

无级电话网动态路由策略的研究

万海¹ 郝小卫² 李磊³

(中山大学信息科学与技术学院 广州510275)¹ (广东省电信公司 广州510275)²

(中山大学软件研究所 广州510275)³

摘要 无级电话网的动态路由技术能有效降低呼损、提高接通率。本文针对所开发的一个集中式、全局优化无级电话网动态路由策略选择系统进行综述;详细分析了求解路由方案的四种算法;介绍了该系统的实现、运行情况和话务模拟结果;该系统在广东省长话网得到了良好的应用。

关键词 无级电话网,动态路由,路由求解算法

Research on Dynamic Routing Strategy of Nonhierarchical Telecommunication Network

WAN Hai¹ HAO Xiao-Wei² LI Lei³

(Department of Computer Science and Technology, SUN YAT-SEN University, Guangzhou 510275)¹

(Guangdong Communication Corporation, Guangzhou 510275)²

(Software Research Institute of SUN YAT-SEN University, Guangzhou 510275)³

Abstract The Dynamic Routing of nonhierarchical telecommunication network is an efficient solution to high traffic loss rate. This paper presents and introduces a concentrated, global optimization dynamic routing strategy selection system used in nonhierarchical telecommunication network, analyzes the four algorithms of computation routing scheme, introduces the implementation of the algorithms and states the result of simulations. This system is used well in Guangdong long-distance telecommunication network.

Keywords Nonhierarchical telecommunication network, Dynamic routing

1 引言

1.1 背景与相关工作

随着现代通信的迅速发展,电话网的选路技术从传统的分级固定路由发展到无级动态路由。无级电话网动态路由技术是指在电话无级网的主叫和被叫交换局之间,当直达路由忙或无直达路由时,动态选择一个迂回节点将话务接通^[9]。动态路由是组合优化问题,具有很高的复杂度^[7];其问题本质是如何充分利用各交换局话务忙时的不一致性;并进行多次选择路由,以选择空闲或费用最小的路由,增加网络的灵活性^[1]。

国外采用的动态路由策略主要是依靠特定的交换机,针对本国电话网特点应用局部或半全局优化策略进行集中控制(如 DNHR^[1,2]、DCR^[3])或分散控制(RTNR^[7]、DAR^[4])。由于网络设备和营运与国外都有很大差异,这些动态路由策略并不适用于广东省电话网^[10]。根据广东省电话网的具体情况,决定在“广东省电话网智能管理系统”项目开发中采用集中式、基于状态的全局优化动态路由策略。这就要求在算法上解决两个问题:(1)如何得到全局优化的路由选择方案;(2)如何在尽量短的时间内计算出路由选择结果^[6]。

1.2 本文的工作

为实现无级电话网动态路由策略,我们所研究实现的:非线性单纯形算法^[10]、遗传算法^[7,8]、Tabu 算法^[6]、IDCR 算

法^[10]四个路由选择算法均能有效地在较短的时间内求解动态路由方案。本文将分析无级电话网的路由问题,路由选择算法的原理和实现方案;并通过模拟仿真程序评估各算法的具体应用效果,评估算法运行效率与性能。本文的分析结果对于“广东省电话网智能管理系统”的实际运行和进一步研究动态路由策略有极大的指导意义。

2 动态路由问题描述

电话网由交换局和局间连通链路组成;在全连通的无级电话网中,任意交换局之间都有双向连通链路。整个网络可以看成是一个有向全连通图 G :

$$G = (N, E) \quad (1)$$

其中: N 是图中的节点(交换局)集合, E 是边(链路)的集合。令 $e = |E|$, $n = |N|$, 则在 G 中, 得到:

$$e = n \times (n - 1) \quad (2)$$

用 c_{ij} ($i \neq j, i, j \in N$) 来表示从节点 i 到 j 的直达中继数; a_{ij} 表示节点 i 到 j 的话务流量, 单位为爱尔兰。

设 $i, j, k \in N$, 则对于从交换点 i 到 j 的话务, 如果超过 i 到 j 间链路能承受的话务量, 就会产生话务溢出, 此时需将溢出话务通过一迂回节点 k 迂回, 即溢出话务通过 e_{ik} 和 e_{kj} 的链路来传送。

由式(2)知:任意的链路 e 都可作为迂回路由, 所以共有 $n \times (n - 1)$ 条链路作为备选路由。用 r_{ij} 表示节点 i 到节点 j 的迂回节点 ($i \neq j$), r_{ij} 是除 i, j 以外的任意节点或者为空(即不

选迂回节点)。

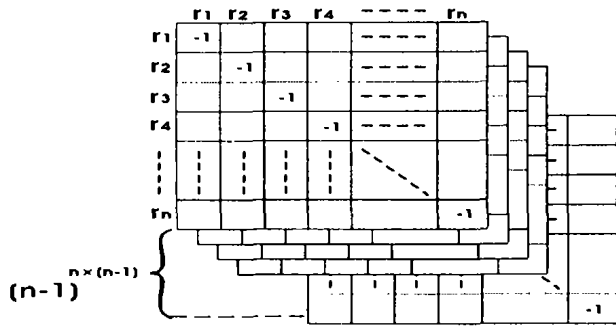


图1 动态路由问题空间

为了将动态路由问题转化为确定路由表的组合优化问题,用矩阵表示路由表(如图2所示):

$$Rt = (r_{ij})_{n \times n} \quad (3)$$

从式(3)可知,动态路由问题:

- (1) 共 $n \times (n-1)$ 个迂回节点需确定;
- (2) 每个迂回节点的备选节点有 $n-1$ 个;
- (3) 一个迂回节点的确定会影响其他迂回节点的确定。

