

基于无向图的专家群体决策会诊调度算法^①

徐 儒^{1,2}, 何中市¹, 徐泽同³

1. 重庆大学 计算机学院, 重庆 400030; 2. 长江师范学院 数学与计算机学院, 重庆 408100;
3. 中国科学院 数学研究所, 北京 100080

摘要: 传统 CAD 的专家调度算法, 大都集中在专家和患者一对一的会诊模式, 或不同领域专家和患者多对一的会诊模式方面的研究, 很少涉及对相同领域专家与患者多对一的会诊调度研究. 针对这一问题, 提出了群体决策思想的多专家会诊调度算法(MESDA), 调度相同领域内的多位专家, 为同一患者展开群体会诊. 算法采用贪心策略, 一方面考虑并行会诊的数量达到最大, 另一方面考虑每一群诊项的专家成员数达到最大, 通过滑动参数控制, 从局部最优出发, 逐步解决并行会诊和专家成员数的优化问题. 实验结果表明: 算法能够解决现实问题中相同领域专家和患者多对一的群体决策会诊的调度问题.

关键词: 多专家会诊; 群体决策; 贪心策略; 无向图

中图分类号: TP312

文献标志码: A

在传统的计算机辅助诊断 CAD(Computer Aided Dignosis)中, 大都采用专家和患者一对一的会诊服务模式, 来为患者展开会诊. 但在实际的医疗会诊中, 一种疾病往往伴随着多个病因, 病情通常呈多样化的趋势, 针对这种情况, 就需要多位医学专家同时为患者进行群体会诊.

现阶段, 关于专家调度会诊的研究成果, 大都集中在专家和患者一对一的调度会诊服务方面, 部分关于多专家调度会诊的研究, 也都仅限于对不同领域的医学专家进行调度, 虽然这也是一种多专家群体会诊, 但将可能涉及到的医学领域会增多, 而采用的是各医学领域专家轮流会诊的模式, 每个领域的专家各不相同, 针对单个领域的专家而言, 其实质也是变相的专家和患者一对一的会诊服务模式, 这种针对不同领域的专家进行的调度, 只解决了会诊中的部分问题.

目前, 很少涉及对相同医学领域的专家进行调度, 实现专家和患者之间真正多对一的群体会诊的研究, 针对某些特殊疾病, 需要相同领域的多位专家, 同时为患者进行会诊. 本文提出了运用群体决策的思想, 来调度同一领域的多位专家, 实现专家和患者多对一的会诊, 从而提高疾病的诊断效率, 为病患者赢得就医的时间.

1 问题描述

1.1 群诊项

设 $I = \{I_1, I_2, I_3, \dots, I_n\}$ 表示 n 个专家, $C = \{C_1, C_2, C_3, \dots, C_k\}$ 是 I 中涉及到的 k 个医学领域, $D = \{D_1, D_2, D_3, \dots, D_m\}$ 是 m 个由患者-专家-病情类别构成的群体决策会诊项, 简称群诊项, 由 3 个域组成: 专家域 D_i_Doctor 、患者域 $D_i_Patient$ 、病情类别域 D_i_LingYu , $D_i_Doctor \in I$ 且 $D_i_Patient \in C$.

定义 1 专家域 D_i_Doctor 由若干专家构成, 专家成员的总量称为 D_i_Doctor 的长度, 用 $Length(D_i_Doctor)$ 表示.

定义 2 对于群诊项 $D_i \in D$, 均有专家成员数 $D_i_Doctor \geq 2$.

① 收稿日期: 2010-02-06

基金项目: 教育部“春晖计划”科研项目(Z2005-1-55003); 重庆市科技攻关项目(CSTC, 2008AB5038).

作者简介: 徐 儒(1982-), 男, 重庆忠县人, 研究生, 研究实习员, 主要从事算法与程序设计、数据库方面的研究.

显然,任何专家的领域类别必须和患者的病情类别相一致,才能为患者就诊,所以,可以把调度多专家为患者会诊的问题,转换为调度专家领域的问题.

