

参赛密码 _____
(由组委会填写)

第十二届“中关村青联杯”全国研究生
数学建模竞赛

学 校 中国人民解放军海军工程大学

参赛队		90038025
队员姓名	1.	何宪文
	2.	李湘平
	3.	周 仁

参赛密码 _____
(由组委会填写)



第十二届“中关村青联杯”全国研究生 数学建模竞赛

题 目 面向节能的列车调度优化决策研究

摘 要：

随着低碳环保、节能减排日益受到关注，轨道交通系统中针对减少列车牵引能耗的列车运行优化控制近年来成为轨道交通领域的重要研究方向。本文针对面向节能的列车调度优化决策问题进行了深入研究。

对于问题1-1，以站间时间、距离、速度限制等为约束条件，建立了单列车单区间节能运行优化模型。按“牵引-巡航-惰行-制动”的模式寻找单列车单区间节能运行轨迹，以一定时间步长进行数值计算，通过始末两端速度、位移边界条件等，计算不同离散时间节点的速度、位移和加速度，搜索出能耗最小的一组运行参数。通过该组运行参数，绘制出该区间的速度位移曲线，得到最小消耗能量 $E_{\min}=36675.823\text{kJ}$ 。

对于问题1-2，同样，以耗能最小为优化目标，建立了单列车双区间节能运行优化模型。当固定一个区间的运行时间，则单列车双区间节能优化问题转化为单列车单区间节能优化问题。因此针对单列车双区间节能运行优化问题，实质上是增加了双区间在运行时间上的寻优。提出了变时搜索的优化策略，得到了单列车双区间的节能运行轨迹，其最小消耗能量 $E_{\min}=67753.4858\text{kJ}$ 。

对于问题2-1，首先给出了单列车多区间的最优节能运行方案，在所有列车在轨道线路上运行模式一致时，以可利用的再生能量最大为目标，以列车发车间隔为决策变量，建立多列车节能运行优化模型。采用联合迭代的算法为模型设定相同的单列车全区间运行轨迹，然后采用区间重合法，以5辆车为一组进行循环发车的思路对优化目标进行计算。得到前5辆车对应的发车间隔 $H'=\{h_1, h_2, h_3, h_4, h_5\}=\{645, 660, 645, 660, 615\}$ 。

对于问题2-2，考虑早晚高峰期，首先，将列车运行分成五个时间段，并依据不同时间段发车间隔的约束条件，分配不同时间段的发车数，然后，将整天的列车运行转化为五个多列车运行问题，采用优化的发车间隔，利用联合迭代算法，调节时间区间

内对应发车数量的集合，以再生能源最大化为目标，获得不同阶段的多列车运行策略。。

对于问题3-1，为了解决如何安全、尽快、耗能小的恢复正常运行问题，建立了考虑延误的列车运行控制模型。选择某特定环境，对特定延误进行两种不同的控制方案进行比较，从而实现优化控制。

对于问题3-2，根据统计数据，提出切实可行的优化调整策略。一方面是建立一个能够准确量化后车延误时间的模型，另一方面对具有随机性延误时间的列车运行控制策略进行了分析。

最后，论文还对给出了模型的优缺点及进一步改进的方向。

关键词： 列车节能运行，多约束条件，定值搜索优化算法，联合迭代算法，制动能量再生

目 录

1.问题重述.....	4
2.基本假设与符号说明.....	5
2.1 基本假设.....	5
2.2 符号说明.....	5
3.问题分析.....	6
4.单列车节能运行建模与求解.....	7
4.1 单列车两站间运行建模与求解.....	7
4.1.1 建模准备.....	7
4.1.2 模型建立.....	8
4.1.3 模型求解.....	9
4.2 单列车两站间运行结果及分析.....	10
4.3 三站间运行能耗建模与求解.....	13
4.3.1 建模准备.....	13
4.3.2 模型建立.....	13
4.3.3 模型求解.....	14
4.4 单列车三站间运行结果及分析.....	14
5.多列车节能运行建模与求解.....	18
5.1 多列车节能运行优化建模.....	18
5.2 多列车节能运行优化模型求解.....	19
5.2.1 单车运行轨迹优化.....	19
5.2.2 实时速度监测.....	21
5.2.3 多列车运行策略优化.....	22
5.3 考虑早晚高峰期多列车节能运行优化建模.....	24
5.4 考虑早晚高峰期多列车节能运行优化模型求解.....	25
6.考虑延误的列车节能运行控制建模与求解.....	28
6.1 基于延误的列车节能运行控制模型.....	28
6.2 延误时间为随机变量时的多列车节能调整策略.....	30
7.模型评价.....	30
参考文献:	32
附录.....	33

1.问题重述

随着低碳环保、节能减排日益受到关注，对能源合理使用、分配和回收成为各个行业关注的重点。铁路各部门中，运输部门的能耗占到铁路能耗的 80%以上。机车牵引能耗占到铁路能耗的 60%~70%，是铁路能耗的主要部分，降低机车牵引能耗对于降低铁路能耗具有举足轻重的作用。列车运行可分为牵引模式、巡航模式、惰行模式和制动模式，根据路况、列车性能、行驶时间可灵活选择各种运行模式进行组合。各种运行模式所对应的能源消耗不同，因此在满足安全、准时、舒适等相关约束条件的情况下，应尽可能减少列车的能量消耗，增加能量的回收利用。

本文旨在研究以下问题：

问题一：单列车数学建模并计算速度距离曲线

(1) 列车在行驶过程中主要受到牵引力、基本阻力、附加阻力、重力、制动力和支持力的共同作用。根据对列车动力学模型的分析，结合列车限制速度、轨道坡度和轨道曲率等约束条件，对单列列车在 110s 内由 A_6 站运行到 A_7 站最小能量消耗问题展开研究，建立适当的数学模型计算速度距离曲线，得到列车各阶段运行模式的详细报告及相应的最小能量消耗值。

(2) 在前阶段计算模型的基础上，分析单列列车在 220s 内从 A_6 站到 A_8 站的运行情况，其中 A_7 站停靠时间为 45s，建立新的模型分析列车从 A_6 站运行到 A_8 站能耗最省的速度距离曲线，并得到各阶段运行模式的详细报告。

问题二：多列车运行优化计算建模

(1) 分析共 100 列列车从 A_1 站依次运行到 A_{14} 站的多列车调度情况，各列列车的发车间隔时间为 $H=\{h_1, \dots, h_{99}\}$ ，各个站点的停靠时间为 $D_{\min} \sim D_{\max}$ ，第一列列车发车时间和最后一列列车发车时间间隔 63900s，每列列车从 A_1 站到 A_{14} 站的运行时间均为 2086s。前后相邻两列列车追踪运行时，跟踪列车速度不能超过限制速度 V_{limit} ，以防止跟踪列车缺乏制动停车时间而发生追尾事故。所有列车处于同一供电区段，某列列车制动回收的能量可用于轨道上正处于牵引或巡航耗能的列车，建立优化模型求解使所有列车运行总能耗最低的最佳间隔时间 H 。

(2) 分析 24h 内共 240 列列车依次从 A_1 站依次运行到 A_{14} 站的多列车调度情况，在出行高峰时间，其中早高峰为 7200 秒至 12600 秒，晚高峰为 43200 至 50400 秒，相邻列车间的发车间隔不大于 2.5min 且不小于 2min，其余非高峰时段发车间隔不小于 5min，制定列车的运行图和相应的速度距离曲线。

问题三：列车延误发车后优化控制

(1) 在前阶段的模型基础上，分析某列列车延误一定时间后，通过建立控制模型在确保安全的情况下，尽快使所有后续列车恢复正点运行，同时使能耗最小，并得到相应的列车运行曲线。

(2) 若延误时间的时长由已知固定值变为随机变量，并服从一定的概率分布，允许列车在各站到、发时间与原时间相比提前不超过 10 秒，根据问题二中的统计数据，如何调整控制方案使能耗最小。

2.基本假设与符号说明

2.1 基本假设

- 1.多列车运行时，后续列车在各个站点间均按照第一辆列车的最优运行轨迹运行；
- 2.列车驾驶人员能够按照制定的控制方法准确的操作列车，忽略列车驾驶员的误操作；
- 3.不考虑洪涝、雪灾、雾霾等极端天气情况以及轨道故障、紧急停车等列车运行过程中意外制动的情况；
- 4.假设所有列车处于同一供电区段,再生制动的能量可被同一供电区段的其它任一列车使用；
- 5.列车停靠站时其中心点正对地标处。

2.2 符号说明

符号	符号说明
E_q	牵引阶段列车能耗
E_x	巡航阶段列车能耗
E_z	列车制动时产生的能量
E_{used}	列车制动过程中被利用的再生能量
F	列车牵引力
v_q	牵引阶段结束时刻速度
v_x	巡航阶段结束时刻速度
v_z	制动阶段开始时刻速度
v_d	惰性阶段结束时刻速度
B	列车运行的制动力
μ_q	牵引加速度系数
μ_z	制动加速度系数
v_a	轨道最大限制速度
v_b	列车自身限制速度
t_q	牵引段运行时间
t_x	巡航段运行时间
t_d	惰行段运行时间
t_z	制动段运行时间
i	线路坡度
R	曲率半径
M	列车质量
D_{min}	单列列车在 A_i 站停站的时间
D_{max}	多列列车在各个车站最小停靠时间
H_{min}	多列列车在各个车站最大停靠时间
H_{max}	两列列车发车最小间隔时间
V_{limit}	轨道限制速度
V_{real}	列车实时速度

3.问题分析

对于问题1-1，以站间时间、距离、速度限制等为约束条件，建立了单列车单区间节能运行优化模型。按“牵引-巡航-惰行-制动”的模式寻找单列车单区间节能运行轨迹，以一定时间步长进行数值计算，通过始末两端速度、位移边界条件等，计算不同离散时间节点的速度、位移和加速度，搜索出能耗最小的一组运行参数。通过该组运行参数，绘制出该区间的速度位移曲线，得到最小消耗能量 $E_{\min}=36675.823\text{kJ}$ 。

对于问题1-2，同样，以耗能最小为优化目标，建立了单列车双区间节能运行优化模型。当固定一个区间的运行时间，则单列车双区间节能优化问题转化为单列车单区间节能优化问题。因此针对单列车双区间节能运行优化问题，实质上是增加了双区间在运行时间上的寻优。提出了变时搜索的优化策略，得到了单列车双区间的节能运行轨迹，其最小消耗能量 $E_{\min}=67753.4858\text{kJ}$ 。

对于问题2-1，首先给出了单列车多区间的最优节能运行方案，在所有列车在轨道线路上运行模式一致时，以可利用的再生能量最大为目标，以列车发车时间间隔为决策变量，建立多列车节能运行优化模型。采用联合迭代的算法为模型设定相同的单列车全区间运行轨迹，然后采用区间重合法，以5辆车为一组进行循环发车的思路对优化目标进行计算。得到前5辆车对应的发车间隔 $H'=\{h_1, h_2, h_3, h_4, h_5\}=\{645, 660, 645, 660, 615\}$ 。

对于问题2-2，考虑早晚高峰期，首先，将列车运行分成五个时间段，并依据不同时间段发车时间间隔的约束条件，分配不同时间段的发车数，然后，将整天的列车运行转化为五个多列车运行问题，采用优化的发车间隔，利用联合迭代算法，调节时间区间内对应发车数量的集合，以再生能源最大化为目标，获得不同阶段的多列车运行策略。。

对于问题3-1，为了解决如何安全、尽快、耗能小的恢复正常运行问题，建立了考虑延误的列车运行控制模型。选择某特定环境，对特定延误进行两种不同的控制方案进行比较，从而实现优化控制。

对于问题3-2，根据统计数据，提出切实可行的优化调整策略。一方面是建立一个能够准确量化后车延误时间的模型，另一方面对具有随机性延误时间的列车运行控制策略进行了分析。

4.单列车节能运行建模与求解

4.1 单列车两站间运行建模与求解

4.1.1 建模准备

1、列车运行模式

对列车受力情况进行分析，判断列车能耗与运行模式的关系，列车运行分为四种模式：牵引模式、巡航模式、惰行模式和制动模式。列车处于牵引模式时，列车消耗能量加速运行；处于巡航模式时，列车消耗能量匀速运行；处于惰行模式时，列车随路况在重力与阻力合成作用下运行，不消耗能量也不产生能量；处于制动模式时，列车减速运行，若为再生制动则可产生能量。

通过运动模式可初步判断列车在一定路况上的最节能运行模式，距离较短时列车一般采用“牵引-惰行-制动”的策略运行。如果站间距离较长，列车通常会采用牵引到接近限制速度后，交替使用惰行、巡航、牵引三种工况，直至接近下一车站采用制动进站停车。

2、列车受力分析

对列车受力情况进行简化，视为一个单质点模型，如图 4.1：

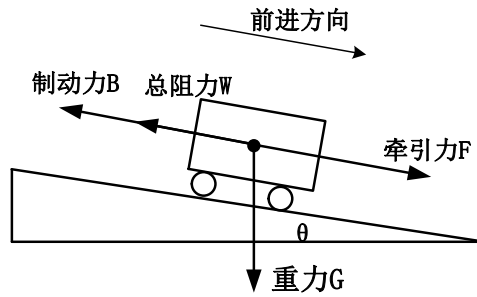


图 4.1 列车单质点模型受力分析

由已知条件，牵引力为

$$F(v) = \begin{cases} 203 \cdot \mu_q(t) & 0 \leq v \leq 51.5 \\ \mu_q(t) \cdot (-0.0020332v^3 + 0.4928v^2 - 42.13v + 1343) & 51.5 \leq v \leq 80 \end{cases} \quad (4.1)$$

阻力为

$$W(v) = (A + Bv_q + Cv_q^2 + i + C/R) \cdot g \cdot M / 1000 \quad (4.2)$$

制动力为

$$B(v) = \begin{cases} 166\mu_z(t) & 0 \leq v \leq 77 \\ \mu_z(t) \cdot (0.1343v^2 - 25.07v + 1300) & 77 \leq v \leq 80 \end{cases} \quad (4.3)$$