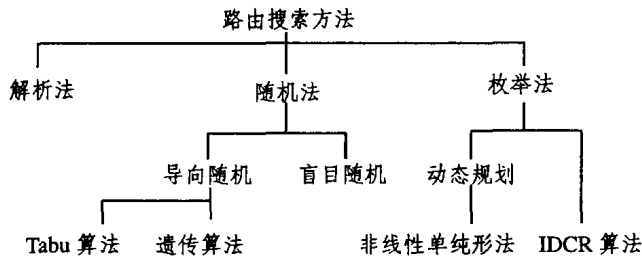


图2 路由搜索算法分类

因此,动态路由问题的状态空间大小达到 $(n-1)^{n \times (n-1)}$,如果用穷举的方法求解这个问题,其空间复杂度将达到 n^n ,且状态空间是非线性、离散的;不可能用全局搜索的方法求得最优解,因此一般只能找到近似的最优解。

3 动态路由算法

无级电话网动态路由问题是组合优化问题,通常有大量的局部极值点,往往是不可微的、不连续的、多维的、有约束条件的、高度非线性的问题,精确求得全局最优解一般是不可能的。现有算法的缺陷主要是为了简化计算而引入的。为了避免已有算法的缺陷,我们需要研究有效的全局搜索算法。

结合广东省电话网的实际情况,我们发现不可能照搬国外的技术。广东省电话网已经完成无级电话网改造,关键问题是如何选择一种高效、可靠的动态选路策略。为此我们提出了以下的实施方案:

(1) 集中式控制 分散式动态选路需要将选路算法集成入交换机。但由于交换机型号众多,这种方法几乎不可能。而采用集中式控制,能够进行全局优化,且成本较低,可以最大限度地减少对交换网、交换机和已有网管系统的改动;还可以充分利用计算机系统的计算能力,并利于系统的扩展。

(2) 基于状态选路 需要及时地从电话网中获取数据,必要时可以对话务数据作修补处理;系统有较强的适应性,能及时对电话网的异常情况作反应;疏通话务流量,尽可能利用网络设备资源。系统能够运行在复杂多变的网络环境中。

这就要求所研究的新算法能够用较短时间,求解一个全局优化的路由方案,满足集中式动态选路对时间和质量的要

求。本文分别对非线性单纯形算法、遗传算法、TABU 算法、IDCR 算法(图2所示:路由搜索算法分类)应用于电话网的原理进行分析,实现全局优化的动态路由目标,即:

确定式(3)中所有的 r_{ij} ($1 \leq i \neq j \leq n$),求解路由表并使全网阻塞率最小。

考虑到话务阻塞的原因是话务整体分布不均衡,我们确定用各链路的负载均衡程度来评价路由表的优劣,即:负载均衡,适应度越高。

3.1 非线性单纯形算法

3.1.1 算法基本原理 非线性单纯形法^[12]的思想是基于单纯形算法^[11]在推广到的 n 维欧氏空间构造一个 n 维单纯形(即初始解集),计算此单纯形上每个顶点的目标函数值,然后反复替换单纯形的一个顶点;每次替换目标函数值最大的顶点,新顶点在被替换顶点向其余顶点的中心所引的射线上取得,并要求每次替换顶点后新顶点与其他的顶点(共 $n+1$ 个顶点)仍构成一个 n 维单纯形。直到找到最优解(或满足某些中止条件时)停止替换。

以三个点在三维空间为例(如图3(a)):假定 $f(x1) > f(x2) \wedge f(x1) > f(x3)$,构造平行四边形(如图3(b)),在对角线上选取点做为 $x1$ 的替换节点,利用 $x2, x3, x4$ 重复上面的计算,直到找到最优解。

为了应用非线性单纯形法,我们首先对状态空间进行重构,并使目标函数曲面尽量平滑;非线性单纯型算法不是随机算法,方向性更强,具有计算量小,速度快的优点。在状态空间的规模不太大的条件下,非线性单纯型算法能找到比较理想的解。在实际应用中,一般来说需要求解路由的链路数(即溢出量不为0的链路)不多,因此非线性单纯形算法能较好地应用于选择动态路由策略。

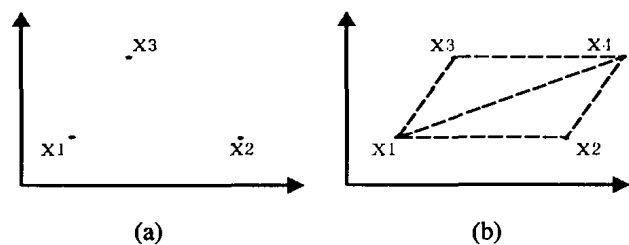


图3 非线性单纯形法

3.1.2 算法具体实现

(1) 把动态路由问题归结为最优化问题。我们采用通过评估函数来评价电话网的溢出率,根据评估结果来确定该路由表是否适用。我们采用了所有链路的各线平均负载的方差作为路由方案的评估函数 $\sigma^2(Rt)$;即对有 n 个节点的网络,求最优化问题:

$$\min_{Rt \in D} \sigma^2(Rt) \\ D = \{Rt | Rt = (r_{ij})_{n \times n}, r_{ij} = 0, r_{ij} \in \{1, n\}, r_{ij} \neq i, j, 1 \leq i \neq j \leq n\}$$

$$\text{其中 } \sigma^2(Rt) = \frac{1}{N} \sum_{i=1}^n \sum_{j=1, j \neq i}^n C_{ij} \cdot \left[\frac{ld_{ij}(Rt)}{C_{ij}} \right]^2 - \left[\sum_{i=1}^n \sum_{j=1, j \neq i}^n ld_{ij}(Rt) / N \right]^2$$

$$N^2 = \frac{1}{N} \sum_{i=1}^n \sum_{j=1, j \neq i}^n \frac{ld_{ij}^2(Rt)}{C_{ij}} - \left[\sum_{i=1}^n \sum_{j=1, j \neq i}^n ld_{ij}(Rt) / N \right]^2 \quad (4)$$

C_{ij} 表示链路 (i, j) 的中继数; N 表示总中继数; $ld_{ij}(Rt)$ 表示链路 (i, j) 话务负载;

