发明名称
一种基于考虑城市交叉口时间延误的实用路径选择方法

摘要
本发明公开一种基于考虑城市交叉口时间延误的实用路径选择方法。通过借鉴弧标号的思想对传统 A*算法进行改进，解决了路径规划问题中的交叉口转向时间延误问题。根据实时交通状况对路网数据进行更新，并在最优路径的选择过程中限制搜索区域，以路段行程时间变化率和交叉口延误时间变化率为依据有选择性地更新最优路径。本发明方法在考虑交叉口时间延误的基础上，可以根据实时路况为车辆提供合适的路径选择，实用性较高，对交通诱导路径规划的研究有重要意义。
1. 一种基于考虑城市交叉口时间延误的实用路径选择方法，其特征在于该方法包括以下步骤：

步骤 (1). 对待测实区域采集路网数据，利用其生成路网，并用有向赋权网络 \( G = (V, A, D, C) \) 表示该路网；

其中 \( V = \{v_i | i = 1, 2, \ldots, n\} \) 为网络 \( G = (V, A, D, C) \) 的节点集合，表示城市道路交叉口；\( A = \{a_{ij} | i, j = 1, 2, \ldots, n\} \) 为网络 \( G = (V, A, D, C) \) 的弧集合，表示城市道路相互交叉口之间的有向路段；\( C = \{c_{ij} | i, j = 1, 2, \ldots, n\} \) 为网络 \( G = (V, A, D, C) \) 的弧权集合，表示车辆在弧 \( a_{ij} \) 上平均路段行程时间；\( D = \{d_{ijk} | i, j, k = 1, 2, \ldots, n\} \) 为网络 \( G = (V, A, D, C) \) 的点权集合，\( d_{ijk} \) 表示弧 \( a_{ij} \) 转向弧 \( a_{jk} \) 时在节点 \( v_j \) 处产生的平均转向延误时间；

所述的路网数据包括交叉口地理坐标，相邻交叉口间平均路段行程时间以及交叉口平均转向延误时间；

步骤 (2). 设定最优路径的起点和终点，根据当前路网数据，应用改进启发式 \( A^* \) 算法计算得到最优路径；

所述的最优路径起点为节点 \( v_o \)，终点为节点 \( v_d \)；最优路径由多条首尾相连的弧 \( a_{ij} \) 构成，其中弧 \( a_{ij} \) 上的节点 \( v_i \) 和节点 \( v_j \) 分别为该弧的尾节点和头节点；

2.1. 考察终点节点 \( v_d \) 所有入弧，建立目标弧集 \( T \)；目标弧集 \( T \) 存放以终点 \( v_d \) 为头节点的所有入弧，初始化 \( T_{OPEN} = T, T_{CLOSE} = \emptyset \)；

2.2. 初始化估价值 \( f_{ij} \)，令 \( f_{ij} = M, M \) 为无穷大正数；

所述的估价值 \( f_{ij} \) 表示弧 \( a_{ij} \) 上产生的起点 \( v_o \) 至终点 \( v_d \) 的估计行程时间；

2.3. 在起点 \( v_o \) 前添加虚拟节点 \( v_o' \)，则弧 \( a_{o'o} \) 是一条虚拟弧，根据公式 \( f_{o'o} = g_{o'o} + h_{o'o}, g_{o'o} = 0 \)，故 \( f_{o'o} = h_{o'o}, p_{o'o} = NULL \) 将弧 \( a_{o'o} \) 移入 \( OPEN \) 表中；其中对于弧 \( a_{ij} \in A \) 而言，\( p_{ij} \) 为起始 \( v_o \) 至弧 \( a_{ij} \) 头节点 \( v_i \) 的最短路径上弧 \( a_{ij} \) 前弧的尾节点的标号；