由图 4.1 可知，列车前进方向上受到牵引力 F 、总阻力 W 和制动力 B 的共同作用，列车的能量消耗只存在于牵引阶段和巡航阶段的牵引力做功。

3、对 A_6 站到 A_7 站进行路况分析

根据坡度、曲率半径和限速，对 A_6 站到 A_7 站的路况进行分析，见图 4.2。

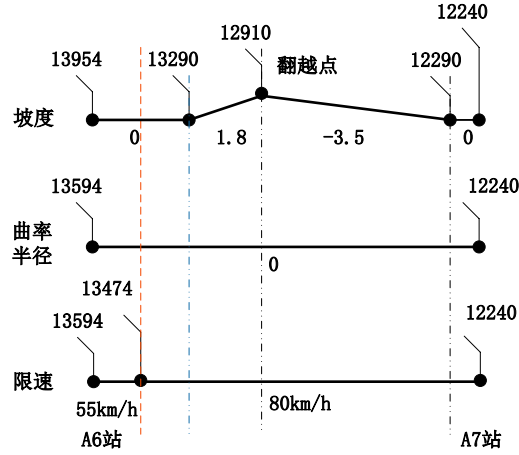


图 4.2 A_6 站与 A_7 站间路况分析

图 4.2 中，两站间不存在转弯路段，路段限速分为两段，速度限制分别为 55km/h 和 80km/h，在车站间距离较短时采用四阶段运行模式作为初始运行模式。

4.1.2 模型建立

1、目标函数

以最小能量消耗作为目标函数，由牵引阶段能耗和巡航阶段能耗组成。

设牵引阶段运行距离为 S_q ，牵引阶段能量消耗为

$$E_q = \int_0^{S_q} F(v) ds \quad (4.4)$$

设巡航阶段运行距离为 S_x ，匀速运行速度为 v_q ，巡航阶段能量消耗为

$$E_x = \int_0^{S_x} W(v_q) ds \quad (4.5)$$

则目标函数为

$$\text{Min } E = E_q + E_x = \int_0^{S_q} F(v) ds + \int_0^{S_x} W(v_q) ds \quad (4.6)$$

2、约束条件

1) 距离约束：四阶段运行总距离应等于两站间的距离：

$$S_1 = S_q + S_x + S_d + S_z = 1354 \quad (4.7)$$

2) 速度约束：牵引结束时刻速度 v_q 为列车整个运行过程中的最大速度，应小于道路最大限制速度 v_a 和自身限速 v_b ，且应大于列车全程匀速运动的速度值，即 $1354/110=12.3 \text{ m/s}$ ；制动阶段起始速度 v_z 小于巡航阶段速度 v_x ；初始速度与末速度均为零。可得速度约束为

$$\begin{cases} v_q \leq \min\{v_a, v_b\} \\ 44 \leq v_q \leq 80 \\ v_0 = v_t = 0 \\ 0 \leq v_z \leq v_x \end{cases} \quad (4.8)$$

3) 加速度约束：

$$\begin{cases} -1 \leq a_q \leq 1 \\ -1 \leq a_z \leq 1 \end{cases} \quad (4.9)$$

4) 时间约束:

$$t = t_q + t_x + t_d + t_z = 110 \quad (4.10)$$

综上所述, 单车从 A_6 站到 A_7 站间运行模型为

$$\begin{aligned} \text{Min } E &= \int_0^{S_q} F(v) ds + \int_0^{S_z} W(v_z) ds \\ &\left\{ \begin{aligned} S &= S_q + S_x + S_d + S_z = 1354 \\ t &= t_q + t_x + t_d + t_z = 110 \\ 0 &< \mu_q \leq 1 \\ 0 &< \mu_z \leq 1 \\ -1 &\leq a_q \leq 1 \\ -1 &\leq a_z \leq 1 \\ 44 &< v_q \leq 80 \\ 0 &\leq v_z \leq v_x \end{aligned} \right. \end{aligned} \quad (4.11)$$

4.1.3 模型求解

以一定时间步长进行数值计算, 通过始末两端速度、位移边界条件等可不断由两端向中间夹逼计算不同离散时间节点的速度、位移和加速度, 进而搜索使能耗最小的一组最优运行参数。模型求解的基本思路:

(1) 确定迭代参数

设定影响总能耗的关键迭代参数, 包括牵引加速度系数 μ_q 、制动加速度系数 μ_z 、牵引结束时刻速度 v_q 、制动开始时刻速度 v_z 。选定一组 μ_q 、 μ_z 、 v_q 、 v_z 进行初始迭代。

(2) 确定时间步长

根据运行总时间、最大限速可初步预估最大时间步长为 0~5s 之间, 其中 5s 由总路程/最大运行限速得到。考虑先采用 1s 作为时间步长进行初判, 再采用 0.1s 时间步长进行调优。

(3) 由牵引结束时刻速度 v_q 确定牵引段时间及位移参数

在时间步长、牵引加速度系数 μ_q 一定的情况下, 可得到牵引段的加速度 a_q , 通过数值积分迭代确定牵引段位移 S_q 。

$$\left\{ \begin{aligned} v_q &= \int_0^{t_1} a_q(t) dt \\ S_q &= \int_0^{t_1} v(t) dt = \int_0^{t_1} \left[\int_0^t a_q(\tau) d\tau \right] dt \end{aligned} \right. \quad (4.12)$$

(4) 由制动开始时刻速度 v_z 确定制动段时间 t_4 及制动位移 S_z

在时间步长、制动加速度系数 μ_z 一定的情况下, 可得到牵引段的加速度 a_z , 通过数值积分迭代, 确定牵引段时间 t_4 以及牵引段位移 S_z 。

$$\left\{ \begin{aligned} v_z &= \int_0^{t_4} a_z(t) dt \\ S_z &= \int_0^{t_4} v(t) dt = \int_0^{t_4} \left[\int_0^t a_z(\tau) d\tau \right] dt \end{aligned} \right. \quad (4.13)$$

(5) 由牵引段位移 S_q 以及制动段时间 S_z 确定惰行段时间 t_3 及位移 S_d

惰行段列车受到的合力包括基本阻力 f_b 以及附加阻力 f_a ， f_b 是关于速度的函数， f_a 是关于路标 s 的函数。由于不确定惰行段的起始位置，故不能获知其初始的加速度 $a_d(t)$ 。但在假定惰行段路程一定的情况下，必然可确定每一时刻的路标，可将 s 看作一系列常数，故计算时可通过将惰行阶段看作是由制动阶段开始点到巡航段结束点的加速过程的逆过程。

$$\begin{cases} v_d = \int_0^{t_3} a_d(t) dt \\ S_z = \int_0^{t_4} v(t) dt = \int_0^{t_3} [\int_0^t a_d(\tau) d\tau] dt \end{cases} \quad (4.14)$$

(6) 根据巡航段距离以及速度约束条件，确定巡航段运行时间 t_3 及位移参数 S_x 。由于巡航段做匀速运动，因此根据巡航段的运行速度 v_x 即可求得其运行时间 t_3 。

$$\begin{cases} S_x = S - S_q - S_z - S_x \\ t_3 = \frac{S_x}{v_x} \end{cases} \quad (4.15)$$

(7) 由运行得到的参数计算优化性能指标

在获得列车在各段运行情况的基础上，得到其能耗指标，其中巡航段列所受阻力与牵引力相等，因此可用阻力代替牵引力计算能耗

$$\begin{cases} E_q = \int_0^{t_1} F(v(t)) \cdot v(t) dt \\ E_x = \int_0^{S_x} W(v_q) ds \end{cases} \quad (4.16)$$

(8) 在约束条件下对 μ_q 、 μ_z 、 v_q 和 v_z 进行搜索寻优，比较每组参数下的能耗值，取最小值作为最优。

整个运行过程的实时速度、加速度、位置信息均已明确，形成了一个先确定约束段两端，再向中间逆向求解的夹逼计算方法，比较不同参数组合下的能耗值，最终得到在同等求解条件及约束条件下的较优能耗值及相应运行参数。

4.2 单列车两站间运行结果及分析

经 Matlab 编程计算，程序见附件 1-1，得到最小消耗能量 $E_{\min}=35967.1453\text{kJ}$ ， $v_q=60\text{km/h}$ ， $v_z=39\text{ km/h}$ ，列车运行时间为 110s，运行距离为 1340.9m，初始牵引加速度为 0.9 m/s^2 ，初始制动加速度为 1 m/s^2 。

速度距离曲线结果如图 4.3：

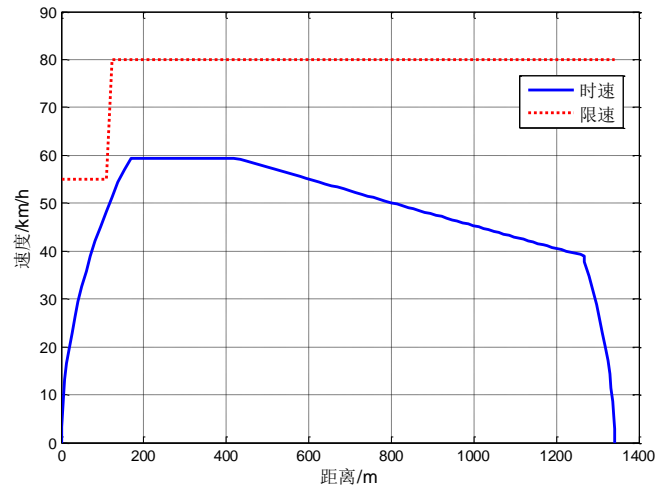


图 4.3 两站间速度距离曲线

速度时间曲线:

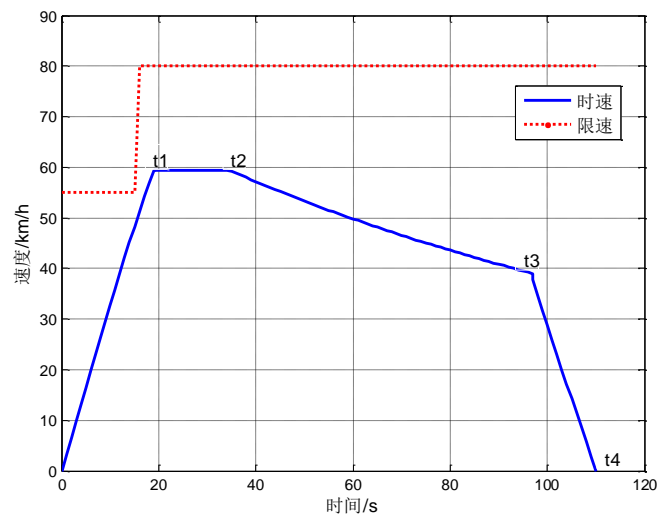


图 4.4 两站间速度时间曲线

由图 4.3 和图 4.4 可知，列车运行分为四个阶段， $0 \sim t_1$ 阶段为牵引阶段， $t_1 \sim t_2$ 阶段为巡航阶段， $t_2 \sim t_3$ 阶段为惰行阶段， $t_3 \sim t_4$ 阶段为制动阶段。从红色限速折线可以看出，列车在整个运行过程中始终满足速度限速约束，且最大时速为 60km/h。

加速度与时间的关系如图 4.5:

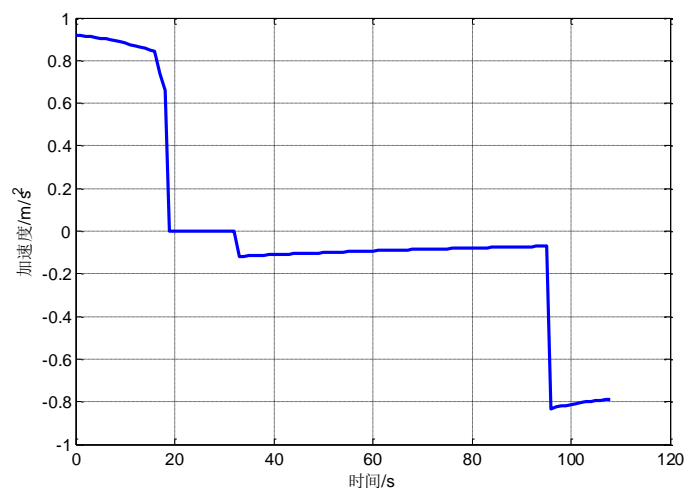


图 4.5 加速度曲线

从图 4.4 可知，整个运行过程中，牵引加速度和制动加速度幅值均在 $1m/s^2$ 以内，满足约束限制。牵引阶段列车速度逐渐增大，但加速度受最大牵引力的限制，逐渐减小；巡航阶段牵引力主要用于克服阻力，使列车匀速运动，加速度为零；惰行阶段停止牵引，列车逐渐减速，由于阻力与速度呈正比，因此阻力逐渐减小，则加速度幅值也随之减小；制动阶段，由于制动力的影响加速度负增长，直至运行速度为零。由于制动过程采用的是固定制动加速度系数的制动方式，因此，尽管到达终点时刻的速度为零，但此时的最大制动力不变，在下一时刻，能够通过置零加速度系数使加速度达到零。

能耗与时间、距离的关系 4.6、图 4.7：

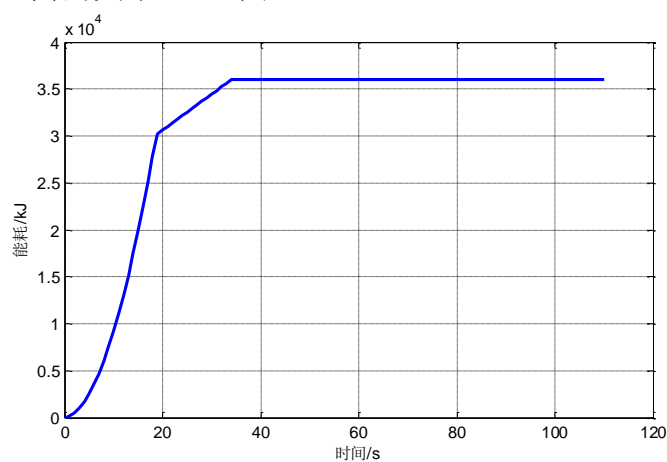


图 4.6 能耗随时间累积曲线

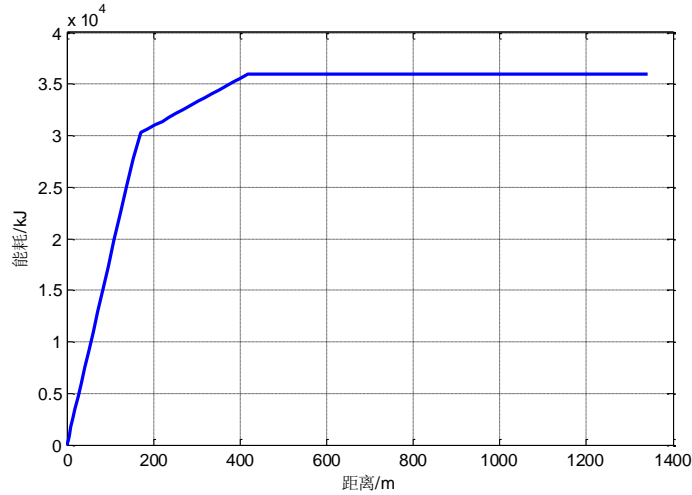


图 4.7 能耗随距离累积曲线

能耗产生过程只存在于牵引阶段和巡航阶段，故后两个阶段列车能耗不再增加。由结果图可知列车运行速度始终处于限速以内，时间、距离、加速度均满足约束限制。

4.3 三站间运行能耗建模与求解

4.3.1 建模准备

列车从 A_6 站出发经过 A_7 站后到达 A_8 站，中间停靠 45s，总运行时间为 220s。可将运行分为两个子运行过程： A_6 站到 A_7 站、 A_7 站到 A_8 站。分析第二个子运行过程的线路约束可知，其最大限速与第一子运行过程相同，且站间路程曲率均为 0，结合两个子运行过程各自的约束，分两个子问题分别循环搜索求解，最后综合搜索总运行时间为 220s 的能耗最优解。