当 $\sigma^2(Rt)$ 越接近0,说明此路由表的全网溢出率越小,其

接通过率就越高,话务流量就越均衡。

(2) 构造状态空间。用式(3)来表示一个路由表的方法很直观,但在编码实现上却很不方便,因此要把矩阵转换为一个 m 维向量。一个路由表 Rt 的矩阵表示为:

$$Rt = \begin{pmatrix} 0 & r_{12} & r_{13} & r_{14} & \dots & r_{1n} \\ r_{21} & 0 & r_{23} & r_{24} & \dots & r_{2n} \\ r_{31} & r_{32} & 0 & r_{34} & \dots & r_{3n} \\ r_{41} & r_{42} & r_{43} & 0 & \dots & r_{4n} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & r_{n3} & r_{n4} & \dots & 0 \end{pmatrix}$$

可把 Rt 的各行元素(对角线上为0的元素除外)按从左到右的顺序连成一个 $n \times (n-1)$ 维向量。

定义 映射 $H(P): P^{n \times (n-1)} \rightarrow \{(r_{ij})_{n \times n} | r_{ij} \in R\}$, 该映射把 $n \times (n-1)$ 维向量 P 还原为对角线上元素为0的 $n \times n$ 矩阵 R 。容易证明该映射是一一对应的,可以利用 $H(P)$ 把原来问题的矩阵空间转换到 $\langle 0, 1, \dots, n-2 \rangle^{n \times (n-1)}$ 的空间上。

很多情况下,并非向量 P 的每个分量都对目标函数 $\sigma^2(H(P))$ 起作用。事实上,若链路 (k, s) 上的溢出话务量 $o_k = 0$, 则目标函数 $\sigma^2(Rt)$ 与 Rt 的元素 r_{ks} 的取值无关。结合映射 H 所作的转换,我们可以对路由表作进一步的转换:把路由表 Rt 按从左到右、从上到下的顺序依次连成一个 t 维变量(t 等于链路总数 $(n \times n)$ 减去对角线元素数 (n) 再减去所有溢出量为0的链路后的链路数)。

定义 映射 $G(P): P^t \rightarrow \{(r_{ij})_{n \times n} | r_{ij} \in R\}$, 该映射把 t 维向量还原为对角线上元素为0的 $n \times n$ 矩阵,其中溢出量为0的链路对应的元素统一取-1。在忽略溢出量为0的链路的前提下,映射 G 是一一对应的。利用映射 G ,我们可以把 $n \times n$ 的矩阵空间转换到 $\langle 0, 1, \dots, n-2 \rangle^n$ 空间上,使计算量达到最小。

(3) 初始解空间的优化。上述的 t 维向量 P 的各分量的取值是0到 n 之间的某一个值,没有大小意义,缺乏方向性。可以通过对向量 P 的各分量的候选值以某指标重新排序编号,以新编号作候选值使之具有某种方向性,使上述算法所述的射线方向尽量指向最优点,减少计算量。实践证明用下面的函数作向量 P 的第 i 个分量的排序指标效果良好:

$$f_i(x) = \sigma^2(G(\underbrace{0, 0, 0, \dots, 0}_{i-1 \text{ 个}}, x_i, \underbrace{0, 0, 0, \dots, 0}_{i-1 \text{ 个}}))$$

3.2 遗传算法

3.2.1 算法基本原理 遗传算法是借鉴生物界自然选择和自然遗传机制的随机化搜索算法,其主要特点是群体搜索策略和群体中个体之间的信息交换,具有简单、通用、鲁棒性强的特点,有很广的应用范围,适合于在复杂而庞大的搜索空间中寻找最优解或准最优解。由于动态路由问题的搜索空间庞大,应用遗传算法于动态路由能发挥其优势。同时能够实现对路由求解空间的全局搜索。

3.2.2 算法具体实现

(1) 编码问题。通常遗传算法使用二进制编码。但在路由求解算法中,使用二进制编码不便于操作和分析,也不能反映问题本身的结构特征,所以我们直接把求解的 $n \times n$ 的路由矩阵作为编码,每一位上可以取的值得从0到 $n-1$ 。

(2) 适应度函数。计算适应度函数的方法是计算电话网全局的平均每线话务量和每一条链路的每线话务量,然后求每线话务量的均方差,再适当定标作为个体的适应度。

采用记号 $LINE, TRAFFIC, ROUTE$, 并引入记号 $R, O-TRAFFIC$, 定义如下:

$$R = (r_{jk})_{n \times n \times n}, \text{ 其中: } r_{jk} = \begin{cases} 1, & \text{若 } Route_{ij} = k; \\ 0, & \text{否则。} \end{cases}$$

$O-TRAFFIC = (OTraffic_{ij})_{n \times n}$ 表示链路上的溢出话务量。溢出话务量等于送入话务量乘以链路上的溢出率,链路的溢出率用 Erlang-B 公式计算。其中: $O-Traffic_{ij} = Erlang(Traffic_{ij} + Traffic_{jn}, Line_{ij}) * Traffic_{ij}$ 。

由于链路是双向的,所以把两个方向的送入话务量相加作为整条链路的送入话务量,而链路中继数满足

$$Line_{ij} = Line_{ji} (0 \leq i, j, k < n)。$$

链路 i, j 负载率:

$$Load_{ij} = (Traffic_{ij} - O-Traffic_{ij} + \sum_{0 \leq i, j, k < n} r_{ik} * O-Traffic_{ik} + \sum_{0 \leq i, j, k < n} r_{kj} * O-Traffic_{kj}) / Line_{ij} \quad (5)$$

式中,第一项为链路 (i, j) 的直达送入话务量,第二项为链路 (i, j) 溢出的话务量,第三和第四项是其它链路溢出到链路 (i, j) 上的话务量。其中第三项是 i 通过 j 迂回到其它节点的溢出话务,第四项是其它节点通过 i 迂回到 j 的溢出话务。