2.4. 判断 \( OPEN \) 表是否为非空，若是则执行以下操作 2.4.1～2.4.4，否则执行 2.5；

2.4.1. 对于 \( OPEN \) 表中的所有弧，选取最小 \( f_{ij} \) 值对应弧，记为 \( a_{ij} \)，将弧 \( a_{ij} \) 从 \( OPEN \) 表中删除，并将弧 \( a_{ij} \) 插入到 \( CLOSE \) 表中；判断弧 \( a_{ij} \) 是否在 \( OPEN \) 表中，若是则执行步骤 2.4.2，若则执行步骤 2.4.3；

2.4.2. 若弧 \( a_{ij} \) 在 \( OPEN \) 表中，则将其从 \( OPEN \) 表中删除，并将弧 \( a_{ij} \) 插入到 \( CLOSE \) 表中；然后判断是否 \( T_{OPEN} = \emptyset \)，若是则执行步骤 2.5，若否则跳转执行步骤 2.4；

2.4.3. 若弧 \( a_{ij} \) 不在 \( OPEN \) 表中，判断弧 \( a_{ij} \) 的头节点 \( v_i \) 所有出弧 \( a_{st} \) 是否都在 \( CLOSE \) 表中，若是则执行步骤 2.4，若否则对在 \( OPEN \) 表中的出弧进行考察；然后判断是否成立 \( f_{st} > g_{rs} + d_{rst} + c_{st} + h_{st} \)，若是则重新赋予 \( f_{st} = g_{rs} + d_{rst} + c_{st} + h_{st} \)，\( p_{st} = r \) 后执行步骤 2.4.4，若否则直接执行步骤 2.4.4；

2.4.4 判断弧 \( a_{st} \) 是否在 \( OPEN \) 表中，若不在则将该弧移入 \( OPEN \) 表后跳转执行步骤 2.4；若在则直接跳转执行步骤 2.4；

2.5. 根据 \( T_{CLOSE} \) 表中各弧紧前弧的尾节点标号 \( p_{ij} \)，回溯得起点 \( v_o \) 至该弧头节点的最短路径；比较终点 \( v_d \) 各入弧的估计值 \( f_{ij} \)，取最小估计值对应的终点 \( v_d \) 入弧；该入弧头节点所对应的最短路径即为起点 \( v_o \) 至终点 \( v_d \) 的最优路径；

步骤 (3). 按照步骤 (2) 得到的最优路径行驶，每间隔 Ts 时间后，根据路面检测器检测
到的实时路网数据对路网进行更新，并检测车辆行驶位置；根据车辆当前行驶位置，选取最近的交叉口设定为当前考察节点，记为 \( v_c \)；

所述的最近的交叉口是指路段上沿车辆行程方向距离车辆位置最近的交叉口；

步骤 (4). 判断当前考察节点 \( v_c \) 是否为终点 \( v_d \)，若是则算法终止；若否则执行步骤 (5)；

步骤 (5). 根据当前考察节点 \( v_c \) 和终点 \( v_d \) 确定限制搜索区域 \( Z \)；

所述的限制搜索区域 \( Z \) 是指根据路网中各交叉口的地理坐标，运用一些几何规则，对路网范围进行划分；在最优路径的搜索过程中只对落入限制区域 \( Z \) 内的节点和弧段进行考察，忽略限制搜索区域 \( Z \) 外的节点和弧段；

步骤 (6). 根据当前最优路径所对应的路网数据，计算该时段限制搜索区域 \( Z \) 内各路段的行程时间变化率和各交叉口延误时间变化率，最后取所有变化率的平均值 \( \lambda \) ；

步骤 (7). 判断变化率平均值 \( \lambda \) 是否大于阈值 \( a \) ，若是则跳转执行步骤 (2)，并重新赋予 \( v_c \) 为起点，\( v_d \) 为终点，以上述步骤 (5) 得到的限制搜索区域 \( Z \) 建立新路网；若否则执行步骤 (3)。
一种基于考虑城市交叉口时间延误的实用路径选择方法