4.3.2 模型建立

1、目标函数

目标函数为列车整个运行过程中的最小能耗，为两子运行过程能耗之和：

$$\text{Min } E_1 = \sum_{i=1}^2 (E_{qi} + E_{xi}) = \sum_{i=1}^2 \left[\int_0^{S_{qi}} F(v) ds + \int_0^{S_{xi}} W(v_{qi}) ds \right] \quad (4.17)$$

2、约束条件

1) 距离约束，各个子运行过程四阶段运行距离应等于两站间的距离：

$$\begin{cases} S_1 = S_{q1} + S_{x1} + S_{d1} + S_{z1} = 1354 \\ S_2 = S_{q2} + S_{x2} + S_{d2} + S_{z2} = 1280 \end{cases} \quad (4.18)$$

2) 速度约束，两站之间运行速度存在限速，且两个子运行过程速度约束相同

$$\begin{cases} 44 \leq v_{qi} \leq 80 \\ 0 \leq v_{zi} \leq v_{xi} \\ v_{qi} \leq \min \{v_a, v_b\} \\ i = 1, 2 \end{cases} \quad (4.19)$$

3) 加速度约束

$$\begin{cases} 0 \leq a_{qi} \leq 1 \\ 0 \leq a_{zi} \leq 1 \\ i = 1, 2 \end{cases} \quad (4.20)$$

4) 时间约束

$$t' = \sum_{i=1}^2 (t_{qi} + t_{xi} + t_{di} + t_{zi}) = 220 \quad (4.21)$$

综上所述，单车从 A_6 站到 A_8 站间运行模型为

$$\begin{aligned} \text{Min } E_1 = \sum_{i=1}^2 & \left[\int_0^{S_{qi}} F(v) ds + \int_0^{S_{xi}} W(v_{qi}) \right] ds \\ \left\{ \begin{array}{l} S_1 = S_1 + S_2 + S_3 + S_4 = 1354 \\ S_2 = S_{q2} + S_{x2} + S_{d2} + S_{z2} = 1280 \\ t' = \sum_{i=1}^2 (t_{1i} + t_{2i} + t_{3i} + t_{4i}) = 220 \\ 0 < \mu_{qi} \leq 1 \\ 0 < \mu_{zi} \leq 1 \\ -1 \leq a_q \leq 1 \\ -1 \leq a_z \leq 1 \\ 44 < v_{qi} \leq 80 \\ 0 \leq v_{zi} \leq v_{xi} \\ i = 1, 2 \end{array} \right. \end{aligned} \quad (4.22)$$

4.3.3 模型求解

模型求解过程与两站间运行问题类似，时间步长为两个子运行时间之和，将运行时间作为总约束循环选优求解总能耗。

4.4 单列车三站间运行结果及分析

最终迭代得到最小消耗能量 $E_{1\min} = 67336.0 \text{ kJ}$ ，运行总时间为220s，停车45s，运行总距离为2602.8m，Matlab程序见附件1-2。

第一子运行过程 A_6 站到 A_7 站牵引阶段结束速度 $v_{q1} = 55 \text{ km/h}$ ，制动开始速度 $v_{z1} = 35 \text{ km/h}$ ，运行时间为111s，实际运行距离为1338.9357m， A_6 站到 A_7 站规定运行距离1354m；

第二子运行过程 A_7 站到 A_8 站牵引阶段结束速度 $v_{q2} = 62 \text{ km/h}$ ，制动开始速度 $v_{z2} = 4 \text{ km/h}$ ，运行时间为109s，实际运行距离为1263.928m， A_7 站到 A_8 站规定运行距离1280m。

采用循环迭代搜索法，得到速度距离曲线如图4.8：

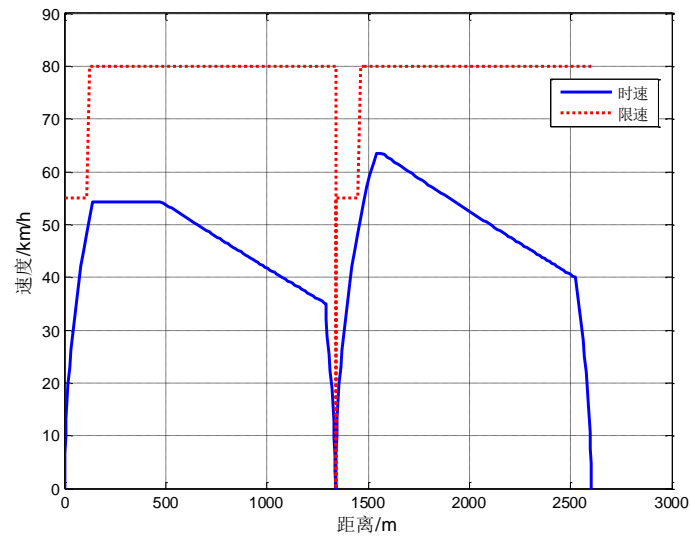


图4.8 三站运行速度距离曲线

同时可得

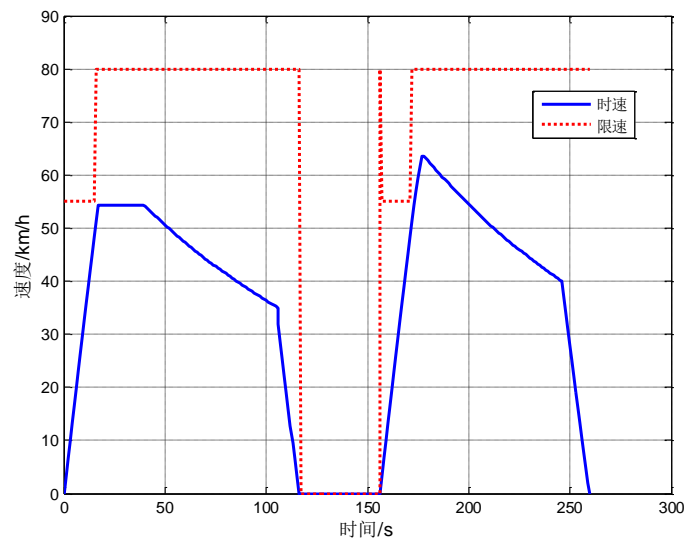


图4.9 三站运行速度时间曲线

由图4.8和图4.9可知，列车运行分为四个阶段，从红色限速折线可以看出，列车在整个运行过程中始终满足速度限速约束，且最大时速为60km/h。

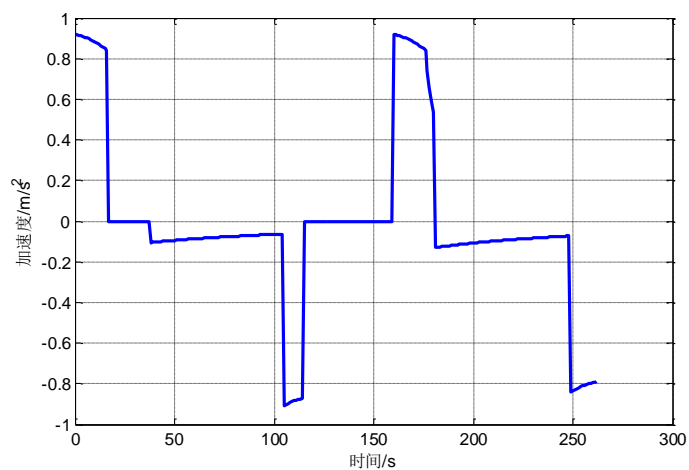


图4.10 三站运行加速度曲线

从图4.10可知，整个运行过程中，牵引加速度和制动加速度均在 $0 \sim 1 \text{ m/s}^2$ 内，满足约束限制。列车加速度是一个逐渐减小的过程，在牵引阶段逐渐减小；巡航阶段加速度为零；惰行阶段列车逐渐减速，由于阻力与速度呈正比，因此阻力逐渐减小，则加速度也随之减小；制动阶段，由于制动力的影响加速度负增长直至停车。

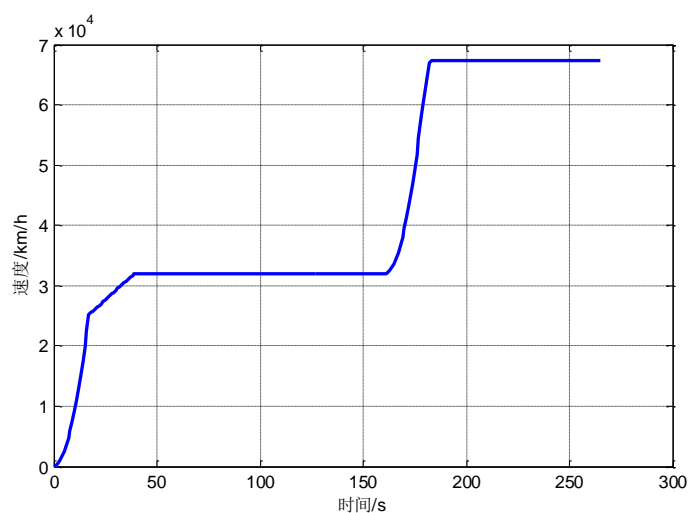


图4.11 三站运行能耗时间曲线

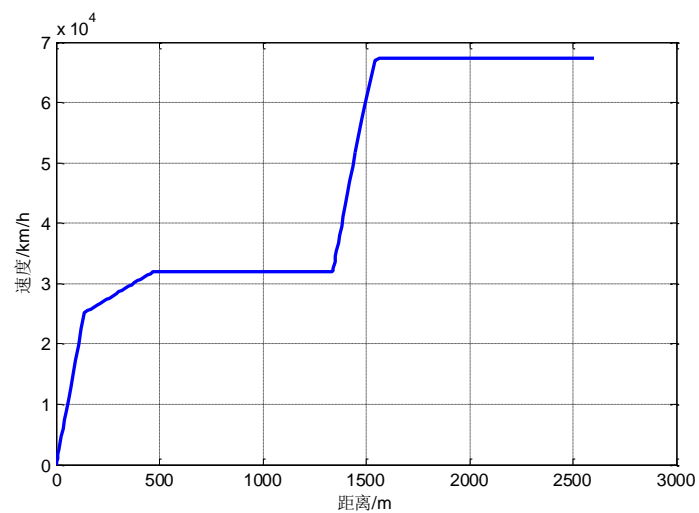


图4.12 三站运行能耗距离曲线

三站间能耗呈阶梯状，为两个子运行过程的能耗累加，能耗产生过程只存在于牵引阶段和巡航阶段，故后两个阶段列车能耗不再增加。由结果图可知列车运行速度始终处于限速以内，时间、距离、加速度均满足约束限制。

5.多列车节能运行建模与求解

5.1 多列车节能运行优化建模

根据已知条件分析：

- 1) D_i 表示在每一站停留的时间， $D_{\min} = 30s$ ， $D_{\max} = 45s$ ；
- 2) h_i 表示第*i*辆列车和第*i+1*辆列车的发车间隔， $H_{\min} = 120s$ ， $H_{\max} = 660s$ ；
- 3) T_0 表示第一列列车发车时间和最后一列列车的发车时间之间间隔；
- 4) T 表示从 A_1 站到 A_{14} 站的总运行时间（包括停站时间）；
- 5) S 表示从 A_1 站到 A_{14} 站的总距离。
- 6) V_{real} 表示实时速度， V_{limit} 表示限制速度， V_{line} 表示线路限速， L 表示后车与前车的距离， B_e 表示最大减速度。
- 7) $L=100$ ，其中 L 表示列车总数。

其中 L 表示列车数量， N 表示对于列车*i*的重合区间， $\Delta E_{used(i)}^j$ 表示第*i*辆列车第*j*段制动区间（与加速段重合部分）产生的可利用的再生能量。

1、目标函数

$$E_{used} = \max \sum_{i=1}^L \sum_{j=1}^N \Delta E_{used(i)}^j \quad (5.1)$$

2、约束条件

$$s.t. \begin{cases} D_{\min} < D_i < D_{\max} \\ H_{\min} < h_i < H_{\max} \\ T_0 = 63900s \\ T = 2086s \\ S = 22728m \\ V_{real} < V_{limit} = \min(V_{line}, \sqrt{2LB_e}) \\ L = 100 \end{cases}$$

多列车节能运行模型结构框图如图4.1所示，多列车节能运行模型包括实时速度检测模块，运行轨迹优化模块和再生能量计算模块三部分。其中实时速度监测模块获取保证安全情况下的最小时间间隔 T_{safe} ，运行轨迹优化模块获取单列车最节能运行轨迹，再生能量计算模块计算制动再生生成的能量，从而构建多列车节能运行模型。

对于多参数 $H=\{h_1, h_2 \dots h_{99}\}$ 的估计是一个复杂的非线性问题，用搜索寻优的方式对于计算量是一个巨大的挑战。根据第1个和第3个限制条件很容易计算得到，在 A_1 站点到 A_{14} 站点运行路程上最多可以存在5辆列车同时运行。因此，该模型对100辆列车进行分组，每5辆列车作为一组，对每一组中的5辆列车的发车时间 $H_1 = \{h_1, h_2, h_3, h_4\}$ 进行优化，计算5辆列车的再生能量，而 h_5 作为补足量以保证第一列列车发车时间和最后一列列车的发车时间之间间隔符合要求。剩余组列车的发车间隔按照第一组前5辆列车的发车时间进行循环，从而计算100辆列车运行过程中的再生能量。