全网平均负载率:

$$avgLoad = (\sum_{0 \leq i, j < n, i \neq j} (Traffic_{ij} - O-Traffic_{ij} + \sum_{0 \leq i, j, k < n} R_{ik} * O-Traffic_{ik} + \sum_{0 \leq i, j, k < n} R_{kj} * O-Traffic_{kj})) / \sum_{0 \leq i, j < n, i \neq j} Line_{ij} \quad (6)$$

全网的平均负载率均方差为:

$$S = \frac{\sum_{0 \leq i, j < n, i \neq j} Line_{ij} * (Load_{ij} - avgLoad)^2}{\sum_{0 \leq i, j < n, i \neq j} Line_{ij}} \quad (7)$$

我们将 S 采用指数定标后作为对个体的评价。最终的适应度函数如下:

$$F = e^{-S} \quad (8)$$

(3) 遗传算子的选择

• 选择算子:采用适应度比例法和最佳个体保存方法。即个体的选择概率与相对适应度成正比。群体大小设为 n , 个体 i 的适应度函数值为 $f(i)$, 则 i 被选择的概率为:

$$f(i) / \sum_{i=1}^n f(i) \quad (9)$$

同时每一代的最优个体被保留下来复制到下代。

• 交叉算子:采用单点交叉,随机生成一个 $[0, n \times n)$ 之间的整数 $cpos$, 将两个进行交叉操作的个体从第 $cpos$ 位到最后一位(第 $n \times (n-1)$ 位)的一段染色体互相交换。交叉概率取为0.1。

• 变异算子:采用单点变异,随机生成一个 $[0, n \times n)$ 区间内的随机整数 $mpos$, 然后把进行变异操作的个体 $mpos$ 基因座上的内容用随机产生的另一合法的内容代替。变异概率取为0.05。

3.3 TABU 算法

3.3.1 算法的原理 TABU 是一种启发式的优化搜索算法,其核心在于对搜索过程使用短期记忆和中长期记忆,以令搜索具有广泛性和集中性。TABU 的基本思想是在可行的解空间中由一个解找到另外一个更好的解。但为了能够逃出局部极值和避免循环,算法中设置了禁止表,当搜索的解在禁止表中,则放弃该解。

3.3.2 算法具体实现

(1) 状态空间重构。TABU 算法直接使用在路由方案问题上还是有一定的困难:状态空间过大,邻域难以定义。

为了克服以上困难,我们令 TABU 表中的元素是状态空间中一个范围,该范围内的所有点都是禁止的。这样 TABU 表能起的作用就更大了。为此,我们首先将问题的状态空间通

过一个映射作了重构,在重构的状态空间里,范围和 x 的邻域都可以得到适当的定义。

定义序数函数

$$\text{Order}(a, A) = \{k | a_k = a, \text{ 且 } k \leq i \text{ 或 } a_k < a, \text{ 对所有 } 1 \leq k \leq n\} \quad (10)$$

其中 $A = (a_1, a_2, \dots, a_n)$, 是包含 n 个实数的有序序列。序数函数实际上是表达了 a 在序列 A 中的大小顺序。

定义空闲指标如下:

$$f(i, j, k) = (l_k + l_i) (1 - \text{Erlang}(t_k, l_k)) (1 - \text{Erlang}(t_i, l_i)) \quad (11)$$

空闲指标是一个经验公式,表达了对中转节点和中转链路的转接能力作的一个评估,

定义有序序列 A_i 为:

$$A_i = (f(i, j, 1), f(i, j, 2), \dots, f(i, j, i-1), f(i, j, i+1), \dots, f(i, j, j-1), f(i, j, j+1), \dots, f(i, j, n)) \quad (12)$$

实际上 A_i 就是除去 $f(i, j, i)$ 和 $f(i, j, j)$ 后所有 $f(i, j, k)$, $1 \leq k \leq n$ 按 k 的顺序排列。

定义转换函数:

$$H(i, j, k) = \begin{cases} \text{Order}(f(i, j, k), A_i), & \text{当 } 0 < k \leq n, k \neq i, k \neq j \\ 0 & \text{当 } k = 0 \end{cases} \quad (13)$$

其中 i 是发端交换点, j 是到端交换点。

我们将路由方案通过一个映射转换为一个空间上的一点:去除式(3)矩阵对角线上的元素,从上到下地将每行的元素从左到右按顺序排列,并对每个元素使用转换函数式,得到一个新的向量。

定义路由方案的转换函数 $H(R)$ 为:

$$H(R) = (H(1, 2, r_{12}), H(1, 3, r_{13}), \dots, H(1, n, r_{1n}), H(2, 1, r_{21}), H(2, 3, r_{23}), \dots, H(2, n, r_{2n}), \dots, H(n, 1, r_{n1}), H(n, 2, r_{n2}), \dots, H(n, n-1, r_{n, n-1}))$$

在新的状态空间上,每个点都代表不同的路由方案,并且可以用一个 $n(n-1)$ 维的坐标表示。

状态空间重构并没有降低问题的复杂度,但通过映射,中转能力强的节点的坐标值就会较大。通过映射,将原来无序和不规则的状态空间转变得相对有序和规则,使得目标函数更加平滑,也使得搜索方向性更强。TABU 搜索算法具有很强的局部搜索功能。

(2) TABU 算法改进。用多维欧氏空间的两点直线距离来定义新的状态空间中两点距离:

设点 p_1 和 p_2 , p_1 的坐标是 $(a_1, a_2, \dots, a_{n(n-1)})$, p_2 的坐标是 $(b_1, b_2, \dots, b_{n(n-1)})$, 则两点的距离是:

$$d(p_1, p_2) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \dots + (a_{n(n-1)} - b_{n(n-1)})^2} \quad (14)$$

