**充电回路构建**

周期性充电

周期性充电：每次为分区内所有的电单车都按一定次序进行充电，假设分区中的电单车个数为*N*；

充电可调度性决策条件：

得到周期性充电方式下路径可调度性的充分必要条件：(1)；(2)

其中D为整个充电路径的路径长度；

T为整个充电路径的充电周期；

N为所有节点个数；

为MC的移动速率；

为MC充电速率；

η为充电效率；

为不同节点的耗电速率；

为MC的总容量；

为MC的移动消耗能量速率；

（以上数据可以参照胡诚论文，如果没有数据的话可以先设一个值，比如不同节点的耗电速率可以设置为（0,s)中的某个随机值）

|  |
| --- |
| Algorithm 确定充电回路R是否可调度 |
| 1：**输入**：N，，，，η，R={}，；  2：**输出**：充电回路R是否可调度  3：应用A\*寻路算法确定N个电单车之间的最佳路径；  4：计算路径长度D；  5：收集N个节点的最大，并计算；  6：**If**满足周期性充电方式路径可调度性条件**then**  7： **return**充电回路R可调度；  8:  **else**  9:  **return** 不可调度；  10: **End If** |

|  |
| --- |
| Algorithm N个电单车的充电子回路构建 |
| 1：**输入**：N个电单车，  2：**输出**：充电子回路及其个数值z；  3：z=1，i=1；  可以省略：4：应用A\*寻路算法确定N个电单车之间的最佳路径，构造无向图G(V,E)；  5：针对G(V,E)应用蚁群算法得到N个电单车的最短哈密顿回路(起始结束于)；  修改：5：按功率从小到大排序，构造回路H。  6：从开始顺时针标志最短哈密顿回路中的电单车分别为，；  7： **While**  do  8： **for** k=i to N **do**  8:  将加入中；  9: 应用可调度性判定算法判定的可调度性；  10:  **If** 是不可调度的 **then**  11: 将从回路中取出；  12: i=k；  13: z=z+1；  14:  **break；**  15: **End If**  16: **End for**  17: **End While** |

最初构建的最短哈密顿回路可以按照电单车的功耗由高到低（或由低到高）的顺序生成H，则最终得到的各个子回路的电单车功耗相近，因此不存在瓶颈电单车，从而MCV为每个回路充电的效率也更高。