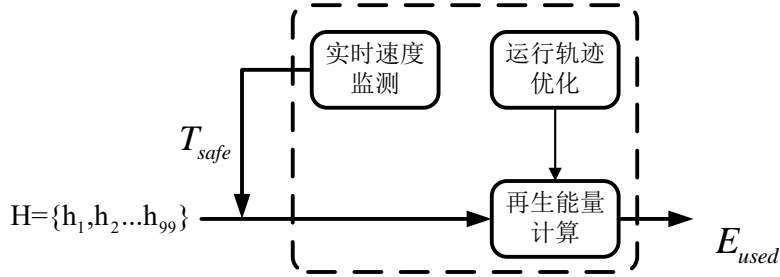


图5.1 多列车节能运行优化模型结构框图

5.2 多列车节能运行优化模型求解

5.2.1 单车运行轨迹优化

1、求解思路与步骤

运行轨迹优化模块用于获取模型中列车采用的最优节能运行轨迹，由于从 A_1 站点到 A_{14} 站点包含 13 段运行路线，如果用每段运行路线的运行时间作为搜索参数进行全局寻优，其计算复杂度大，计算时间长。本模块采用部分寻优和整体联合迭代的算法搜寻较优解。

本模块首先对 13 段运行路线进行分区段，并对区段进行标号，如图 4.2 所示。

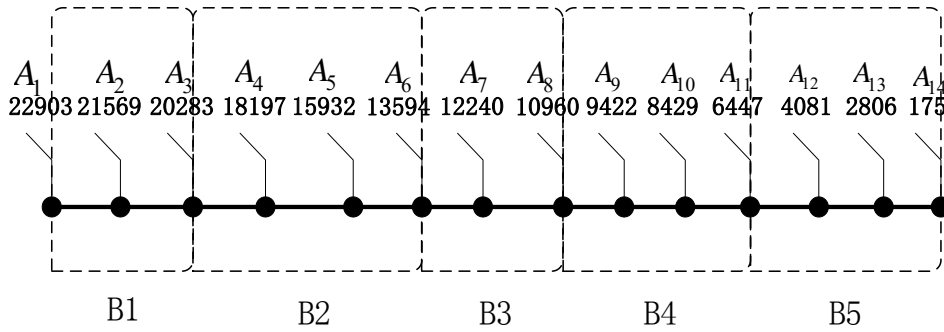


图 5.2 运行航线分区段图

其中 $B_1 = \{A_1, A_2, A_3\}$, $B_2 = \{A_3, A_4, A_5, A_6\}$, $B_3 = \{A_6, A_7, A_8\}$, $B_4 = \{A_8, A_9, A_{10}, A_{11}\}$, $B_5 = \{A_{11}, A_{12}, A_{13}, A_{14}\}$ 。

然后将三站间节能运行模型推广到四站间节能运行模型，分别对 B_1, B_2, B_3, B_4, B_5 按定时的约束条件进行求解运行轨迹。每个区段内运行时间 Tg_i 由下式可得：

$$Tg_i = \frac{T_{total} - T_{wait}}{S_{total}} \times S_{Bi} \quad (5.2)$$

其中 T_{total} 为 2086s, T_{wait} 为 A2 站点到 A13 站点停车的总时间 $T_{wait} = 12D_{min}$, S_{total} 为 A1 站点到 A14 站点的总路程, S_{Bi} 为对应区段的区段路程。

根据上节提出的站间节能运行模型可以求解出在粗略定时下，相对较优的各區段运行轨迹。该运行轨迹作为局部寻优的较优解。

最后，采用整体联合迭代算法，对较优解进行全局寻优。联合迭代算法的流程图如图 4.3 所示。

1) 确定最大迭代次数 N;

2) 计算区段 $\{B_1, B_2, B_3, B_4, B_5\}$ 时间增加 ΔT_α 后各区段的运行轨迹，从而计算出各区段的能耗减少量 ΔE_m ;

3) 计算区段{B1,B2,B3,B4,B5}时间减少 ΔT_a 后各区段的运行轨迹，从而计算出各区段的能耗增加量 ΔE_a ；

4) 比较区段{B1,B2,B3,B4,B5}的能耗增减量 ΔE_m 和 ΔE_a ，确定能耗减少量 ΔE_m 最大的区段Bi和能耗增加量 ΔE_a 最大的区段Bj。

5) 对Bi和Bj段进行区段运行时间修定。具体修定方法为：Bi区段运行时间 Tg_i 增加 ΔT_a ，Bj区段运行时间 Tg_j 减少 ΔT 。

6) 继续进行迭代，直到迭代次数达到最大迭代次数N。

7) 迭代结束后，最后依据每个区段内运行时间 Tg_i 计算运行轨迹。得到单列车最节能全程运行轨迹。

2、求解结果

总能耗随迭代次数变化如图5.3所示：

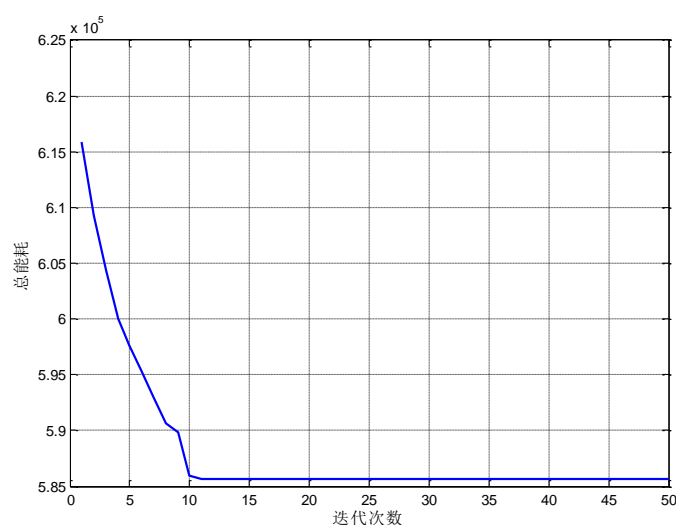


图5.3 总能耗随迭代次数变化图

由图可知：随着迭代次数的增加，运行轨迹的总能耗不断减少，说明迭代能够优化运行轨迹，在迭代次数为10次时，总能耗不再发生变化，说明运行运行轨迹已经得到优化。

单列车节能运行轨迹速度距离曲线如图5.4所示：

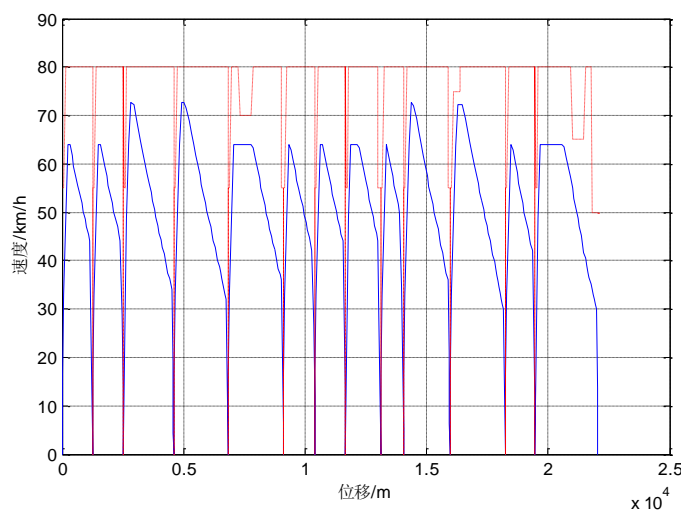


图5.4 全程位移速度曲线图

由图可知：单列车最节能运行轨迹为上图蓝色曲线，全路段限制速度与位移的关系曲线为上图红色曲线，能够满足定距离和限制速度的约束条件。该运行轨迹能够遵从限速条件保证运行安全，同时保证能耗最少，全程能耗量为 $6.39 \times 10^5 \text{ kJ}$ 。

单列车最节能运行轨迹速度时间曲线如图5.5所示。

由图可知：蓝色曲线表示单列车最节能轨迹时间速度曲线，能够做到在定时条件下完成从A1站点到A14站点的全程运行。其中每个站点停留时间设置为 D_{solid} 。

附件程序见附件 2-1。

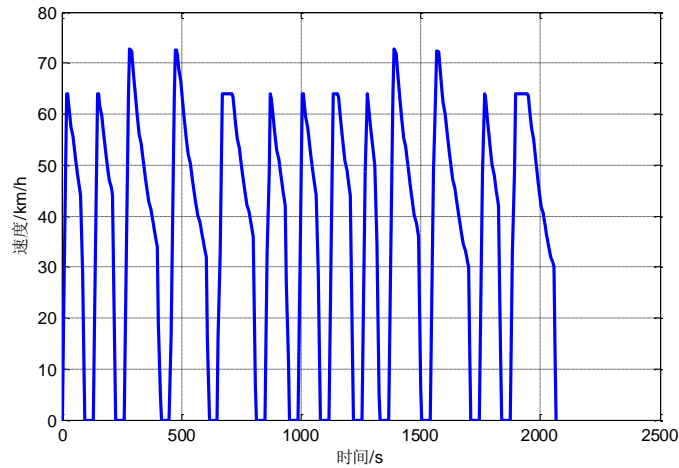


图5.5 全程时间速度曲线图

3. 误差分析

在限制条件中 $T = 2086s$ ，而实际运行时间为 $T_{real} = 2065s$ ，定时计算存在1%的时间误差。在限制条件中 $S = 22728m$ ，而实际运行距离为 $S_{real} = 22537m$ ，计算过程存在0.8%的距离误差。其中误差的主要来源是步长。在计算条件充裕的情况下，可以降低步长以满足精度计算的需求。

5.2.2 实时速度监测

1. 求解思路

多列车追踪运行时，前后列车之间的最小安全间隔时间 T_{safe} 是保证追踪列车运行安全的必要条件。追踪列车最小间隔时间 T_{safe} 是指在列车追踪运行时，能够保证安全的最小发车间隔时间。追踪列车以最小间隔时间 T_{safe} 发车，可以保证跟踪列车速度不能超过限制速度 V_{limit} ，以免后车无法及时制动停车，发生追尾事故。实时速度监测模块对于多列车追踪运行具有重要意义，对于控制系统运行方案的设计具有引导作用。其限制速度表示为：

$$V_{limit} = \min(V_{line}, \sqrt{2LB_e}) \quad (5.3)$$

其中， V_{line} 是列车当前位置的线路限速， L 是当前时刻前后车之间的距离， B_e 是列车制动的最大减速度。

实时监测模块在多列车节能运行优化模型的最前端，用于确定追踪列车的最小时间间隔 T_{safe} ，追踪列车的最小时间间隔是该优化模型的基础，保证多列车运行安全。实时监测的具体流程如下所述：

1) 设定起始时间间隔 T_0 ，步长为 ΔT_β 。

- 2) 前车按照单列车最节能全程运行轨迹开始运行。
- 3) 后车设定时间间隔 T_0 同样按照单列车最节能全程运行轨迹开始运行。
- 4) 按照离散化时间间隔点 ΔT ，在 $n\Delta T$ 时刻计算后车实时速度。
- 5) 与此同时计算前车与后车的距离间隔 $L = S_{T_0+n\Delta T} - S_{n\Delta T}$ 。
- 6) 判断是否 $V_{real} < V_{limit}$ ，如果否，则跳出循环，以时间间隔 $T_0 + \Delta T_\beta$ 重新开始运行该流程。
- 7) 如果全程的实时速度满足 $V_{real} < V_{limit}$ ，则认定该时间间隔为追踪列车的最小时间间隔 T_{safe} 。

2. 求解结果

根据实时速度监测模块的模型计算出 $T_{safe} = 220s$ ，给出某站点之间实时速度监测示意图，如图5.6所示。

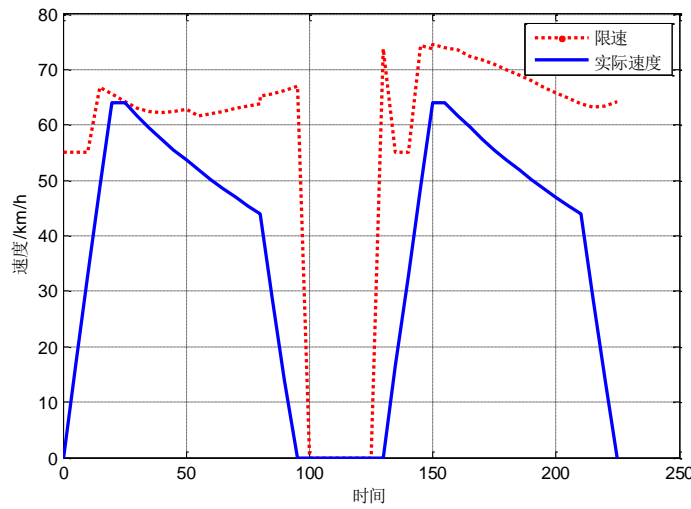


图5.4 某站点之间实时速度监测示意图

其中红色曲线表示限制速度，蓝色曲线表示实际速度，根据上图可知，实际运行的速度可以符合限制条件中限制速度的要求。要保证多列车安全运行，时间间隔限制条件设定为 $T_{safe} < h_i < H_{max}$ 。附件程序见Q2-1。

5.2.3 多列车运行策略优化

1. 求解思路

随着制动技术的进步，再生制动技术已经广泛应用于城市轨道交通。这一技术对于轨道交通的节能运行有着重大的意义。轨道交通列车在制动过程将动能转化为电能，从而减少以此同时从变电站获得的能量。即再生能量仅为所处供电区段内其他列车加速使用，如果该时刻无列车加速则意味着再生能量被浪费。

多列车节能运行过程中都采用单列车最节能运行轨迹，因此所有列车运行轨迹相同。多列车运行轨迹对比示例图如图 5.5 所示，其中红色线段表示加速段，蓝色段表示制动段， h_i 表示第 i 辆列车和第 $i+1$ 辆发车间隔。由图 5.5 可知，当在 t_i 和 τ_i 段时表示再生能量被利用。为了准确计算再生能量，本节提出了区间重合算法。

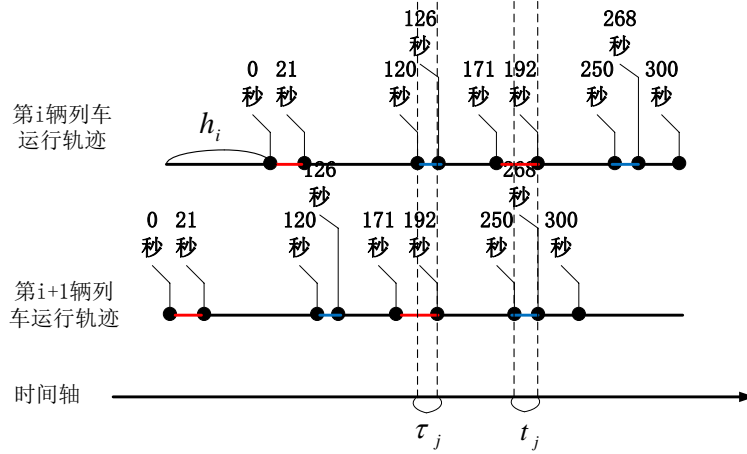


图 5.5 运行轨迹对比示例图

在计算再生能量之前需要对运行轨迹获得的数据信息作特定处理：单列车最节能全程运行轨迹过程中记录向量数据：所有向量长度为全路程运行轨迹的时间长度，向量 $\varphi_{accelerate}$ 用于在时间点上记录加速段的路径，加速段上的时间点用 1 表示，其余段用 0 表示；向量 φ_{brake} 与 $\varphi_{accelerate}$ 相似，用于在时间点上记录制动段的路径，制动段上的时间点用 1 表示，其余段用 0 表示；向量 φ_{vec} 记录各个时间点速度；向量 φ_{road} 记录各个时间点路况信息。