设路由方案 R_1 和 R_2 , 定义两个路由方案之间的距离 $d(R_1, R_2)$ 是:

$$d(R_1, R_2) = d(H(R_1), H(R_2))$$

令 TABU 表中的一个元素由两个部分组成: (R, dr) , 其中路由方案 R 和距离 dr 。规定满足以下条件的路由方案 R' 为属于该元素的范围:

$$d(R, R') \leq dr \quad (15)$$

对于基本 TABU 算法步骤(1)中生成候选解集中 x 的邻域,我们为了提高程序的效率,作了一个简单的定义。已知 H

$(x) = (a_1, a_2, \dots, a_i, \dots, a_{n(n-1)})$, 定义 $N(x, i)$ 如下:

$$N(x, i) = \{(a_1, a_2, \dots, k, \dots, a_{n(n-1)}) | 0 \leq k \leq n-2 \text{ 且 } k \neq a_i\} \quad (16)$$

其中 $H(x)$ 中的第 i 个分量是 a_i , 式子的意义实际上是把 $H(x)$ 的第 i 个分量替换成其他所有的可能值,这些新的向量组成 x 的邻域。

使用评估函数式(4)作为目标函数 $z(x)$ 。

3.4 IDCR 算法

3.4.1 算法的原理 IDCR 算法是改进动态控制选路 DCR^[3] 得到的算法,DCR 的选路策略是一种近似最优策略,其思想是尽量把呼叫分配到接受一个追加呼叫所费总代价最小的路由上;其选路原则是:在没有可用直达链路时,呼叫总是选取具有相对空闲电路数最多的链路进行迂回。

在 IDCR 算法设计中,我们采用了这种思想,但采用以下公式所计算的相对空闲指标来代替原 DCR 指标:

相对空闲指标 = $(1 - \text{占用话务量} \div \text{可用电路数}) \times \text{空闲电路数} \times \text{经验控制权值}$

$$\text{flag}_{ir} = (1 - O_{ij} / V_{ir}) \times I_{ir} \times \text{Exp}_{ir} \quad (17)$$

任意节点对 (i, j) 的转接节点 r , 的选择应为使下面的公式达到最大值的 r :

$$\text{Max}(\min[\text{flag}_{ir}, \text{flag}_{jr}]) \quad (r \neq i, j) \quad (18)$$

其中, O_{ij} : 链路 (i, r) 和链路 (r, j) 的占用话务量之和; V_{ir} : 链路 (i, r) 的可用中继数; I_{ir} : 链路 (i, r) 的空闲中继数; Exp_{ir} : 链路 (i, r) 的经验控制权值。

IDCR 算法实际上是一种贪婪法(greedy method),它对每个待选的迂回路由的选择是根据空闲指标进行的,在选择后不再回退,因此不能保证选到最优解。

IDCR 算法的解是路由表,最多要做 $O(n^2)$ 次路由选择,每次路由选择有 $O(n)$ 个候选节点,对每个候选节点计算其空闲性指标只需常数时间,因此, IDCR 算法的复杂性是 $O(n^3)$ 。

3.4.2 算法具体实现 IDCR 算法具体实现流程:

- 1) 从数据库读入话务数据。
- 2) 计算各链路的空闲电路数。
- 3) 如果 $i \neq j$ 且链路 (i, j) 的占用话务 > 0 , 则跳到5); 否则,跳到6)。
- 4) 把所有不等于 i 和 j 的节点当作链路 (i, j) 的迂回节点,计算其相对空闲指标,从中找到相对空闲指标最大的节点作为链路 (i, j) 的迂回节点 r 。
- 5) 把链路 (i, j) 的可能溢出话务量加到迂回节点 r 的占用话务量上。
- 6) 如果还没有把所有的链路的迂回节点算出,则跳到4)把计算结果写入数据库。

4 系统实现与算法对比分析

4.1 GTNIMS 的研制与开发

本文所讨论的四种路由搜索算法都在“广东省长话网无级动态路由智能管理系统(GTNIMS)”得到具体应用。GTNIMS 由广东省电信公司(原广东省邮电管理局)和中山大学软件研究所合作研究开发,实现了从广东省长途电话网实时数据采集,分析修补数据,实时计算路由选择方案取代原来网络管理需要人工干预的工作,实现了动态路由策略;该系统正在广东省长途电话网络管理中心使用;系统主要由:系统管理

子系统、路由求解子系统、数据处理子系统、话务仿真/评估子系统、后台监控子系统、数据接口子系统组成,且系统可以分布运算;选用的操作系统:Sun Solaris,数据库系统:Oracle9i(如图4)。

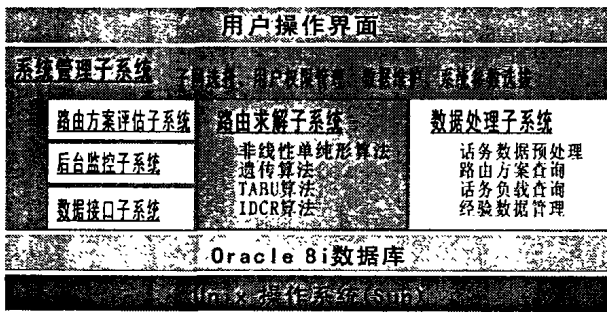


图4 GTNIMS体系结构

GTNIMS系统的话务仿真/评估子系统用于评估各算法的性能。该子系统模拟一段时间内交换局之间的话务发生、到达、接通和阻塞的情况,统计全网发生的话务总量和话务阻塞的总次数,由此求出话务的阻塞率。假设电话系统中每个呼叫的到达呈泊松分布,任意两个节点对*i-j*之间平均呼叫到达率 $\lambda_{ij} = \text{traf}_{ij}$ 为已知常数,这里 traf_{ij} 为局*i*与局*j*间的平均话务量;呼叫占用时长呈指数分布,均值 $1/\mu$ 为每次呼叫平均占用时长,取为3分钟^[10]。模拟过程先经过一个瞬态模拟阶段,使网络从占用状态为0进入稳定的占用状态。再开始稳态模拟阶段,两个阶段处理基本一样,只是仅在稳态模拟阶段进行数据统计,以减少统计上的误差。模拟算法中,瞬态模拟400秒,稳态模拟1小时,模拟时间共4000秒,普通时间约发生100万次呼叫,忙时发生约130万次呼叫。统计稳态模拟中的全网总阻塞率。