1.2 研究的问题

1) 存在专家 I_i 在多个领域 C_1, C_2, \dots, C_x 都具有扎实的理论基础或丰富的临床经验或取得公认的研究成果,换句话说,在医学的多个领域,都称之为专家,因此,针对这类专家必须进行调度控制,使得专家 I_i 一次只能为一位患者进行会诊,避免发生会诊“冲突”.

2) 专家集 I 中,对每一群诊项 D_i 而言,专家域 D_i_Doctor 的长度越大,则专家成员数越多,可以认为患者准确会诊的概率就越高,患者治愈的希望就越大,反之亦然.

3) 近似地认为,群诊项 D_i 中, D_i_Doctor 的长度越小,则专家成员数越少,其它 D_x 可供选择的专家 I_x 就越多,则并行会诊的数量就越大,反之亦然.

为了提高患者会诊的准确率,我们总是希望尽可能多的专家成员,加入到同一组群诊项 D_i 中;为了提高系统并行会诊的数量,我们又总是希望尽可能少的专家成员,加入到同一群诊项 D_i 中,以便更多的 D_i 能够有可供选择的专家.因此,为了解决这一存在问题,必须设计一种算法,既要使得并行会诊的数量最大,同时也要使 D_i 专家成员数 D_i_Doctor 最多,实现两者兼顾的最优调度方案.

2 算法规则

2.1 规则提取

在调度算法中,首先要解决的问题是:明确会诊中有哪些规则,以及对应规则的处理事务.这里,我们考虑如下规则:

规则 1: 任意两个群诊项 D_i 和 D_j 能够并行会诊,则必须满足 $D_i_Doctor \cap D_j_Doctor = \emptyset$, 否则,不能并行会诊;

规则 2: 当领域 C_i 所对应的专家,都处于“会诊”状态时,不能够接受病情类别为 C_i 的新患者,此时,新患者必须进入队列排队等候,直到出现专家空闲时为止.

规则 3: 调度过程中,当同时出现多组群诊项,都能够为患者进行会诊时(即存在多组专家会诊方案),一般采用随机抽取的选择策略,从多组群诊项中任选其一,为患者进行就诊.

规则 4: 存在情况 1: 患者病情特殊,需对群诊项中专家成员的数量要求优先,使得尽可能多的专家能够同时为患者进行会诊;存在情况 2: 患者的数量急剧增多但不紧急,需对并行会诊的数量要求优先,使得每次尽可能多的患者能够同时进行会诊.为此,我们引入滑动参数,来表示调度过程的紧迫程度($0 \leq \epsilon \leq 1$),通过 ϵ 控制单位群诊项中专家域 D_i_Doctor 的长度,从而调解情况 1 和情况 2 中的矛盾问题.

以满负载时的会诊事务数 T 为基数,当患者总人数 $W_1 > T$ 且 W_1 逐渐增加时, ϵ 逐渐向 0 递减,则调度策略优先考虑并行会诊的数量;当患者总人数 $W_1 < T$ 且 W_1 逐渐减少时, ϵ 逐渐向 1 递增,则调度算法优先考虑 D_i 中专家域 D_i_Doctor 的长度,即专家的成员数量;默认情况下,紧迫程度系数 $\epsilon = 0$ 表示并行会诊优先. ϵ 由紧迫度函数 $F_Slide()$ 产生,则 $F_Slide()$ 定义如下:

$$F_Slide() = \begin{cases} 1, & w_1 \leq T \\ \frac{T}{W_1}, & W_1 > T \end{cases}$$

根据紧迫度 ϵ , 调节专家域中滑动专家成员的数量,用 ΔD 表示. ΔD 由滑动专家函数 $Expert_Num()$ 产生,其算法如下:

- 1) 调用紧迫度函数,获取滑动参数 ϵ ;
- 2) 扫描集合 D , 产生满足条件的群诊项事务集合 D' ;
- 3) 统计 D'_i 专家域的长度 $Length(D'_i_Doctor)$, 获取 $Max[Length(D'_i_Doctor)]$ 及对应的 D'_i , 获取 $Min[Length(D'_j_Doctor)]$ 及对应的 D'_j ;
- 4) 计算 ΔD : $Max[Length(D'_i_Doctor)]$ 所对应的 D'_i , 则表示专家成员数最多即并行会诊最不好的情况; $Min[Length(D'_j_Doctor)]$ 所对应的 D'_j , 则表示并行会诊最好的情况; $Max[Length(D'_i_Doctor)] - Min[Length(D'_j_Doctor)]$ 表示滑动参数 ϵ 实际可以滑动调节的专家成员的总数, 则 ΔD