区间重合算法的核心思想即为通过比较第 i 辆列车的向量 φ_{brake} 和第 $i+1$ 辆列车推移 h_i 之后的向量 $\varphi_{accelerate}$ 的区间重叠部分。即两个向量都为 1 的时间长度即为 $t_{overlap}$ ；在第 $i+1$ 辆列车的向量 φ_{brake} 中求该重叠部分所在的制动过程的时间 t_{brake} 。根据向量 φ_{vec} 和向量 φ_{road} 可以计算 E_f, E_{mech} 。其中 $E_{mech} = E_M + E_G$ ，表示在制动过程中机械能转化为动能和势能。根据公式 $E_{used} = (E_{mech} - E_f) \times 95\% \times \frac{t_{overlap}}{t_{brake}}$ ，可以求出再生能量 τ_j ；同理可得，第 $i+1$ 辆列车处于制动段，第 i 辆列车处于加速段，产生的再生能量 t_j 。

但在多列车节能运行模型中，根据已知条件，会在 A1 站点到 A14 站点同时出现多辆列车的情况（不考虑早晚高峰的情况下，最多会出现 5 辆列车的情况）。同样可以计算多辆列车的区间重叠部分，用第 i 辆列车的向量 φ_{brake} 与其他辆列车推移发车间隔时间之后的 $\varphi_{accelerate}$ 进行比较，若有至少一辆列车的 $\varphi_{accelerate}$ 与第 i 辆列车的 φ_{brake} 重叠，则取最大重叠长度，其再生能量标记为可用，再生能量的计算的方法与上述方法类似，在此就不再赘述。之后再依次计算其他辆列车的再生能量，并且进行累加。

2. 求解结果

通过该模型可以根据不同组输入的 $H=\{h_1, h_2, \dots, h_{99}\}$ 计算出目标函数 E_{used} ，前 6 辆列车的发车时间为 $H'=\{h_1, h_2, h_3, h_4, h_5\}$ ，剩余组列车按照第一组前 5 辆列车的发车间隔进行循环发车，表 5.1 为不同发车间隔再生能量数据表，为了方便起见，只显示 $H'=\{h_1, h_2, h_3, h_4, h_5\}$ 。

表 5.1 发车间隔对应再生能量数据表

	h_1	h_2	h_3	h_4	h_5	E_{used}
数据 1	645	660	645	660	615	196115.4
数据 2	645	645	645	645	645	162644.1
数据 3	645	645	659	645	631	193101.9
数据 4	632	645	659	645	658	169629.9
数据 5	632	645	659	653	637	138647
数据 6	632	645	649	653	641	173417.1
数据 7	632	645	654	653	636	171839.4
数据 8	632	645	654	636	654	130042.8
数据 9	632	647	654	643	654	80883.59
数据 10	632	647	646	644	641	139126.3
数据 11	632	644	646	646	641	151406.2
数据 12	643	644	647	646	641	161208.5
数据 13	642	648	647	646	641	181379.4
数据 14	650	648	647	640	641	112172.7
数据 15	650	648	653	637	641	108744.8

根据上述表格数据可知：组 1 对应的 $H'=\{h_1, h_2, h_3, h_4, h_5\}$ ，其目标函数 E_{used} 为最大，使得全程总能耗最小，为最优的时间间隔 $H=\{h_1, h_2 \dots h_{99}\}$ ，其时间间隔的数据是基于前车制动时后车发车的特殊情况下。为了确认最优解，选择大量的数据进行寻优工作，经过反复寻优，确认该组数据可以作为较优解。

附件程序见附件 2-1

5.3 考虑早晚高峰期多列车节能运行优化建模

该模型建立在多列车节能运行优化模型的基础上，不同之处在于，该模型更为宏观地考虑在不同时间节点之间采用不同的时间间隔运行方案。不同时间节点示意图如图 5.6 所示。简单地把凌晨记作 M_1 ，该区段内发车总数为 L_1 ，早高峰记作 M_2 ，该区段内发车总数为 L_2 ，白天时间记作 M_3 ，该区段内发车总数为 L_3 ，晚高峰记作 M_4 ，该区段内发车总数为 L_4 ，夜晚记作 M_5 ，该区段内发车总数为 L_5 。 $L=\{L_1, L_2, L_3, L_4, L_5\}$ 的限定条件可以根据 $120s \leq h_{early}, h_{late} \leq 150s$ ， $300s \leq h_{other} \leq 660s$ 时间限定条件推导出，其中 h_{early}, h_{late} 分别代表早、晚高峰的发车间隔， h_{other} 代表其他时间段的发车间隔。

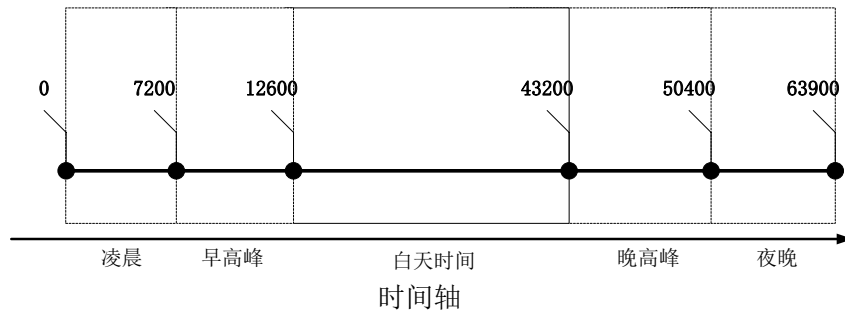


图 5.6 不同时间节点示意图

1、目标函数

$$E_{used} = \max \sum_{i=1}^5 \sum_{l=1}^{L_i} \sum_{j=1}^N \Delta E_{used(l)}^j \quad (i)$$

2、约束条件

$$s.t. \begin{cases} 10 \leq L_1 \leq 24 \\ 36 \leq L_2 \leq 45 \\ 46 \leq L_3 \leq 102 \\ 48 \leq L_4 \leq 60 \\ 20 \leq L_5 \leq 45 \\ L_1 + L_2 + L_3 + L_4 + L_5 = 240 \end{cases}$$

一旦确定某时间区段内出发的列车总数 L_i ，则可以把每一时间区段当做问题 2-1 中多列车节能运行优化模型来进行解析，可以求出该时间区间内再生能量值，通过对每一个时间区段进行求解，则可以求出目标函数 E_{used} 。因此，基于早晚高峰列车节能运行优化模型可以简化为对 L 的合理优化。即通过合理安排 $L = \{L_1, L_2, L_3, L_4, L_5\}$ ，来使目标函数 E_{used} 最大化，从而实现较节能的基于早晚高峰的多列车运行方案。

5.4 考虑早晚高峰期多列车节能运行优化模型求解

1、求解思路

模型建立的整体思路是采用联合迭代算法，对初始值进行全局寻优。

联合迭代算法的流程如下所述：

Step1: 确定最大迭代次数 N 并且对区段 $\{M_1, M_2, M_3, M_4, M_5\}$ 对应的

$L = \{L_1, L_2, L_3, L_4, L_5\}$ 进行初始化；

Step2: 根据 $L = \{L_1, L_2, L_3, L_4, L_5\}$ ，计算区段 $\{M_1, M_2, M_3, M_4, M_5\}$ ，从而计算出各区段的再生能量总和 $E'_{used} = \{E'_{used1}, E'_{used2}, E'_{used3}, E'_{used4}, E'_{used5}\}$ 。

Step3: $L = \{L_1, L_2, L_3, L_4, L_5\}$ 分别增加 ΔL ，计算区段 $M = \{M_1, M_2, M_3, M_4, M_5\}$ 对应的再生能量 $E_{used}^{inc} = \{E_{used1}^{inc}, E_{used2}^{inc}, E_{used3}^{inc}, E_{used4}^{inc}, E_{used5}^{inc}\}$ ，计算 $E_{used}^{inc} - E'_{used}$ ，选择其中最大的再生能量对应的区段 M_i ；

Step4: $L = \{L_1, L_2, L_3, L_4, L_5\}$ 分别减少 ΔL ，计算区段 $M = \{M_1, M_2, M_3, M_4, M_5\}$ 对应的再生能量 $E_{used}^{dec} = \{E_{used1}^{dec}, E_{used2}^{dec}, E_{used3}^{dec}, E_{used4}^{dec}, E_{used5}^{dec}\}$ ，计算 $E'_{used} - E_{used}^{dec}$ 选择其中最大的再生能量对应的区段 M_j ；

Step4: 对 M_i 和 M_j 区段进行 $L = \{L_1, L_2, L_3, L_4, L_5\}$ 修定。具体修定方法为： M_i 区段列车数量增加 ΔL ， M_j 区段列车减少增加 ΔL 。

Step5: 返回 Step2 继续进行迭代，直到迭代次数达到最大迭代次数 N 。

联合迭代过程中如果出现 $L = \{L_1, L_2, L_3, L_4, L_5\}$ 超出限定值，则跳出迭代。迭代结束后， $L = \{L_1, L_2, L_3, L_4, L_5\}$ 为最佳的节能运行方案。

2、求解结果与分析

通过该模型可以根据较优 $L = \{L_1, L_2, L_3, L_4, L_5\}$ 计算出目标函数 E_{used} ，根据较优 $L = \{L_1, L_2, L_3, L_4, L_5\}$ ，可以根据上文提到的多列车节能运行优化模型计算各时间区间的发车间隔。表 5.2 为不同时间区段列车数再生能量对应数据表，为了更清楚地表现，表格中显示其他 L 对应的数据作对照。表 5.3 为对应时间区段发车间隔循环表，以 A1 站

点到 A14 站点全路程内最多列车数作为周期，对发车间隔进行循环，循环周期为 $T_{\max} = \{8, 15, 5, 15, 5\}$ 。

表 5.2 不同时间区段列车数量对应再生能量数据表

	L_1	L_2	L_3	L_4	L_5	L	E_{used}
数据 1	21	37	90	54	38	240	16966306.16
数据 2	21	41	86	54	38	240	16053531.6
数据 3	21	37	90	54	38	240	15856942.25
数据 4	11	44	86	59	40	240	15416473.56
数据 5	24	37	90	51	38	240	15142276.79
数据 6	12	44	86	59	39	240	14217450.54
数据 7	19	42	90	51	38	240	12731517.5
数据 8	19	42	86	51	42	240	12731517.5
数据 9	22	42	86	51	39	240	12731517.5
数据 10	17	42	86	56	39	240	12731517.5
数据 11	19	42	90	51	38	240	12413649.93
数据 12	16	44	86	59	35	240	12392809.12
数据 13	14	42	86	59	39	240	12381090.03
数据 14	14	44	84	59	39	240	11716770.39

表 5.3 对应时间区段发车间隔循环表格

	M_1	M_2	M_3	M_4	M_5
h_1	337	131	345	135	359
h_2	340	146	340	144	341
h_3	342	131	352	128	352
h_4	346	146	361	119	362
h_5	342	121	377	133	362
h_6	348	125	..	133	..
h_7	342	136	..	148	..
h_8	339	131	..	118	..
h_9	..	128	..	119	..
h_{10}	..	119	..	117	..
h_{11}	..	140	..	133	..
h_{12}	..	141	..	151	..
h_{13}	..	129	..	148	..
h_{14}	..	131	..	134	..
h_{15}	..	120	..	135	..

根据上述表格数据可知：组 1 对应的最优 $L = \{L_1, L_2, L_3, L_4, L_5\}$ ，其目标函数 E_{used} 为 $1.69 \times 10^7 kJ$ ，再生能量值最大，从而使得总能耗最小， $L = \{21, 37, 90, 54, 38\}$ 确认为基于早晚高峰多列车较佳节能运行方案。表中选择大量的数据进行对比，以确认该组数据可以作为较优解。其中..表示继续循环。

附件程序见附件 2-2。

6.考虑延误的列车节能运行控制建模与求解

6.1 基于延误的列车节能运行控制模型

1、建模准备

当 DT_i^j 在运行过程中出现的时刻 (DT_i^j 表示第 i 辆列车, 在第 j 站出现延误, 其中 $DT_i^j = 10s$), 可能会引起后续的列车同时延迟的连锁反应。为了解决延误带来的一系列问题, 对延误的列车提出了安全, 快速, 耗能最小的恢复正常运行要求。为了满足上述要求, 需要减少延误列车到达下一站或者下两站的运行时间。

如果利用要求最快速的恢复正常运行, 则要求到达下一站的运行时间为 $T_{j,j+1} - DT_i^j$ ($T_{j,j+1}$ 为从 j 站点到 $j+1$ 站点正常运行的时间); 如果要求尽快且耗能最小, 则要求到达下一站的运行时间为 $T_{j,j+1} - \Delta A_j$, 到达下两站的运行时间为 $T_{j+1,j+2} - \Delta A_{j+1}$, 其中 $DT_i^j = \Delta A_j + \Delta A_{j+1}$; 如果速度不受限, 仅要求耗能最小, 则接下来站点与站点之间的运行时间调整为, $\{T_{j,j+1}, T_{j+1,j+2} \dots T_{13,14}\} - \{\Delta A_j, \Delta A_{j+1} \dots \Delta A_{13}\}$, 其中 $DT_i^j = \Delta A_j + \Delta A_{j+1} + \dots + \Delta A_{13}$ 。

基于延误的列车节能运行控制模型的一般情况下的目标函数:

$$E_{add} = \min \sum_{i=1}^L \sum_{k=1}^{Ki} \Delta E_i^{j,k} \quad (6.1)$$

其中, $\Delta E_i^{j,k}$ 表示在第 i 辆列车, 在第 j 站出现延误, 并且通过增加运行时间补偿延误时间的方法, 导致能耗的增加量, $\Delta E_i^{j,k} = f(\Delta A_k)$, 函数 $f(x)$ 为第 2 节中计算能耗的模型, 其中 L 为引起延误的列车数量, 列车 i 在 K 个站点之后恢复正常运行。