4.2 广东省长话话务分析

目前广东省内共有23个长话汇接局,大部分都配备了两台或两台以上的自动程控交换机,共有43个交换机,相应的状态空间约为 $41^{1806} \approx 10^{2912}$ 。

4.2.1 负载分析 选取17个节点进行链路负载分析(图5)。可知:电话网总体资源配备较充足,平均负载率总体还比较低,网络效能还有潜力可挖;各个局负载情况有较大差别,表明了话务分布的不均衡或资源配备的不合理,这为实施动态路由调控提供了条件。

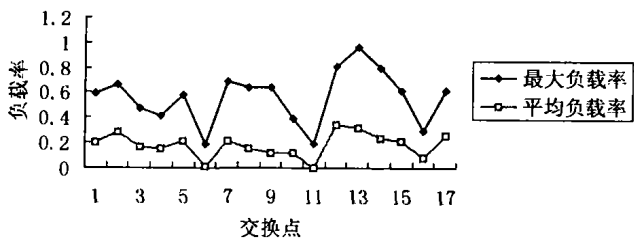


图5 链路负载图

4.2.2 呼损分析 呼损主要由几部分组成:用户使用不当,被叫用户忙或久叫不应,线路资源不足等。其中线路资源不足造成的阻塞呼损是我们所要考虑的部分,本文所提呼损是指这一部分。一般来说,话务忙时集中在上午10~12点、下午3~5点、晚上7~9点几个时段,这几个时段是话务高呼损时段,是实施动态路由的关键时段。由于各地地理位置及社会经济发展水平的不同,各地话务的高峰忙时会略有差别。