技术领域
[0001] 本发明属于交通工程领域，涉及一种基于考虑城市交叉口时间延误的实用路径选择方法。

背景技术
[0002] 为出行车辆提供路径选择是城市交通系统研究的重要组成部分。由于交叉口处信号控制及拥堵造成的排队现象等原因，车辆行驶至交叉口处时将不可避免地产生时间延误。对于车辆出行的整个行程时间而言，交叉口时间延误的比重不可忽略。因此，在进行路径选择时，将交叉口时间延误作为考虑因素之一是十分有必要的。此外，城市交通路网是一个动态变化的复杂系统，交通流参数是实时变化的，路径选择需要依据实时路况信息作出判断，有选择性地重新选择路线。
[0003] 目前，国内外对于路径选择算法的研究已较为成熟，但大多停留在理论阶段，当结合到实际路网中时，这些算法需要进行一些改进才能进行应用。

发明内容
[0004] 本发明的目的是针对现有技术的不足，提供一种基于考虑城市交叉口时间延误的实用路径选择方法。
[0005] 该方法包括以下步骤：
[0006] 步骤（1）：对待测实地区域采集路网数据，利用其生成路网，并用有向赋权网络G = (V, A, D, C) 表示该路网；
[0007] 其中 V = {v_i | i = 1, 2, …, n} 为网络 G = (V, A, D, C) 中的节点集合，表示城市道路交叉口；A = {a_{ij} | i, j = 1, 2, …, n} 为网络 G = (V, A, D, C) 中的弧集合，表示城市道路相邻交叉口之间的有向路段；C = {c_{ij} | i, j = 1, 2, …, n} 为网络 G = (V, A, D, C) 中的弧权集合，c_{ij} 表示车辆在弧 a_{ij} 上平均路段行程时间；D = {d_{ik} | i, j, k = 1, 2, …, n} 为网络 G = (V, A, D, C) 中的点权集合，d_{ik} 表示弧 a_{ij} 转向弧 a_{ik} 时在节点 v_k 处产生的平均转向延误时间；
[0008] 所述的路网数据包括交叉口地理坐标、相邻交叉口间平均路段行程时间以及交叉口平均转向延误时间；
[0009] 步骤（2），设定最优路径的起点和终点，根据当前路网数据，调用改进启发式 A* 算法计算得到最优路径，具体步骤如 2.1 ~ 2.5；
[0010] 所述的最优路径起点为节点 v_s，终点为节点 v_d；最优路径由多条首尾相连的弧 a_{ij} 构成，其中弧 a_{ij} 上的节点 v_i 和节点 v_j 分别为该弧的尾节点和头节点；
[0011] 2.1 考察终点节点 v_d 的所有入弧，建立目标弧集 T；目标弧集 T 存放以终点 V_d 为头节点的所有入弧，初始化 T_OPEN = T，T_CLOSE=∅；
[0012] 2.2 初始化结点值 f_{ij}，令 f_{ij} = M，M 为无穷大正数；
[0013] 所述的结点值 f_{ij} 表示在弧 a_{ij} 上产生的起点 v_i 至终点 v_j 的估计行程时间；
2.3 在起点 $v_0$ 前添加虚拟节点 $v'_0$，则弧 $a_{v'_0}$ 是一条虚拟弧。根据公式 $f_{v'_0} = g_{v'_0} + h_{v'_0}$，$g_{v'_0} = 0$，故 $f_{v'_0} = h_{v'_0}$。将弧 $a_{v'_0}$ 移入 OPEN 表中。对于弧 $a_{ij} \in A$ 而言，$p_{ij}$ 为起点 $v_j$ 至弧 $a_{ij}$ 头节点 $v_i$ 的最短路径上弧 $a_{ij}$ 紧前弧的尾节点的标号。

2.4 判断 OPEN 表是否为空，若是则执行以下操作步骤 2.4.1 ～ 2.4.4，若否则执行步骤 2.5。

2.4.1 对于 OPEN 表中的所有弧，选取最小 $f_{ij}$ 值对应的弧，记为 $a_{rs}$；将 $a_{rs}$ 从 OPEN 表中删除，并将 $a_{rs}$ 插入到 CLOSE 表中；判断弧 $a_{rs}$ 是否在 \text{T\_OPEN} 表中，若是则执行步骤 2.4.2，若否则执行步骤 2.4.3。