2、建模与求解

模型 1:

设定一种场景进行建模求解, 假定第 i 辆列车在 A3 站点发生了延误, 则在此特定情况下的目标函数为:

$$E_{add} = \min (\Delta E_i^{3,1} + \Delta E_i^{3,2}) \quad (6.2)$$

$$s.t. \begin{cases} DT_{i+1}^2 = 10s \\ K = 2 \\ L = 1 \end{cases}$$

采用搜索寻优的方法, 以 A3 站点到 A4 站点与 A4 站点到 A5 站点运行时间的减少量 $\{\Delta A_3, \Delta A_4\}$ 作为搜索量, $E_{add} = \min (\Delta E_i^{3,1} + \Delta E_i^{3,2})$ 为目标函数, 则最优搜索量 $\{\Delta A_3, \Delta A_4\} = \{7, 3\}$, 其 $E_{add} = 7 \times 10^3 kJ$ 。恢复延误路线与正常运行路线对比图如图 6.1 所示。

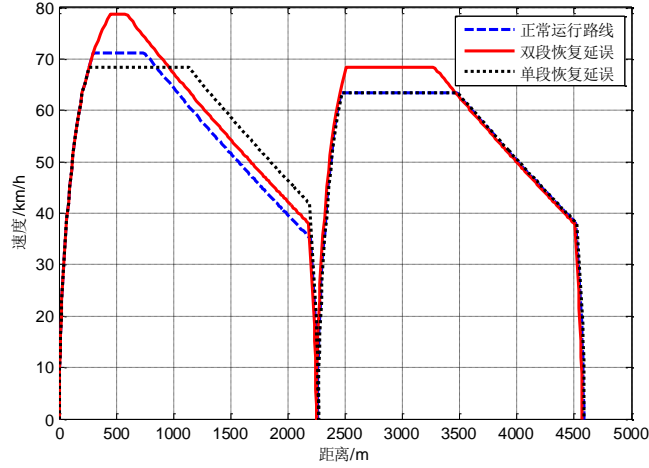


图 6.1 恢复延误路线与正常运行路线对比图

图中用 $K=1$ 时恢复路线作对比，其目标函数 $E_{add}=1.5 \times 10^4 kJ$ ，可以看出， $K=1$ 时恢复策略虽然速度最优，但是耗能较大，因此针对尽快及耗能较小的要求，采用 $K=2$ 时恢复策略最优。

模型 2:

同样设定第 i 辆列车在 A3 站点发生了延误，设定其后仅有第 $i+1$ 辆列车受延误影响， $DT_{i+1}^2=3s$ ，则在此特定情况下的目标函数为：

$$E_{add} = \min(\Delta E_i^{3,1} + \Delta E_i^{3,2} + \Delta E_{i+1}^{2,1}) \quad (6.3)$$

$$s.t. \begin{cases} DT_{i+1}^2 = 3s \\ DT_i^3 = 10s \\ K_i = 2 \\ K_{i+1} = 1 \\ L = 2 \end{cases}$$

模型 2 与模型 1 相比，多了后车受延误的限定条件，在模型 1 的基础上，以 $K_{i+1}=1$ 的恢复策略恢复正常运行，其目标函数 $E_{add}=1.68 \times 10^4 kJ$ ，恢复延误路线与正常运行路线对比图如图 6.2 所示：

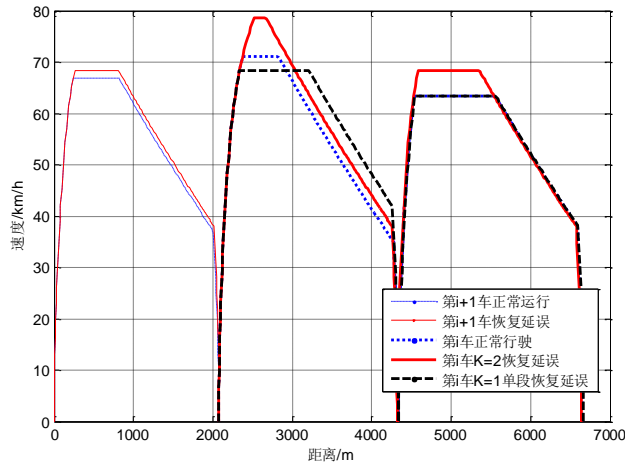


图 6.2 恢复延误路线与正常运行路线对比图

蓝色虚线为后车正常运行路线，红色虚线为后车恢复延迟路线。受延迟影响的后车能耗增多 $\Delta E_{i+1}^{2,1} = 1.8 \times 10^3 \text{ kJ}$ 。附件程序见附件 3-1。

6.2 延误时间为随机变量时的多列车节能调整策略

6.2.1 延误时间对后续车辆的影响分析

设第 $i+1$ 辆车与第 i 辆车的原始发车时间间隔为 h_i ，第 i 辆车在车站 A_j 延误 DT_j^i 后，两车时间间隔减少至 $h_i' = h_i - DT_j^i$ ，若在此间隔 h_i' ，两车在后面的车站运行过程中的最短距离为 Δd_i ，两车安全距离为 ΔD_i 。则当 $\Delta d_i \geq \Delta D_i$ 时，第 i 辆车延误 DT_j^i 对后续的第 $i+1$ 辆车无影响，若 $\Delta d_i < \Delta D_i$ ，则后续的第 $i+1$ 辆车必然要延误，若两车距离最近时的第 $i+1$ 辆车速度为 v_{i+1} ，则第 $i+1$ 辆车应该在车站 A_{j-1} 延时 $DT_{j-1}^{i+1} = \frac{\Delta D_i - \Delta d_i}{v_{i+1}}$ 。

由 $\Delta d_i = \Delta D_i$ ，可确定影响第 $i+1$ 辆车是否延时的第 i 辆车延时时间临界点 \tilde{DT}_j^i ，若第 $i+1$ 辆车延时时间 $DT_{j-1}^{i+1} > \tilde{DT}_j^i$ ，则第 $i+1$ 辆车与需要延时，否则不延时...由此可得出第 i 辆车在车站 A_j 延时 \tilde{DT}_j^i 对后续有影响的车辆数。

6.2.2 调整策略

当 DT_j^i 为随机变量时，由于允许列车在各站的发车时间与原时间对比可提前10秒，因此，当普通延时（ $0 < DT_j^i < 10\text{s}$ ）发生时，只需让该车在下一路段直接赶上即可；当发生严重延误时（概率为10%），可将后续车辆的延时时间均在使用上述方法的基础上减去10秒，即 $\overline{DT}_{j-1}^{i+1} = DT_{j-1}^{i+1} - 10$, $\overline{DT}_{j-2}^{i+2} = DT_{j-2}^{i+2} - 10, \dots$ ，再据此调整控制方案。

7.模型评价

本文针对列车调度问题进行了优化，采用强约束条件（时间、速度、加速度、能量以及距离等限制）下的循环搜索算法，首先通过简化列车运行需要优化的参数，在已有参数的基础上，从物理意义上精确推导其它的运行参数，极大提高了运行速度及精度。其次，基于物理情景及目标耦合迭代寻优思想，先确定单列车运行模式，减少列车之间的耦合参数，简化对于多列车的耦合优化问题。最后，对整个求解的过程进行模块化，从概念上以及代码的实现上，提出了模块化的框架，方便模型的求解及扩充。但本文模型未考虑列车变运动模式下的运行情况，优化算法存在一定的片面性。

对于单列车的节能运行建模，采用搜索寻优思想，构建了列车节能优化模型，同时考虑了多站之间的耦合限制因素；对于多列车的节能运行建模，建立了不同发车间隔、不同运行时间下的多列车耦合运动模式，同时给出了同等条件下的相对最优解。但是当列车运行情况发生变化时，尤其是发生突发事件时，未对模型进行全面考虑。受运算速度的限制，仿真步长较大，计算误差较大，可考虑采用变步长的算法进行数值计算。同时，在寻优算法的选择上，可以选择更为智能、自适应性更强的算法，如遗传算法、神

经网络等。

列车的调度优化问题是解决铁路能耗的关键问题，本文的研究能够对列车的实际调度具有一定的指导作用，但是需考虑更为全面的运行模型，以更好地适用真实轨道交通的运行状况。

参考文献：

- [1] P. Howlett, An Optimal Strategy for the Control of A Train, Journal of the Australian Mathematical Society. Series B. Applied Mathematics, Volume 31, Issue 04, April 1990, pp. 454-471.
- [2] 丁勇, 毛保华, 刘海东, 张鑫, 王铁城, 列车节能运行模拟系统的研究, 北京交通大学学报, 第 28 卷第 2 期, 第 76-81 页
- [3] 金炜东, 靳蕃, 李崇维, 胡飞, 苟先太, 列车优化操纵速度模式曲线生成的智能计算研究, 铁道学报, 第 20 卷, 第 5 期, 第 47-52 页.
- [4] 王峰, 刘海东, 丁勇, 陈善亮, 毛保华, 列车节能运行的算法及实施技术研究, 北方交通大学学报, 第 26 卷, 第 5 期, 第 13-18 页.

附录

A6-A7 运行数据

时 刻 (hh:mm:ss)	实 际 速 度(cm/s)	实 际 速 度(km/h)	计 算 加 速 度 (m/s ²)	计 算 距 离(m)	计 算 公 里标(m)	当 前 坡 度(%)	计 算 牵 引 力(N)	计 算 牵 引 功 率 (Kw)
0:00:00	0	0	0	0	0	0	182700	0
0:00:01	92.04189 052	3.313508 059	0.920418 905	0.920418 905	0.920418 905	0	182700	168.1605 34
0:00:02	183.8623 6	6.619044 959	0.918204 695	2.759042 505	2.759042 505	0	182700	335.9165 317
0:00:03	275.4231 955	9.915235 038	0.915608 355	5.513274 46	5.513274 46	0	182700	503.1981 782
0:00:04	366.6865 964	13.20071 747	0.912634 009	9.180140 423	9.180140 423	0	182700	669.9364 115
0:00:05	457.6152 358	16.47414 849	0.909286 394	13.75629 278	13.75629 278	0	182700	836.0630 357
0:00:06	548.1723 21	19.73420 356	0.905570 852	19.23801 599	19.23801 599	0	182700	1001.510 83
0:00:07	638.3216 518	22.97957 946	0.901493 308	25.62123 251	25.62123 251	0	182700	1166.213 658
0:00:08	728.0276 768	26.20899 637	0.897060 25	32.90150 928	32.90150 928	0	182700	1330.106 566
0:00:09	817.2555 477	29.42119 972	0.892278 709	41.07406 475	41.07406 475	0	182700	1493.125 886
0:00:10	905.9711 708	32.61496 215	0.887156 231	50.13377 646	50.13377 646	0	182700	1655.209 329
0:00:11	994.1412 565	35.78908 524	0.881700 858	60.07518 903	60.07518 903	0	182700	1816.296 076
0:00:12	1081.733 366	38.94240 119	0.875921 097	70.89252 269	70.89252 269	0	182700	1976.326 86
0:00:13	1168.715 956	42.07377 441	0.869825 895	82.57968 225	82.57968 225	0	182700	2135.244 051
0:00:14	1255.058 417	45.18210 3	0.863424 607	95.13026 641	95.13026 641	0	182700	2292.991 727
0:00:15	1340.731 113	48.26632 008	0.856726 969	108.5375 775	108.5375 775	0	182700	2449.515 744
0:00:16	1425.705 42	51.32539 512	0.849743 066	122.7946 317	122.7946 317	0	182700	2604.763 802
0:00:17	1509.953 75	54.35833 5	0.842483 3	137.8941 692	137.8941 692	0	164379.5 303	2482.054 882
0:00:18	1584.020 383	57.02473 38	0.740666 332	153.7343 731	153.7343 731	0	149618.4 549	2369.986 824
0:00:19	1649.801 304	59.39284 693	0.657809 203	170.2323 861	170.2323 861	0	138072.9 415	2277.929 189
0:00:20	1649.801 304	59.39284 693	0	186.7303 992	186.7303 992	0	23038.44 648	380.0885 904
0:00:21	1649.801 304	59.39284 693	0	203.2284 122	203.2284 122	0	23038.44 648	380.0885 904
0:00:22	1649.801 304	59.39284 693	0	219.7264 252	219.7264 252	0	23038.44 648	380.0885 904
0:00:23	1649.801 304	59.39284 693	0	236.2244 383	236.2244 383	0	23038.44 648	380.0885 904
0:00:24	1649.801 304	59.39284 693	0	252.7224 513	252.7224 513	0	23038.44 648	380.0885 904
0:00:25	1649.801	59.39284	0	269.2204	269.2204	0	23038.44	380.0885

	304	693		643	643		648	904
0:00:26	1649.801 304	59.39284 693	0	285.7184 774	285.7184 774	0	23038.44 648	380.0885 904
0:00:27	1649.801 304	59.39284 693	0	302.2164 904	302.2164 904	0	23038.44 648	380.0885 904
0:00:28	1649.801 304	59.39284 693	0	318.7145 034	318.7145 034	0	23038.44 648	380.0885 904
0:00:29	1649.801 304	59.39284 693	0	335.2125 165	335.2125 165	1.8	23041.87 385	380.1451 351
0:00:30	1649.801 304	59.39284 693	0	351.7105 295	351.7105 295	1.8	23041.87 385	380.1451 351
0:00:31	1649.801 304	59.39284 693	0	368.2085 426	368.2085 426	1.8	23041.87 385	380.1451 351
0:00:32	1649.801 304	59.39284 693	0	384.7065 556	384.7065 556	1.8	23041.87 385	380.1451 351
0:00:33	1649.801 304	59.39284 693	0	401.2045 686	401.2045 686	1.8	23041.87 385	380.1451 351
0:00:34	1649.801 304	59.39284 693	0	417.7025 817	417.7025 817	1.8	23041.87 385	380.1451 351
0:00:34	1649.801 304	59.39284 693	0	417.7025 817	417.7025 817	1.8	0	0
0:00:35	1646.072 292	59.25860 252	-0.03729 0115	434.2005 947	434.2005 947	1.8	0	0
0:00:36	1634.363 309	58.83707 911	-0.11708 9835	450.6613 176	450.6613 176	1.8	0	0
0:00:37	1622.766 463	58.41959 265	-0.11596 8461	467.0049 507	467.0049 507	1.8	0	0
0:00:38	1611.280 077	58.00608 278	-0.11486 3854	483.2326 153	483.2326 153	1.8	0	0
0:00:39	1599.902 509	57.59649 032	-0.11377 5682	499.3454 161	499.3454 161	1.8	0	0
0:00:40	1588.632 147	57.19075 729	-0.11270 3621	515.3444 412	515.3444 412	1.8	0	0
0:00:41	1577.467 411	56.78882 68	-0.11164 7356	531.2307 627	531.2307 627	1.8	0	0
0:00:42	1566.406 753	56.39064 312	-0.11060 658	547.0054 368	547.0054 368	1.8	0	0
0:00:43	1555.448 654	55.99615 155	-0.10958 0991	562.6695 043	562.6695 043	1.8	0	0
0:00:44	1544.591 624	55.60529 848	-0.10857 0298	578.2239 908	578.2239 908	1.8	0	0
0:00:45	1533.834 203	55.21803 131	-0.10757 4213	593.6699 071	593.6699 071	1.8	0	0
0:00:46	1523.174 957	54.83429 846	-0.10659 2459	609.0082 491	609.0082 491	1.8	0	0
0:00:47	1512.612 481	54.45404 932	-0.10562 4761	624.2399 987	624.2399 987	1.8	0	0
0:00:48	1502.145 396	54.07723 424	-0.10467 0855	639.3661 235	639.3661 235	1.8	0	0
0:00:49	1491.772 348	53.70380 451	-0.10373 048	654.3875 775	654.3875 775	1.8	0	0
0:00:50	1481.492 009	53.33371 234	-0.10280 3382	669.3053 009	669.3053 009	1.8	0	0
0:00:51	1471.308 226	52.96709 613	-0.10183 7835	684.1202 21	684.1202 21	-3.5	0	0
0:00:52	1461.214	52.60372	-0.10093	698.8333	698.8333	-3.5	0	0