为了公平地进行算法比较分析,分别选定广东省长途电

话网的普通时间(2002年1月22日)和忙时(2002年五一节期间、春节)数据分别进行了路由求解和模拟。以下是这几个时段的话务量统计图。

(1) 2002年2月21日11:00—28日 23:00

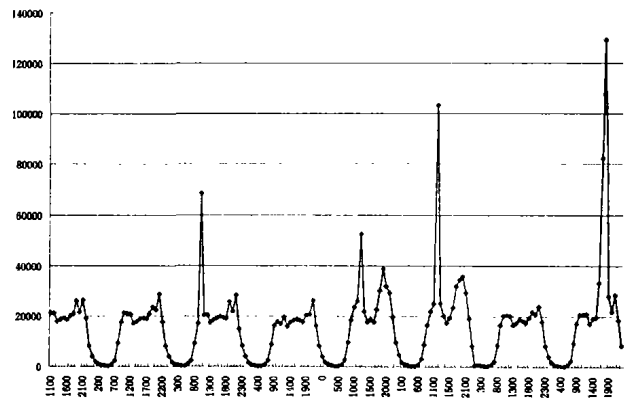


图6 各时段的话务量统计图

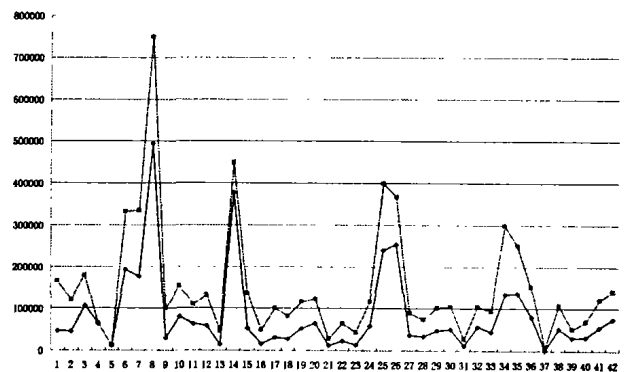


图7 各交换局发出/接收话务量统计图

(2) 2002年5月1日0—5月6日23:00

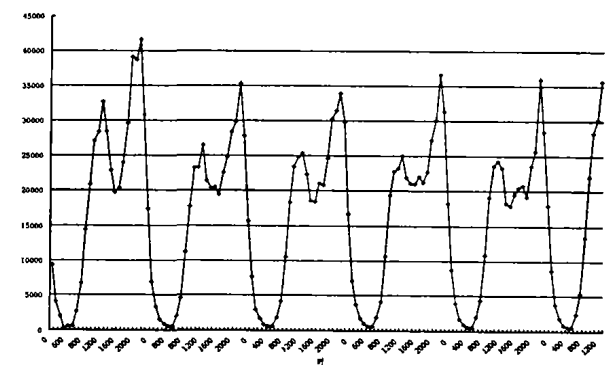


图8 各时段的话务量统计图

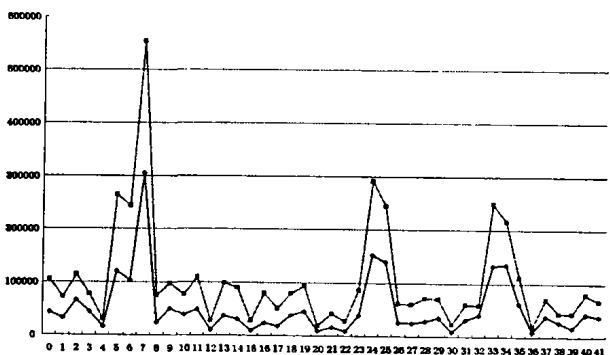


图9 各交换局发出/接收话务量统计图

(3) 2002年1月22日 3:00—24:00

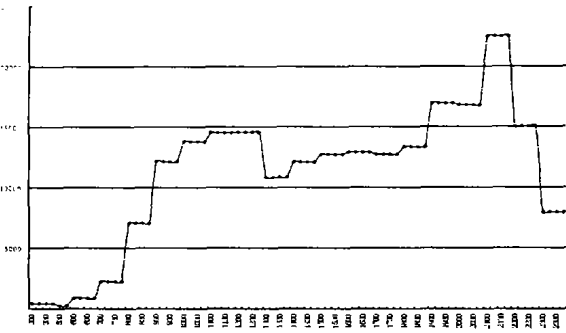


图10 各时段的话务量统计图

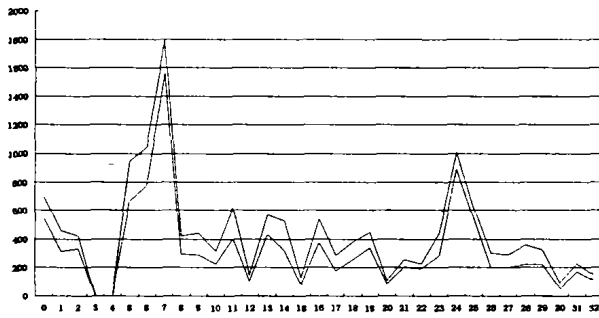


图11 各交换局发出\接收话务量统计图

4.3 算法评估分析

4.3.1 评估环境介绍 评估环境采用了一台 Enterprise 3500 Server (E3500-C62,512M/8GB)和四台 SUN Ultra 10 Workstation(WS U10/300,C3SB,256M/4GB)。为保证评估的公平性,E3500 Server 作为数据库服务器,运行 GT-NIMS 系统的系统管理子系统、数据处理子系统、数据接口子系统和后台监控子系统;一台 SUN U10运行路由求解子系统的非线性单纯形算法、遗传算法并统计这两个算法求解时间;一台 SUN U10运行路由求解子系统的 TABU 算法、IDCR 算法并统计这两个算法求解时间;二台 SUN U10分别运行话务仿真/评估子系统分别评估四个算法和固定路由的全网总阻塞率,并为各算法的评估分配独立进程。

4.3.2 评估指标 利用非线性单纯形算法、遗传算法、TABU 算法、IDCR 算法分别求解某一时间的路由表,统计求解路由表时间;对求解得到的路由表和固定路由表,分别利用话务仿真/评估子系统仿真/评估全网总阻塞率。

4.3.3 评估结果比较分析

(1)2002年2月21日 11:00—28日 23:00

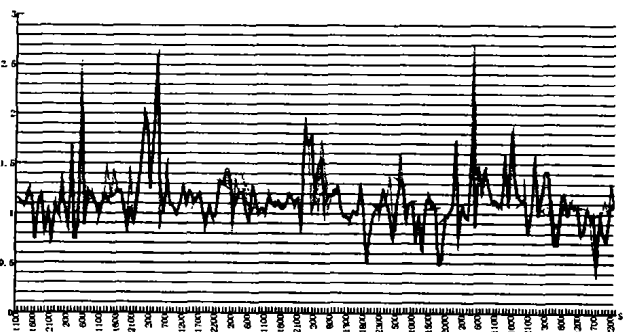


图12 各算法阻塞率对比图

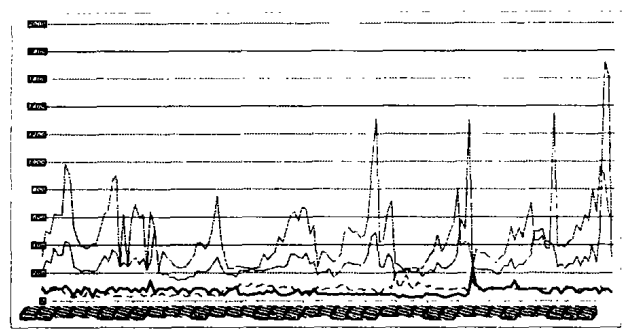


图13 各算法计算时间对比图

(2)2002年5月1日0—5月6日 23:00

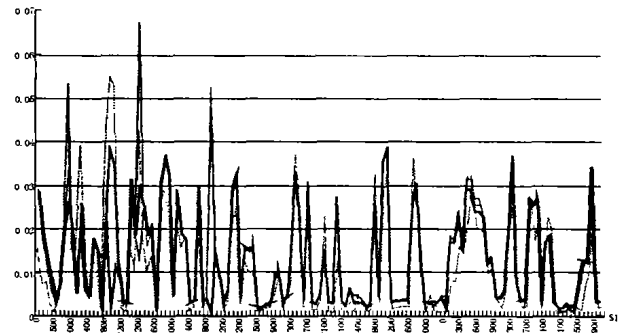


图14 各算法阻塞率对比图

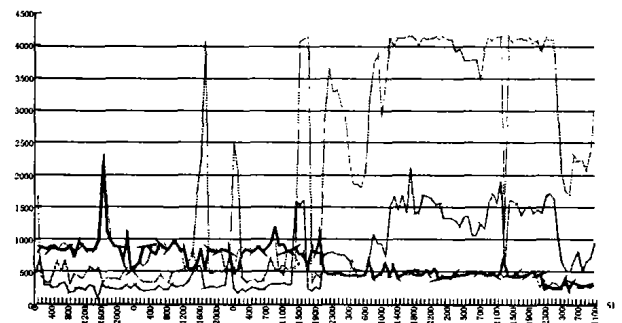


图15 各算法计算时间对比图

(3)2002年1月21日 3:00—24:00(周日)