$= \lfloor \epsilon * \{ \text{Max}[\text{Length}(D'_i_\text{Doctor})] - \text{Min}[\text{Length}(D'_j_ \text{Doctor})] \} \rfloor$ 表示经过滑动调节后, 当前最佳滑动的专家成员数;

5) $\text{Min}[\text{Length}(D'_j_ \text{Doctor})] + \Delta D$ 所对应的群诊项, 即为当前的结果。

方法描述为:

```
Int Expert_Num(char * D, int  $\epsilon$ )
{ // 满足条件的群诊项集合  $D$  和  $\epsilon$ 
  for( $i = 0$ ;  $i$  小于获取  $D$  集合中的群诊项的个数;  $i++$ )
    int Max = length( $D_i\_ \text{Doctor}$ ); // 获取专家域长度最大的值
    int Min = length( $D_i\_ \text{Doctor}$ ); // 获取专家域长度最小的值
  Int  $k = \text{Min} + \lfloor \epsilon * [\text{Max} - \text{Min}] \rfloor$ ; // 计算专家域的长度
  Return  $k, D_k$ ; // 返回专家域的长度和群诊项
}
```

例 1. 假设 $\epsilon = 0.5$, 调度结果得到 4 个可供选择的群诊项 D_1 、 D_2 、 D_3 和 D_4 , 其对应专家域的长度分别为 $\text{Length}(D_1_ \text{Doctor}) = 2$, $\text{Length}(D_2_ \text{Doctor}) = 3$, $\text{Length}(D_3_ \text{Doctor}) = 4$, $\text{Length}(D_4_ \text{Doctor}) = 5$, 需从 4 个群诊项中选择一组 D_i 为患者会诊. 由已知可得: $\text{Max}(\text{Length}(D_4_ \text{Doctor}) = 5$, $\text{Min}(\text{Length}(D_1_ \text{Doctor}) = 2$, 通过 ϵ 和 $\text{Expert_Num}()$ 计算可得 $\Delta D = \lfloor 0.5 * [5 - 2] \rfloor = 1$, 则选择专家域的长度为满足 $\text{Min}(\text{Length}(D_i_ \text{Doctor}) + \Delta D = 3$ 的群诊项, 此时既不是选择并行会诊最优的 D_1 , 也不选择专家成员数最大的 D_4 . 最后结果为: D_2 满足条件, 输出。

规则 5: 当调度过程中, 已经不存在满足滑动专家成员数条件的群诊项时, 专家成员数依次减少, 直到不满足定义 2 时为止。

2.2 运行机制

设计调度算法的流程主要包括 4 个部分, 分别为: 规则提取、参数判定、专家调度和结果优化. x 为初始值输入, $C_e(S)$ 为规则提取模块, r 为判定参数, $Y_e(S)$ 为判定模块, u 为紧迫度系数, $F_e(S)$ 为调度算法模型, $G_p(S)$ 为优化策略, Y 为调度结果输出. 它的运行机制如图 1 所示. 病人或现场主治医师需要会诊时, 向系统发出会诊请求, 通过初始值 x 对会诊请求进行规制提取, 进行 $Y_e(S)$ 资源判定, 通过判定参数 r 确定判定结果. 如 $r=1$ 即系统资源能够满足病人方面的请求, 则进行专家调度. 系统根据紧迫度系数 u , 对多个用户对资源的申请, 进行优化计算, 并将优化的结论与资源申请者协商, 在病人方同意的情况下, 才可能启动会诊活动. 如 $r=0$, 即系统资源不能满足病人方面的请求, 则把原因反馈病人, 病人或放弃会诊、或延迟会诊、或修改会诊条件 x 重新申请。