	525	29	7007	033	033			
0:00:53	1451.209 653	52.24354 751	-0.10004 8722	713.4454 485	713.4454 485	-3.5	0	0
0:00:54	1441.292 378	51.88652 561	-0.09917 2748	727.9575 451	727.9575 451	-3.5	0	0
0:00:55	1431.461 492	51.53261 372	-0.09830 886	742.3704 688	742.3704 688	-3.5	0	0
0:00:56	1421.715 808	51.18176 91	-0.09745 6838	756.6850 838	756.6850 838	-3.5	0	0
0:00:57	1412.054 162	50.83394 982	-0.09661 6467	770.9022 419	770.9022 419	-3.5	0	0
0:00:58	1402.475 408	50.48911 469	-0.09578 7535	785.0227 835	785.0227 835	-3.5	0	0
0:00:59	1392.978 424	50.14722 328	-0.09496 9837	799.0475 376	799.0475 376	-3.5	0	0
0:01:00	1383.562 107	49.80823 586	-0.09416 3172	812.9773 218	812.9773 218	-3.5	0	0
0:01:01	1374.225 373	49.47211 343	-0.09336 7343	826.8129 429	826.8129 429	-3.5	0	0
0:01:02	1364.967 157	49.13881 766	-0.09258 2158	840.5551 966	840.5551 966	-3.5	0	0
0:01:03	1355.786 414	48.80831 092	-0.09180 7428	854.2048 682	854.2048 682	-3.5	0	0
0:01:04	1346.682 117	48.48055 623	-0.09104 297	867.7627 323	867.7627 323	-3.5	0	0
0:01:05	1337.653 257	48.15551 726	-0.09028 8603	881.2295 535	881.2295 535	-3.5	0	0
0:01:06	1328.698 842	47.83315 831	-0.08954 4152	894.6060 861	894.6060 861	-3.5	0	0
0:01:07	1319.817 897	47.51344 431	-0.08880 9445	907.8930 745	907.8930 745	-3.5	0	0
0:01:08	1311.009 466	47.19634 078	-0.08808 4312	921.0912 535	921.0912 535	-3.5	0	0
0:01:09	1302.272 607	46.88181 386	-0.08736 859	934.2013 481	934.2013 481	-3.5	0	0
0:01:10	1293.606 396	46.56983 024	-0.08666 2116	947.2240 742	947.2240 742	-3.5	0	0
0:01:11	1285.009 922	46.26035 721	-0.08596 4733	960.1601 381	960.1601 381	-3.5	0	0
0:01:12	1276.482 294	45.95336 257	-0.08527 6287	973.0102 374	973.0102 374	-3.5	0	0
0:01:13	1268.022 631	45.64881 472	-0.08459 6625	985.7750 603	985.7750 603	-3.5	0	0
0:01:14	1259.630 071	45.34668 256	-0.08392 5601	998.4552 866	998.4552 866	-3.5	0	0
0:01:15	1251.303 764	45.04693 551	-0.08326 3069	1011.051 587	1011.051 587	-3.5	0	0
0:01:16	1243.042 876	44.74954 352	-0.08260 8887	1023.564 625	1023.564 625	-3.5	0	0
0:01:17	1234.846 584	44.45447 703	-0.08196 2916	1035.995 054	1035.995 054	-3.5	0	0
0:01:18	1226.714 082	44.16170 695	-0.08132 502	1048.343 52	1048.343 52	-3.5	0	0
0:01:19	1218.644 575	43.87120 472	-0.08069 5066	1060.610 66	1060.610 66	-3.5	0	0
0:01:20	1210.637	43.58294	-0.08007	1072.797	1072.797	-3.5	0	0

	283	219	2923	106	106			
0:01:21	1202.691 437	43.29689 172	-0.07945 8464	1084.903 479	1084.903 479	-3.5	0	0
0:01:22	1194.806 281	43.01302 61	-0.07885 1563	1096.930 393	1096.930 393	-3.5	0	0
0:01:23	1186.981 071	42.73131 855	-0.07825 2097	1108.878 456	1108.878 456	-3.5	0	0
0:01:24	1179.215 076	42.45174 275	-0.07765 9946	1120.748 267	1120.748 267	-3.5	0	0
0:01:25	1171.507 577	42.17427 277	-0.07707 4993	1132.540 418	1132.540 418	-3.5	0	0
0:01:26	1163.857 865	41.89888 314	-0.07649 7121	1144.255 493	1144.255 493	-3.5	0	0
0:01:27	1156.265 243	41.62554 875	-0.07592 6218	1155.894 072	1155.894 072	-3.5	0	0
0:01:28	1148.729 026	41.35424 493	-0.07536 2173	1167.456 724	1167.456 724	-3.5	0	0
0:01:29	1141.248 538	41.08494 737	-0.07480 4877	1178.944 015	1178.944 015	-3.5	0	0
0:01:30	1133.823 116	40.81763 216	-0.07425 4224	1190.356 5	1190.356 5	-3.5	0	0
0:01:31	1126.452 105	40.55227 577	-0.07371 0109	1201.694 731	1201.694 731	-3.5	0	0
0:01:32	1119.134 862	40.28885 502	-0.07317 2429	1212.959 252	1212.959 252	-3.5	0	0
0:01:33	1111.870 753	40.02734 712	-0.07264 1085	1224.150 601	1224.150 601	-3.5	0	0
0:01:34	1104.659 155	39.76772 959	-0.07211 5978	1235.269 308	1235.269 308	-3.5	0	0
0:01:35	1097.499 454	39.50998 036	-0.07159 7011	1246.315 9	1246.315 9	-3.5	0	0
0:01:36	1090.391 045	39.25407 763	-0.07108 4089	1257.290 895	1257.290 895	-3.5	0	0
0:01:37	1083.333 333	39	-0.07057 7121	1268.194 805	1268.194 805	-3.5	0	0
0:01:37	1048.685 829	37.75268 985	-0.34647 5042	1268.194 805	1268.194 805	0	0	0
0:01:38	965.5470 66	34.75969 438	-0.83138 7631	1278.681 663	1278.681 663	-3.5	0	0
0:01:39	882.9400 244	31.78584 088	-0.82607 0416	1288.337 134	1288.337 134	-3.5	0	0
0:01:40	800.8304 622	28.82989 664	-0.82109 5622	1297.166 534	1297.166 534	-3.5	0	0
0:01:41	719.1848 84	25.89065 583	-0.81645 5782	1305.174 839	1305.174 839	-3.5	0	0
0:01:42	637.9704 84	22.96693 742	-0.81214 4	1312.366 688	1312.366 688	-3.5	0	0
0:01:43	557.1516 777	20.05746 04	-0.80818 8063	1318.746 393	1318.746 393	0	0	0
0:01:44	476.7003 042	17.16121 095	-0.80451 3735	1324.317 909	1324.317 909	0	0	0
0:01:45	396.5853 044	14.27707 096	-0.80114 9997	1329.084 912	1329.084 912	0	0	0
0:01:46	316.7761 021	11.40393 967	-0.79809 2024	1333.050 765	1333.050 765	0	0	0
0:01:47	237.2425	8.540732	-0.79533	1336.218	1336.218	0	0	0

	556	003	5464	526	526			
0:01:48	157.9549 125	5.686376 851	-0.79287 6431	1338.590 952	1338.590 952	0	0	0
0:01:49	78.88376 377	2.839815 496	-0.79071 1488	1340.170 501	1340.170 501	0	0	0
0:01:50	0	0	-0.78883 7638	1340.959 339	1340.959 339	0	0	0

A6-A8 运行数据

时 刻 (hh:mm:ss)	实 际 速 度(cm/s)	实 际 速 度(km/h)	计 算 加 速 度 (m/s ²)	计 算 距 离(m)	计 算 公 里标(m)	当 前 坡 度(‰)	计 算 牵 引 力(N)	计 算 牵 引 功 率 (Kw)
0:00:00	0	0	0	0	0	0	182700	0
0:00:01	92.04189 052	3.313508 059	0.920418 905	0.920418 905	0.920418 905	0	182700	168.1605 34
0:00:02	183.8623 6	6.619044 959	0.918204 695	2.759042 505	2.759042 505	0	182700	335.9165 317
0:00:03	275.4231 955	9.915235 038	0.915608 355	5.513274 46	5.513274 46	0	182700	503.1981 782
0:00:04	366.6865 964	13.20071 747	0.912634 009	9.180140 423	9.180140 423	0	182700	669.9364 115
0:00:05	457.6152 358	16.47414 849	0.909286 394	13.75629 278	13.75629 278	0	182700	836.0630 357
0:00:06	548.1723 21	19.73420 356	0.905570 852	19.23801 599	19.23801 599	0	182700	1001.510 83
0:00:07	638.3216 518	22.97957 946	0.901493 308	25.62123 251	25.62123 251	0	182700	1166.213 658
0:00:08	728.0276 768	26.20899 637	0.897060 25	32.90150 928	32.90150 928	0	182700	1330.106 566
0:00:09	817.2555 477	29.42119 972	0.892278 709	41.07406 475	41.07406 475	0	182700	1493.125 886
0:00:10	905.9711 708	32.61496 215	0.887156 231	50.13377 646	50.13377 646	0	182700	1655.209 329
0:00:11	994.1412 565	35.78908 524	0.881700 858	60.07518 903	60.07518 903	0	182700	1816.296 076
0:00:12	1081.733 366	38.94240 119	0.875921 097	70.89252 269	70.89252 269	0	182700	1976.326 86
0:00:13	1168.715 956	42.07377 441	0.869825 895	82.57968 225	82.57968 225	0	182700	2135.244 051
0:00:14	1255.058 417	45.18210 3	0.863424 607	95.13026 641	95.13026 641	0	182700	2292.991 727
0:00:15	1340.731 113	48.26632 008	0.856726 969	108.5375 775	108.5375 775	0	182700	2449.515 744
0:00:16	1425.705 42	51.32539 512	0.849743 066	122.7946 317	122.7946 317	0	182700	2604.763 802
0:00:17	1509.953 75	54.35833 5	0.842483 3	137.8941 692	137.8941 692	0	164379.5 303	2482.054 882
0:00:18	1509.953 75	54.35833 5	0	152.9937 067	152.9937 067	0	20471.76 525	309.1141 87
0:00:19	1509.953 75	54.35833 5	0	168.0932 442	168.0932 442	0	20471.76 525	309.1141 87
0:00:20	1509.953 75	54.35833 5	0	183.1927 818	183.1927 818	0	20471.76 525	309.1141 87
0:00:21	1509.953 75	54.35833 5	0	198.2923 193	198.2923 193	0	20471.76 525	309.1141 87

0:00:22	1509.953 75	54.35833 5	0	213.3918 568	213.3918 568	0	20471.76 525	309.1141 87
0:00:23	1509.953 75	54.35833 5	0	228.4913 943	228.4913 943	0	20471.76 525	309.1141 87
0:00:24	1509.953 75	54.35833 5	0	243.5909 318	243.5909 318	0	20471.76 525	309.1141 87
0:00:25	1509.953 75	54.35833 5	0	258.6904 693	258.6904 693	3	20477.47 752	309.2004 397
0:00:26	1509.953 75	54.35833 5	0	273.7900 068	273.7900 068	3	20477.47 752	309.2004 397
0:00:27	1509.953 75	54.35833 5	0	288.8895 443	288.8895 443	3	20477.47 752	309.2004 397
0:00:28	1509.953 75	54.35833 5	0	303.9890 818	303.9890 818	3	20477.47 752	309.2004 397
0:00:29	1509.953 75	54.35833 5	0	319.0886 193	319.0886 193	3	20477.47 752	309.2004 397
0:00:30	1509.953 75	54.35833 5	0	334.1881 568	334.1881 568	3	20477.47 752	309.2004 397
0:00:31	1509.953 75	54.35833 5	0	349.2876 943	349.2876 943	3	20477.47 752	309.2004 397
0:00:32	1509.953 75	54.35833 5	0	364.3872 318	364.3872 318	3	20477.47 752	309.2004 397
0:00:33	1509.953 75	54.35833 5	0	379.4867 693	379.4867 693	3	20477.47 752	309.2004 397
0:00:34	1509.953 75	54.35833 5	0	394.5863 068	394.5863 068	3	20477.47 752	309.2004 397
0:00:35	1509.953 75	54.35833 5	0	409.6858 443	409.6858 443	3	20477.47 752	309.2004 397
0:00:36	1509.953 75	54.35833 5	0	424.7853 818	424.7853 818	3	20477.47 752	309.2004 397
0:00:37	1509.953 75	54.35833 5	0	439.8849 193	439.8849 193	3	20477.47 752	309.2004 397
0:00:38	1509.953 75	54.35833 5	0	454.9844 568	454.9844 568	3	20477.47 752	309.2004 397
0:00:39	1509.953 75	54.35833 5	0	470.0839 943	470.0839 943	3	20477.47 752	309.2004 397
0:00:39	1509.953 75	54.35833 5	0	470.0839 943	470.0839 943	1.8	0	0
0:00:40	1504.950 598	54.17822 154	-0.05003 1518	485.1835 318	485.1835 318	1.8	0	0
0:00:41	1494.552 396	53.80388 626	-0.10398 2022	500.2330 377	500.2330 377	1.8	0	0
0:00:42	1484.247 258	53.43290 129	-0.10305 1379	515.1785 617	515.1785 617	1.8	0	0
0:00:43	1474.033 875	53.06521 949	-0.10213 3832	530.0210 343	530.0210 343	1.8	0	0
0:00:44	1463.910 961	52.70079 46	-0.10122 9137	544.7613 73	544.7613 73	1.8	0	0
0:00:45	1453.877 255	52.33958 12	-0.10033 7056	559.4004 826	559.4004 826	1.8	0	0
0:00:46	1443.931 52	51.98153 471	-0.09945 7357	573.9392 552	573.9392 552	1.8	0	0
0:00:47	1434.072 538	51.62661 138	-0.09858 9813	588.3785 704	588.3785 704	1.8	0	0
0:00:48	1424.299 118	51.27476 826	-0.09773 4201	602.7192 958	602.7192 958	1.8	0	0