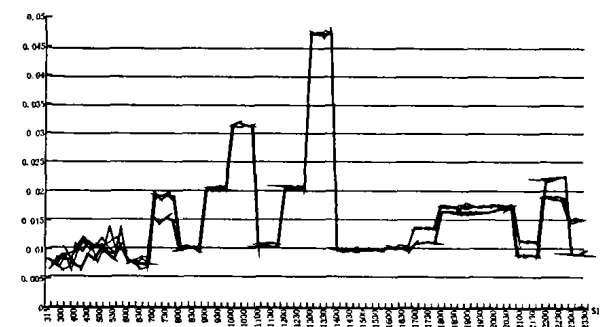


图16 各算法阻塞率对比图

(3)评估结果表

将以上评估结果统计如表1。

4.3.4 数据分析结论 综合分析以上计算模拟结果,有以下结论:

1) 所研究的四种算法:非线性单纯形算法、遗传算法、TABU 算法、IDCR 算法均可以有效求解路由表,所求解路由表的质量要优于固定路由;

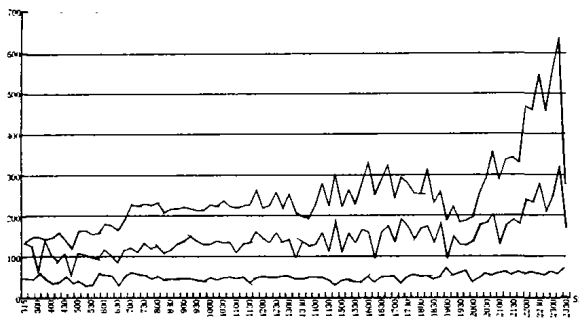


图17 各算法计算时间对比图

表1 评估结果统计表

统计时间	指标	单纯形法	遗传算法	IDCR 算法	TABU 算法	固定路由
忙时 2002/5/1 0—5/6 23:00	全网平均阻塞率	0.029471	0.033384	0.033786	0.03173	0.033838
	平均计算时间	42.5	263.2	48.7	219.7	
闲时 2002/2/21 11:00—2/28 23:00	全网平均阻塞率	0.012156	0.013475	0.013938	0.012681	0.01398
	平均计算时间	36.4	259.0	39.1	194.3	
忙时 2002/1/21 3:00—24:00	全网平均阻塞率	0.015449	0.015942	0.015971	0.015922	0.016014
	平均计算时间	48.1	249.6	49.6	145.1	

结论 本文讨论了无级电话网动态路由选择的问题,重点分析了我们所研究实现的:非线性单纯形算法、遗传算法、Tabu 算法、IDCR 算法 四个路由选择算法,并进行了对比实验分析。

我们研究的遗传算法、Tabu 算法能有效地得到路由方案,但在运算时间上略显得过长,IDCR 算法虽然运算时间较短,但得到的路由方案不甚理想;非线性单纯形算法能够快速得到良好的结果,但当问题空间复杂时,求解质量有下降。

我们开发的“广东省电话网智能管理系统”在广东省电话中得到了实际的应用,本文通过模拟仿真程序评估各算法的具体应用效果,评估算法运行效率与性能。则在“广东省电话网智能管理系统”的实际运行中可以提供多个有效的参考方案,为路由选择提供客观的依据。并为进一步研究动态路由策略有极大的指导意义。

现在,我们解决的问题的模型还是比较简单的。如果把电话网的一些实际情况(如对每一条链路,不仅提供一个路由,而且还提供2到3个备选路由;或者算法不仅提供路由表,而且还要对转接话务进行调配等等)考虑在内的话,问题的复杂度就更大,状态空间更加复杂。还可研究系统预测功能,系统自主学习机制等。如何解决这些问题,是我们今后继续深入研究的方向。

致谢 在此,我们向对本文的工作给予支持和建议的同行,尤其广东省电话网智能管理系统项目组的老师、同学表示诚挚的感谢。

2) 所研究的四种算法均能在五分钟之内完成计算的要求,可实现动态路由;

3) 其中非线性单纯形算法、IDCR 算法由于采用枚举的方法,速度较快;遗传算法、TABU 算法由于采用的是随机算法,速度较慢;

4) 从求解路由表质量来看,非线性单纯形算法提供了一个速度最快的实时动态路由的解决办法,但当问题空间复杂时,求解质量有下降;IDCR 算法求解结果不稳定,会有波动;遗传算法和 TABU 算法在空间规模不太大时,求解效果较好。

参考文献

- 1 Ash G R, Kafker A H, Krishnan K R. Servicing and Real-Time Control of Networks with Dynamic Routing. Bell System Technical Journal, Oct. 1981. 60(8)
- 2 Ash G R. Design and Control of Networks with Dynamic Nonhierarchical Routing. IEEE Communications Magazine, Oct. 1990. 35~40
- 3 Régnier J. Dynamically Controlled Routing in Networks with Non-DCR-Compliant Switches. IEEE Communications Magazine, July 1995
- 4 Kawashima K. State- and Time-Dependent Routing in the NTT Network. IEEE Communications Magazine, July 1995
- 5 邵亮,李磊,陈嫦娟. Tabu 搜索算法在电话网智能管理中的应用. 软件学报, 2002, 13(8): 1705~1709
- 6 邵亮,李磊,张赞波. 评估无级电话网选路方案的一个解析模型. 通信学报, 2003, 24(2): 37~43
- 7 张赞波. 电话网的动态路由问题及遗传算法在其中的应用: [硕士学位论文]. 广州: 中山大学软件研究所, 2001
- 8 张赞波,李磊. 基于遗传算法的电话动态路由策略. 南京大学学报(自然科学), 2000, 36(11): 136~142
- 9 王海,李磊,陈嫦娟. 广东省长话网呼损原因分析及解决办法: [硕士学位论文]. 广州: 中山大学软件研究所, 1999
- 10 广东省电话网智能管理系统项目组: [广东省电话网智能管理系统技术报告]. 广州: 中山大学软件研究所, 2002
- 11 费培之. 线性和非线性规划引论及其应用. 四川大学出版社, 1989
- 12 Wismer D A, Chattergy R. 非线性最优化引论. 北京工业学院出版社, 1987