2.4.2 若弧 $a_{rs}$ 在 \text{T\_OPEN} 表中，则将其从 \text{T\_OPEN} 表中删除，并将 $a_{rs}$ 插入到 \text{T\_CLOSE} 表中；然后判断是否 $\text{T\_OPEN} = \emptyset$，若是则执行步骤 2.5，若否则跳转执行步骤 2.4。

2.4.3 若弧 $a_{rs}$ 不在 \text{T\_OPEN} 表中，判断弧 $a_{rs}$ 的头节点 $v_s$ 的所有弧 $a_{st}$ 是否都在 CLOSE 表中，若是则执行步骤 2.4，若否则对不在 CLOSE 表中的弧进行考察，然后判断是否成立 $f_{st} = g_{st} + d_{st} + c_{st} + h_{st}$，若是则重新赋子 $f_{st} = g_{st} + d_{st} + c_{st} + h_{st}$，$p_{st} = r$ 后执行步骤 2.4.4；若否则直接执行步骤 2.4.4。

2.4.4 判弧 $a_{rs}$ 是否在 OPEN 表中，若不在则将该弧移入 OPEN 表后跳转执行步骤 2.4；若在则直接跳转执行步骤 2.4。

2.5 根据 \text{T\_CLOSE} 表中各弧紧前弧的尾节点标号 $p_{ij}$，回溯得起点 $v_0$ 至该弧头节点的最短路径；比较终点 $v_q$ 各弧的估计值 $f_{i}$，取最小估计值对应的终点 $v_q$ 入弧；该弧头节点所对应的最短路径即为起点 $v_0$ 至终点 $v_q$ 的最优路径。

步骤 (3)。按照步骤 (2) 得到的最优路径行驶，每隔 $T_s$ 时间后，根据路面检测器检测到的实时路网数据对路网进行更新，并检测车辆行驶位置；根据车辆当前行驶位置，选取最近的交叉口设定为当前考察节点，记为 $v_0$；

步骤 (4)。判断当前考察节点 $v_i$ 是否为终点 $v_q$，若是则算法终止；若否则执行步骤 (5)；

步骤 (5)。根据当前考察节点 $v_i$ 和终点 $v_q$ 确定限制搜索区域 $Z$；

步骤 (6)。根据当前最优路径所对应的路网数据，计算该时段限制搜索区域 $Z$ 内各路段的行驶时间变化率和各交叉口延误变化率，最后取上述所有变化率的平均值 $v_{av}$；

步骤 (7)。判断变化率平均值 $v_{av}$ 是否大于阈值 $a_{av}$，若是则跳转执行步骤 (2) 并重新赋予 $v_i$ 为起点，$v_q$ 为终点，以上述步骤 (5) 得到的限制搜索区域 $Z$ 建立新路网；若否则执行步骤 (3)。

本发明的有益效果是：

应用本发明方法可以解决城市道路路径规划中的交叉口转向时间延误问题，并能根据实时路网数据为出行车辆提供符合实际交通状况的路径选择，对城市道路交通诱导体系的研究具有重要的应用价值。
附图说明
[0030] 图1是实际路网转换为有向赋值网络G = (V, A, D, C)的示例图，其中图(a)为实际交叉口路网示意图（有两个交叉口），图(b)为抽象后的有向赋值网络示意图；
[0031] 图2是弧和节点关系示意图；
[0032] 图3是本发明方法流程图。

