 1, 2, 4, 8, 16, 20, 32$. Based on our computational experiments, we established the conjecture:

$$\Gamma_{est} \approx G_{est}/10. \tag{2}$$

3. GENETIC LOCAL SEARCH FOR MULTICAST ROUTING

Over the past few years, genetic local search has been investigated in the context of a variety of combinatorial optimisation problems; cf. [7] and the literature therein. The basic idea is relatively simple: a (quasi-)deterministic local search with continuous improvements of the objective function is executed for all individuals of a population; if the individual runs are stuck in local minima, a crossover operation is applied in order to leave local minima.

In our heuristic, we employ such a "modest random" procedure: if after $L = 50$ unsuccessful trials no neighbour with a better value of the objective function has been found, the current solution S is declared to be a potential local minimum, and the procedure switches from downward steps to a sequence of strictly upward steps with random selection in neighbourhoods. The upward steps are executed until either an S' with $Z(S') \geq Z(S) + \Gamma_{\text{est}}$ has been reached, or after L unsuccessful trials no neighbour with a larger value of the objective function has been found. In either of the two cases, the procedure switches back to strictly downward steps with random selection in neighbourhoods. Thus, for each individual, a random walk through the landscape is executed, and after K steps, the walk is interrupted by the partially mixed crossover (PMX) operation [2, 8] and "roulette wheel" selection under the elitist model is applied in order to generate a new population of the same size.

Apart from Γ_{est}, L and K, the parameters are M = *population size* and N = *total number of PMX applications to a single element*. The parameters were chosen in such a way that $M \cdot K \cdot N$ is in the region of T from our experiments with LSA (Section 2): $M = 7, 10$, $K = 70, 80$, and $N = 20, 25$.

In Table 1, the numbers in bold face indicate improvements in comparison to the results obtained by LSA; cf. Section 2. The genetic local search with LSA pre-processing performs better on larger multicast routing instances compared to applications of either LSA or PMX only, especially for a total number of operations $M \cdot K \cdot N$ that is equivalent to $T = 2 \cdot 10^4$. For most of the instances, LSA produces better results than the use of PMX crossover only.

Table 1: Z_{best} (GLS: elitist PMX, steinb18).

Size of P_i	$N=20$, $K=70$, $L=50$, $M=7$	$N=25$, $K=80$, $L=50$, $M=10$	
	GLS	GLS	PMX only
15	2216	2204	2218
16	2379	2375	2385
17	2523	**2500**	2539
18	2646	**2634**	2653
19	2736	**2723**	2749
20	2967	**2947**	2993

4. REFERENCES

[1] E. Aarts. *Local Search in Combinatorial Optimization*. Wiley & Sons, New York, 1998.

[2] P. Baudet, C. Azzaro, L. Pibouleau, S. Domenech. A genetic algorithm for batch chemical plant scheduling. In *Proc. Int. Congr. Chemical & Process Engineering*, pp. 25–30, 1996.

[3] J. Beasley. OR library: http://people.brunel.ac.uk/~mastjjb/jeb/info.html.

[4] C. Diot, W. Dabbous, J. Crowcroft. Multipoint communication: A survey of protocols, functions, and mechanisms. *IEEE Journal on Selected Areas in Communication*, 15:277–290, 1997.

[5] B. Hajek. Cooling schedules for optimal annealing. *Mathem. of Operations Research*, 13:311–329, 1988.

[6] L. Kou, G. Markowsky, L. Berman. A fast algorithm for Steiner trees. *Acta Informatica*, 15:141–145, 1981.

[7] P. Merz, B. Freisleben. Fitness landscapes, memetic algorithms, and greedy operators for graph bipartitioning. *Evolutionary Computation*, 8(1):61–81, 2000.

[8] L. Zhu, R. Wainwright, D. Schoenefeld. A genetic algorithm for the point to multipoint routing problem with varying number of requests. In *Proc. IEEE Int. Conf. on Evolutionary Computation (1998)*, pp. 171-176, 1998.