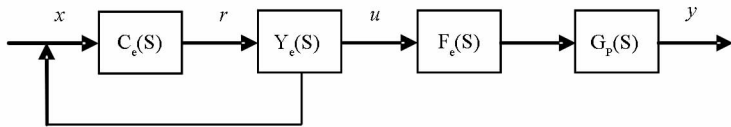


图 1 算法运行机制

3 基于无向图的 MESDA 算法

3.1 无向图结构化表示

我们用无向图 $G = (V, E)$ 来结构化表示群诊项在调度过程中之间的关系, 其中, E 是边集. 从集合 I 中, 将所有可能出现共同会诊的专家, 用穷举的方法表示出群诊项, 在无向图中用 V 表示, 其中, V 是结点集, 每个结点表示一个群诊项 D_i . $V = \{D_i \mid D_i \in D\}$; $E = \{ \langle x, y \rangle \mid x, y \in V \}$ 表示结点之间邻接关系的有穷集合, 若存在一条无向边 $\langle V_i, V_j \rangle$, 则 V_i 和 V_j 互为邻结点, 表示群诊项 D_i 和 D_j 中的专家成员有交集, 即 $D_i_ \text{Doctor} \cap D_j_ \text{Doctor} \neq \emptyset$, 两群诊项不能并行会诊, 否则会引起专家冲突。

3.2 Greedy Algorithm 策略

算法采用 Greedy Algorithm 策略, 是从问题的某一个初始解出发逐步逼近给定的目标, 以尽可能快地获得更好的解, 当达到算法中的某一步不能再继续前进时, 算法停止. 首先从 V 中, 任意选择 D_i 作为初始

结点,从 D 中获取与 D_i 不相邻的所有结点集合并删除,更新集合 V ,继续从 V 中选择新结点 D_{i+1} 作为下一个初始结点,从 D 中获取与 D_{i+1} 不相邻的所有结点集合并删除;重复操作,直到集合元素 $V = \emptyset$ 时停止,则每次选择的初始结点所构成的集合,即为调度算法求解的最后结果。

算法过程描述如下:

算法 2: 多专家会诊调度算法(Scheduling Algorithm for Multi-expert Diagnosis)

输入: I

输出: D_i

过程:

- 1) 初始化: 将 I 结构化表示,根据无向图 $G = (V, E)$ 建立集合 D ;
- 2) 调用: 调用 $F_Slide()$, 获取滑动参数 ϵ ; 调用 $Expert_Num()$, 获取专家域的长度;
- 3) 选择: 从 D 集合中,选择满足专家域长度的任意一结点 D_i 作为首结点;
- 4) 遍历: 遍历 D 集合,寻找所有与 D_i 相邻的结点;
- 5) 输出: 输出首结点 D_i ;
- 6) 枝剪: 将 D_i 和与 D_i 相邻的所有结点,从 D 中删除,更新 G ;
- 7) 判断: 如果 $D = \emptyset$,跳转到第 9 步骤;
- 8) 迭代: 更新 G ,跳转到第 3 步骤,重复迭代操作;
- 9) 结束.

4 算 例

设专家的医学领域集合为 $C = \{\text{皮肤科, 骨科, 肝脏科, 泌尿科, 肿瘤科}\}$, 专家为 $X_1, X_2, X_3, X_4, X_5, X_6, X_7, \epsilon = 0.5$. 专家和领域之间的关系如表 1 所示.

表 1 专家和领域之间的关系

专家	医学领域	专家	医学领域
X_1	皮肤科	X_3	肿瘤科
X_1	骨科	X_4	肝脏科
X_1	肝脏科	X_5	肝脏科
X_2	皮肤科	X_5	肿瘤科
X_2	泌尿科	X_6	皮肤科
X_3	骨 科	X_6	肿瘤科
X_3	泌尿科	X_7	泌尿科

根据算法 MESDA 和表 1 建立起用无向图表示的结构化调度模型. 其中,每个结点表示一个 D_i , 例如,由结点 $\langle X_1, X_2, X_6 \rangle$ 可得: $D_i_Doctor = \{X_1, X_2, X_6\}$, $D_i_LingYu = \text{“皮肤科”}$, $D_i_Patient = \text{“皮肤患者”}$; 表示 3 位专家 X_1, X_2 和 X_6 共同为皮肤患者展开群体决策会诊, 依此类推.