具体实施方式
[0033] 下面结合附图对本发明的技术方案作进一步说明。
[0034] 本发明方法包括以下步骤：
[0035] 步骤(1)。通过现有成熟技术对待测实际区域采集路网数据，利用其生成路网，并用有向赋值网络G = (V, A, D, C)表示该路网，如图1所示；其中V = {v_i | i = 1, 2, ..., n}为网络G = (V, A, D, C)中的节点集合，表示城市道路交叉口；A = {a_ij | i, j = 1, 2, ..., n}为网络G = (V, A, D, C)中的弧集合，表示城市道路相邻交叉口之间的有向路段；C = {c_ij | i, j = 1, 2, ..., n}为网络G = (V, A, D, C)中的赋值集合，c_ij表示车辆在弧a_ij上平均路段行程时间；当考虑交叉口转向延迟时，有向网络的节点带有权重，且权重不唯一，与节点相邻的弧段有关，表示的是与节点前后相邻的节点在该节点产生的时间延迟，D = {d_ijk | i, j, k = 1, 2, ..., n}为网络G = (V, A, D, C)中的延迟集合，d_ijk表示弧a_ij转向弧a_ik时在节点v_j处产生的平均转向延迟时间。
[0036] 所述的路网数据包括交叉口地理坐标、相邻交叉口间平均路段行程时间以及交叉口平均转向延迟时间；
[0037] 步骤(2)。设定最优路径的起点和终点，根据当前路网数据，调用改进启发式A*算法计算得到最优路径；
[0038] 所述的最优路径起点为节点v_s，终点为节点v_d，最优路径由多条首尾相连的弧a_ij构成，其中弧a_ij上的节点v_i和节点v_j分别为该弧的尾节点和头节点；
[0039] 2.1 考察终点节点v_d的所有入弧，建立目标弧集T；目标弧集T存放以终点v_d为头节点的所有入弧，初始化T OPEN = T，T CLOSE = ∅；实际上，在本算法中，T OPEN 表用于存放未移入CLOSE表中的目标弧集T中的弧，T CLOSE 表用于存放已经移入到CLOSE表中的目标弧集T中的弧，OPEN表存放已经搜索过的但尚未完成考察的弧，CLOSE表用于存放已经完成考察不需要再搜索的弧。
[0040] 所述的入弧是指网络G = (V, A, D, C)中所有以节点v_s为头节点的弧；
[0041] 2.2 初始化估价值f_s，令f_s = M，M为无穷大正数；
[0042] 所述的估价值f_s表示在弧a_s上产生的起点v_s至终点v_d的估计行程时间；
[0043] 实际上，在本算法中f_s = g_s + h_s，其中g_s为起点v_s至弧a_s头节点v_j最短路径的实际行程时间，h_s为弧a_s头节点v_j至终点v_d的估计行程时间。本发明实施例h_s取弧a_ij头节点v_j至终点v_d的欧氏距离除以当前路网中车辆平均速度。
[0044] 2.3 在起点v_s前添加虚拟节点v_0，弧a_0s是一条虚拟弧，根据公式f_0s = g_0s + h_0s，g_0s = 0，故f_0s = h_0s。p_0s = NULL，将弧a_0s移入OPEN表中，其中对于弧a_ij ∈ A而言，p_ij为起点v_i至弧a_ij头节点v_j的最短路径上弧a_ij紧前弧的尾节点的标号，如图2所示；
2.4 判断 OPEN 表是否为非空，若是则执行以下操作 2.4.1 至 2.4.4，若否则执行 2.5；

2.4.1 对于 OPEN 表中的所有弧，选取最小 $f_{ij}$ 值对应的弧，记为 $a_{rs}$；将 $a_{rs}$ 从 OPEN 表中删除，并将 $a_{rs}$ 插入到 CLOSE 表中。判断弧 $a_{rs}$ 是否在 T_OPEN 表中，若是则执行步骤 2.4.2，否则执行步骤 2.4.3；

2.4.2 若弧 $a_{rs}$ 在 T_OPEN 表中，则将其从 T_OPEN 表中删除，并将 $a_{rs}$ 插入到 T_CLOSE 表中，然后判断是否 $T_{OPEN} = \emptyset$，若是则执行 2.5，若否则跳转执行 2.4；