无向图 G 初始结点数 $V = 17, E = 95$, 选取 V 中任意结点 $X_1 X_6$ 作为首结点, 将相邻 $X_1 X_6$ 的所有结点提取出来(如图 2 所示), 这些结点即为不能够并行会诊的群诊项结点, 并将这些结点从 G 中删除, 更新 G 中 V 结点, 此时 $V = 6$ 不满足结束条件, 执行第 2 次调度, 继续从 V 中选择一结点作为首结点, 因为 $\epsilon = 0.5$ 表示专家成员数优先, 因此选择专家成员数最大的结点 $X_2 X_3 X_7$ 作为首结点, 将相邻 $X_2 X_3 X_7$ 的所有结点提取出来(如图 3 所示), 并将这些结点从 G 中删除, 再次更新 G 中 V 结点,

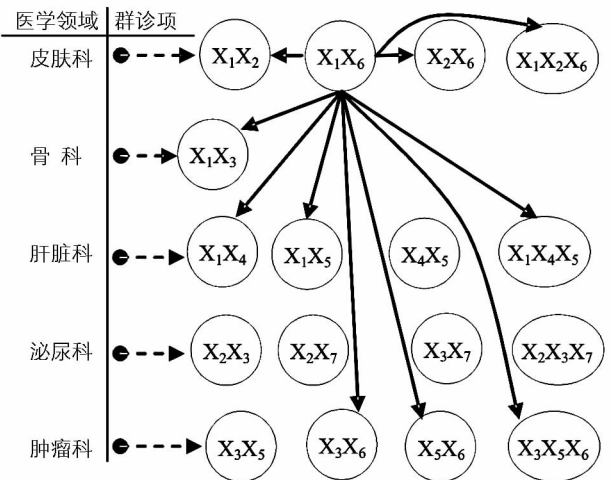


图 2 第 1 次调度过程

此时 $V=1$ 仍不满足结束条件, 执行第 3 次调度, V 中只有唯一结点 X_4X_5 , 则选择其为首结点并从 G 中删除。

调度全过程由 3 次循环操作完成, 则表示有 3 个群诊项可以并行会诊, 每次循环操作产生一个首选结点(如表 2 所示), 分别是 X_1X_6 、 $X_2X_3X_7$ 、 X_4X_5 , 表示为: 专家 X_1 和 X_6 共同为皮肤患者进行会诊, 专家 X_2 、 X_3 和 X_7 共同为泌尿患者进行会诊, 专家 X_4 和 X_5 共同为肝脏患者进行会诊。

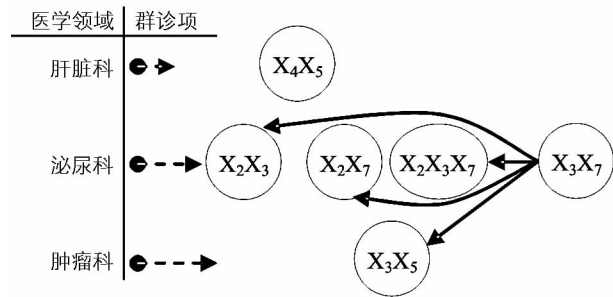


图 3 第 2 次调度过程

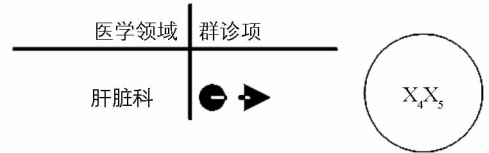


图 4 第 3 次调度过程

表 2 调度过程和结果

次数	首结点	枝剪结点	保留结点
1	X_1X_6	X_1X_2 ; X_2X_6 ; $X_1X_2X_6$; X_1X_3 ; X_1X_4 ; X_1X_5 ; $X_1X_4X_5$; X_3X_6 ; X_3X_6 ; $X_3X_5X_6$	X_4X_5 ; X_2X_3 ; X_2X_7 ; X_3X_7 ; $X_2X_3X_7$; X_3X_5
2	X_3X_7	X_2X_3 ; X_2X_7 ; $X_2X_3X_7$; X_3X_5	X_4X_5
3	X_4X_5	X_4X_5	NULL

执行算法调度之后, 实现了对皮肤科、肝脏科和泌尿科患者的调度会诊, 此时, 由于专家成员均无空闲, 所以, 骨科和肿瘤科患者暂时进入等待队列, 在专家空闲时发出调度请求, 并在下一轮调度过程中, 首先从骨科或肿瘤科患者的专家成员中, 选择首结点. 由算法调度可得, 下一轮可以并行会诊的群诊项有 X_1X_3 、 X_5X_6 、 X_2X_7 .

假设 $\epsilon = 1$ 时, 表示并行会诊优先, 则在执行第 2 次调度中, 不应该选择专家成员数最大的 $X_2X_3X_7$ 结点作为首结点, 而应该选择专家成员数最小的结点, 例如 X_4X_5 等。

5 结 论

该算法主要针对相同领域的多位专家同时为患者会诊的问题, 提出了专家对患者多对一的会诊模式, 真正意义上实现群体的会诊模式, 解决了传统意义上专家与患者一对一的会诊问题或多领域专家与患者轮流一对一的会诊问题, 特别是在存在多个领域都是专家的情况. 采用滑动参数控制的方式, 解决了并行会诊数量达到最大和专家成员数最多的冲突问题, 从而得出专家和患者之间的最佳会诊策略, 实现两者均优的会诊调度, 这种情况在远程医疗系统中比较常见。

该算法在多个领域都是在专家成员数较多的情况下, 能够得出较好的调度结果, 并且随着专家数量的增大, 算法优势越明显, 在专家成员相对较少的情况下, 算法的时间开销将会增大。

参考文献:

- [1] Industry Seminar of Transportation. Intelligent Transport System [M]. Beijing: The Press of the People's Transport, 2000.
- [2] YAO Ya-fu, LIU Kan. Genetic Neural Network Based Traffic Flow Forecasting Research [J]. Highways & Automotive Applications, 2007, 6: 282.
- [3] 廖明宏. 数据结构与算法 [M]. 北京: 高等教育出版社, 2007: 4-20.
- [4] 邢欣来, 何中市, 王 健. 组合分类器辅助诊断肺栓塞的研究 [J]. 计算机工程与应用, 2009, 45(12): 214-217.
- [5] 覃仁超, 李 涛, 张 瑜. 基于免疫原理的控制器参数整定方法 [J]. 电子科技大学学报, 2009(7): 609-612.
- [6] 肖业伟, 黄辉先, 王宸昊. 蚁群算法的交叉路口多相位信号配时优化 [J]. 计算机工程与应用, 2008, 44(19):

241—244.

- [7] 陈琳, 刘翔, 孙优贤. 单交叉路口交通流的通用多相位智能控制策略 [J]. 浙江大学学报: 工学版, 2006, 40(11): 1947—1950.

Expert Group Diagnosis Scheduling Algorithm Based on Undirected Graph

XU Ru^{1,2}, HE Zhong-shi¹, XU Ze-tong³

1. College of Computer Science, Chongqing University, Chongqing 400030, China;

2. College of Mathematics & Computer Science, Yangtze Normal University, Chongqing 408100, China;

3. Institute of Mathematics, Chinese Academy of Sciences, Beijing 100080, China

Abstract: Traditional CAD expert scheduling algorithm, is concentrated on one to one consultation of experts and patient model, or a different domain experts and the consultation patterns of patients with many to one study, very few experts in the field involving the same patients with many to one of consultation with Scheduling. In view of this problem, it is proposed that the scheduling algorithm for multi-expert diagnosis (SAMED), scheduling the same number of experts in the field, starting for the same patient group consultation. Algorithm uses the greedy strategy, on the one hand considering the maximum number of parallel consultations, on the other hand considering every member of a group of experts attending items to maximize the number, by sliding the parameter control, starting from a local optimum, and gradually solve the parallel consultation and expert membership optimization problem. Experimental results show that: the algorithm can solve practical problems, the same experts in the field, and consultation in patients with many to one of the group decision problem.

Key words: multi-expert diagnosis; group decision-making; greedy algorithm; undirected graph

责任编辑 汤振金