2.4.3 若弧 $a_{rs}$ 不在 T_OPEN 表中，判断弧 $a_{rs}$ 的头节点 $v_s$ 的所有出弧 $a_{st}$ 是否都在 CLOSE 表中。若是则执行步骤 2.4，否则则在在 T_CLOSE 表中的出弧先进行考察，如果，判断是否成立

$$f_{st} > g_{rst} + c_{st} + h_{st}$$

若是则重新赋予

$$f_{st} = g_{rst} + d_{rst} + c_{st} + h_{st}, p_{st} = r$$

然后执行步骤 2.4.4；若否则直接执行步骤 2.4.4；

2.4.4 判断弧 $a_{rs}$ 是否在 OPEN 表中，若不在则将该弧移入 OPEN 表中，然后跳转执行步骤 2.4；若在则直接跳转执行步骤 2.4；

2.5 根据 T_CLOSE 表中各弧紧前弧的尾节点标号 $p_{ij}$，回溯得起点 $v_s$ 至该弧头节点的最短路径；比较终点 $vd$ 各入弧的估计值 $f_{ij}$，取最小估计值对应的终点 $v_d$ 入弧；该入弧头节点所对应的最短路径即为起点 $v_s$ 至终点 $v_d$ 的最优路径。

步骤 (3)。按照步骤 (2) 得到的最优路径行驶，每次时间隔后，根据路面检测器检测到的实时路网数据对路网进行更新，并检测车辆行驶位置；根据车辆当前位置及各交叉口设置为当前考察节点，记为 $v_c$。

所述的 T_s 是指以分钟为单位的时间间隔，一般取 5～10min，根据路面交通流检测器的检测周期进行确定。

所述的最近的交叉口是指路段上沿车辆行驶方向距离车辆位置最近的交叉口。如果车辆当前位置位于路段上的非交叉口处，则该最近交叉口是指车辆将要到达的下一个交叉口；如果车辆当前位置处于交叉口处，则该最近交叉口是指当前交叉口。

步骤 (4)。判断当前考察节点 $v_c$ 是否为终点 $v_d$，若是则算法终止；若否则执行步骤 (5)；

步骤 (5)。根据当前考察节点 $v_c$ 和终点 $v_d$ 确定限制搜索区域 Z；

所述的限制搜索区域 Z 是指根据路网中各交叉口的地理坐标，运用一些几何规则，对路网进行划分，其中最优路径的选择过程只对位于限制区域 Z 内的节点和弧段进行覆盖，消除搜索区域 Z 外的节点和弧段。这样可以控制最优路径的搜索方向，防止搜索无效路径，提高搜索效率。

本专利实施例采用路线算法来限制搜索区域，根据当前考察节点 $v_c$ 和终点 $v_d$ 的平面坐标，结合路线坐标公式，根据本领域技术人员经验建立一个路线范围，即限制搜索区域 Z。

所述的路线算法是依据路线是平面内到两个焦点的距离之和是常数的点的轨迹，以起点和终点为焦点建立的路线区域，则该区域内部任意点到起点和终点的距离之和限定在一个常数范围内，这样给起点和终点之间路径的长度规定了一个上限，从而到达限制的目的。

步骤 (6)。根据当前最优路径所对应的路网数据，计算该时段限制搜索区域内各
路段的行程时间变化率和各交叉口延误时间变化率，最后取所有变化率的平均值 $\lambda$；

[0060] 所述的当前最优路径所对应的路网数据是指计算出当前行驶的最优路径时，所用的路网数据；

[0061] 所述的变化率是指相对变化率。

[0062] 步骤 (7)：判断变化率平均值 $\lambda$ 是否大于阈值 $\alpha$，若是则跳转执行步骤 (2)，并重新赋予 $v_c$ 为起点，$v_d$ 为终点，以上述步骤 (5) 得到的限制搜索区域 $Z$ 建立新路网；若否则执行步骤 (3)；

[0063] 所述的阈值 $\alpha$ 是经验值，本专利实施例取 20%。

[0064] 上述实施例并非是对于本发明的限制，本发明并非仅限于上述实施例，只要符合本发明要求，均属于本发明的保护范围。
图1 (a)

图1 (b)

图 1
图 2
图 3