0:00:49	1414.610 088	50.92596 316	-0.09689 0305	616.9622 87	616.9622 87	1.8	0	0
0:00:50	1405.004 297	50.58015 468	-0.09605 7912	631.1083 878	631.1083 878	1.8	0	0
0:00:51	1395.480 615	50.23730 214	-0.09523 6816	645.1584 308	645.1584 308	1.8	0	0
0:00:52	1386.037 934	49.89736 561	-0.09442 6814	659.1132 37	659.1132 37	1.8	0	0
0:00:53	1376.680 313	49.56049 127	-0.09357 6205	672.9736 163	672.9736 163	-3.5	0	0
0:00:54	1367.401 49	49.22645 364	-0.09278 8231	686.7404 194	686.7404 194	-3.5	0	0
0:00:55	1358.200 414	48.89521 49	-0.09201 0762	700.4144 343	700.4144 343	-3.5	0	0
0:00:56	1349.076 053	48.56673 79	-0.09124 3612	713.9964 385	713.9964 385	-3.5	0	0
0:00:57	1340.027 393	48.24098 613	-0.09048 6601	727.4871 99	727.4871 99	-3.5	0	0
0:00:58	1331.053 437	47.91792 375	-0.08973 9552	740.8874 729	740.8874 729	-3.5	0	0
0:00:59	1322.153 208	47.59751 55	-0.08900 2291	754.1980 073	754.1980 073	-3.5	0	0
0:01:00	1313.325 744	47.27972 677	-0.08827 4649	767.4195 394	767.4195 394	-3.5	0	0
0:01:01	1304.570 097	46.96452 351	-0.08755 646	780.5527 968	780.5527 968	-3.5	0	0
0:01:02	1295.885 341	46.65187 228	-0.08684 7563	793.5984 978	793.5984 978	-3.5	0	0
0:01:03	1287.270 562	46.34174 022	-0.08614 7797	806.5573 512	806.5573 512	-3.5	0	0
0:01:04	1278.724 861	46.03409 498	-0.08545 7008	819.4300 568	819.4300 568	-3.5	0	0
0:01:05	1270.247 356	45.72890 483	-0.08477 5044	832.2173 054	832.2173 054	-3.5	0	0
0:01:06	1261.837 181	45.42613 85	-0.08410 1756	844.9197 79	844.9197 79	-3.5	0	0
0:01:07	1253.493 481	45.12576 531	-0.08343 6998	857.5381 508	857.5381 508	-3.5	0	0
0:01:08	1245.215 418	44.82775 505	-0.08278 0627	870.0730 856	870.0730 856	-3.5	0	0
0:01:09	1237.002 168	44.53207 804	-0.08213 2504	882.5252 398	882.5252 398	-3.5	0	0
0:01:10	1228.852 919	44.23870 507	-0.08149 2491	894.8952 615	894.8952 615	-3.5	0	0
0:01:11	1220.766 873	43.94760 743	-0.08086 0455	907.1837 906	907.1837 906	-3.5	0	0
0:01:12	1212.743 247	43.65875 687	-0.08023 6265	919.3914 594	919.3914 594	-3.5	0	0
0:01:13	1204.781 267	43.37212 563	-0.07961 9791	931.5188 918	931.5188 918	-3.5	0	0
0:01:14	1196.880 177	43.08768 635	-0.07901 0909	943.5667 045	943.5667 045	-3.5	0	0
0:01:15	1189.039 227	42.80541 218	-0.07840 9493	955.5355 063	955.5355 063	-3.5	0	0
0:01:16	1181.257 685	42.52527 665	-0.07781 5425	967.4258 985	967.4258 985	-3.5	0	0

0:01:17	1173.534 826	42.24725 374	-0.07722 8584	979.2384 754	979.2384 754	-3.5	0	0
0:01:18	1165.869 941	41.97131 786	-0.07664 8856	990.9738 237	990.9738 237	-3.5	0	0
0:01:19	1158.262 328	41.69744 381	-0.07607 6126	1002.632 523	1002.632 523	-3.5	0	0
0:01:20	1150.711 3	41.42560 679	-0.07551 0283	1014.215 146	1014.215 146	-3.5	0	0
0:01:21	1143.216 178	41.15578 241	-0.07495 1218	1025.722 259	1025.722 259	-3.5	0	0
0:01:22	1135.776 296	40.88794 664	-0.07439 8822	1037.154 421	1037.154 421	-3.5	0	0
0:01:23	1128.390 996	40.62207 587	-0.07385 2993	1048.512 184	1048.512 184	-3.5	0	0
0:01:24	1121.059 634	40.35814 682	-0.07331 3626	1059.796 094	1059.796 094	-3.5	0	0
0:01:25	1113.781 572	40.09613 659	-0.07278 062	1071.006 69	1071.006 69	-3.5	0	0
0:01:26	1106.556 184	39.83602 263	-0.07225 3877	1082.144 506	1082.144 506	-3.5	0	0
0:01:27	1099.382 854	39.57778 275	-0.07173 33	1093.210 068	1093.210 068	-3.5	0	0
0:01:28	1092.260 975	39.32139 509	-0.07121 8793	1104.203 896	1104.203 896	-3.5	0	0
0:01:29	1085.189 948	39.06683 814	-0.07071 0263	1115.126 506	1115.126 506	-3.5	0	0
0:01:30	1078.169 187	38.81409 072	-0.07020 7619	1125.978 406	1125.978 406	-3.5	0	0
0:01:31	1071.198 11	38.56313 195	-0.06971 077	1136.760 098	1136.760 098	-3.5	0	0
0:01:32	1064.276 147	38.31394 128	-0.06921 9629	1147.472 079	1147.472 079	-3.5	0	0
0:01:33	1057.402 736	38.06649 849	-0.06873 4109	1158.114 84	1158.114 84	-3.5	0	0
0:01:34	1050.577 323	37.82078 364	-0.06825 4125	1168.688 867	1168.688 867	-3.5	0	0
0:01:35	1043.799 364	37.57677 71	-0.06777 9595	1179.194 641	1179.194 641	-3.5	0	0
0:01:36	1037.068 32	37.33445 953	-0.06731 0435	1189.632 634	1189.632 634	-3.5	0	0
0:01:37	1030.383 664	37.09381 189	-0.06684 6568	1200.003 318	1200.003 318	-3.5	0	0
0:01:38	1023.744 872	36.85481 54	-0.06638 7913	1210.307 154	1210.307 154	-3.5	0	0
0:01:39	1017.151 433	36.61745 159	-0.06593 4393	1220.544 603	1220.544 603	-3.5	0	0
0:01:40	1010.602 84	36.38170 222	-0.06548 5934	1230.716 117	1230.716 117	-3.5	0	0
0:01:41	1004.098 594	36.14754 937	-0.06504 246	1240.822 146	1240.822 146	-3.5	0	0
0:01:42	997.6382 038	35.91497 534	-0.06460 3898	1250.863 132	1250.863 132	-3.5	0	0
0:01:43	991.2211 861	35.68396 27	-0.06417 0177	1260.839 514	1260.839 514	-3.5	0	0
0:01:44	984.8470 635	35.45449 428	-0.06374 1227	1270.751 725	1270.751 725	-3.5	0	0

0:01:45	978.5153 657	35.22655 317	-0.06331 6977	1280.600 196	1280.600 196	-3.5	0	0
0:01:46	972.2222 222	35	-0.06293 1435	1290.385 35	1290.385 35	0	0	0
0:01:46	888.0700 325	31.97052 117	-0.84152 1897	1290.385 35	1290.385 35	0	0	0
0:01:47	797.4366 572	28.70771 966	-0.90633 3753	1299.266 05	1299.266 05	-3.5	0	0
0:01:48	707.3122 923	25.46324 252	-0.90124 3649	1307.240 417	1307.240 417	-3.5	0	0
0:01:49	617.6538 824	22.23553 977	-0.89658 4099	1314.313 54	1314.313 54	0	0	0
0:01:50	528.4259 712	19.02333 496	-0.89227 9112	1320.490 078	1320.490 078	0	0	0
0:01:51	439.5903 623	15.82525 304	-0.88835 609	1325.774 338	1325.774 338	0	0	0
0:01:52	351.1094 82	12.63994 135	-0.88480 8803	1330.170 242	1330.170 242	0	0	0
0:01:53	262.9463 148	9.466067 334	-0.88163 1671	1333.681 337	1333.681 337	0	0	0
0:01:54	175.0643 406	6.302316 263	-0.87881 9742	1336.310 8	1336.310 8	0	0	0
0:01:55	87.42747 308	3.147389 031	-0.87636 8676	1338.061 443	1338.061 443	0	0	0
0:01:56	0	0	-0.87427 4731	1338.935 718	1338.935 718	0	0	0
0:01:57	0	0	0	1338.935 718	1338.935 718	3	0	0
0:01:58	0	0	0	1338.935 718	1338.935 718	3	0	0
0:01:59	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:00	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:01	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:02	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:03	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:04	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:05	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:06	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:07	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:08	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:09	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:10	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:11	0	0	0	1338.935 718	1338.935 718	3	0	0

0:02:12	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:13	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:14	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:15	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:16	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:17	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:18	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:19	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:20	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:21	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:22	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:23	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:24	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:25	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:26	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:27	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:28	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:29	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:30	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:31	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:32	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:33	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:34	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:35	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:36	0	0	0	1338.935 718	1338.935 718	3	0	0
0:02:37	92.04189 052	3.313508 059	0.920418 905	0.920418 905	1339.856 137	0	182700	168.1605 34
0:02:38	183.8623 6	6.619044 959	0.918204 695	2.759042 505	1341.694 76	0	182700	335.9165 317
0:02:39	275.4231 955	9.915235 038	0.915608 355	5.513274 46	1344.448 992	0	182700	503.1981 782

0:02:40	366.6865 964	13.20071 747	0.912634 009	9.180140 423	1348.115 858	0	182700	669.9364 115
0:02:41	457.6152 358	16.47414 849	0.909286 394	13.75629 278	1352.692 011	0	182700	836.0630 357
0:02:42	548.1723 21	19.73420 356	0.905570 852	19.23801 599	1358.173 734	0	182700	1001.510 83
0:02:43	638.3216 518	22.97957 946	0.901493 308	25.62123 251	1364.556 95	0	182700	1166.213 658
0:02:44	728.0276 768	26.20899 637	0.897060 25	32.90150 928	1371.837 227	0	182700	1330.106 566
0:02:45	817.2555 477	29.42119 972	0.892278 709	41.07406 475	1380.009 783	0	182700	1493.125 886
0:02:46	905.9711 708	32.61496 215	0.887156 231	50.13377 646	1389.069 494	0	182700	1655.209 329
0:02:47	994.1412 565	35.78908 524	0.881700 858	60.07518 903	1399.010 907	0	182700	1816.296 076
0:02:48	1081.733 366	38.94240 119	0.875921 097	70.89252 269	1409.828 241	0	182700	1976.326 86
0:02:49	1168.715 956	42.07377 441	0.869825 895	82.57968 225	1421.515 4	0	182700	2135.244 051
0:02:50	1255.058 417	45.18210 3	0.863424 607	95.13026 641	1434.065 984	0	182700	2292.991 727
0:02:51	1340.731 113	48.26632 008	0.856726 969	108.5375 775	1447.473 295	0	182700	2449.515 744
0:02:52	1425.705 42	51.32539 512	0.849743 066	122.7946 317	1461.730 35	0	182700	2604.763 802
0:02:53	1509.953 75	54.35833 5	0.842483 3	137.8941 692	1476.829 887	0	164379.5 303	2482.054 882
0:02:54	1584.020 383	57.02473 38	0.740666 332	153.7343 731	1492.670 091	0	149618.4 549	2369.986 824
0:02:55	1649.801 304	59.39284 693	0.657809 203	170.2323 861	1509.168 104	0	138072.9 415	2277.929 189
0:02:56	1709.007 405	61.52426 659	0.592061 016	187.3224 602	1526.258 178	0	128814.0 298	2201.441 309
0:02:57	1762.861 8	63.46302 481	0.538543 951	204.9510 782	1543.886 796	0	121231.0 666	2137.136 163
0:02:58	1762.861 8	63.46302 481	0	222.5796 962	1561.515 414	0	25240.99 772	444.9639 068
0:02:58	1762.861 8	63.46302 481	0	222.5796 962	1561.515 414	0	0	0
0:02:59	1755.564 8	63.20033 282	-0.07296 9998	240.2083 142	1579.144 032	3	0	0
0:03:00	1742.776 465	62.73995 273	-0.12788 3357	257.7639 622	1596.699 68	3	0	0
0:03:01	1730.116 812	62.28420 524	-0.12659 6526	275.1917 268	1614.127 445	3	0	0
0:03:02	1717.583 831	61.83301 79	-0.12532 9816	292.4928 95	1631.428 613	3	0	0
0:03:03	1705.175 549	61.38631 977	-0.12408 2813	309.6687 333	1648.604 451	3	0	0
0:03:04	1692.890 038	60.94404 138	-0.12285 511	326.7204 887	1665.656 207	3	0	0
0:03:05	1680.725 407	60.50611 465	-0.12164 6313	343.6493 891	1682.585 107	3	0	0
0:03:06	1668.679 803	60.07247 292	-0.12045 6037	360.4566 432	1699.392 361	3	0	0

0:03:07	1656.751 413	59.64305 085	-0.11928 3906	377.1434 412	1716.079 159	3	0	0
0:03:08	1644.938 457	59.21778 446	-0.11812 9555	393.7109 554	1732.646 673	3	0	0
0:03:09	1633.239 194	58.79661 1	-0.11699 2627	410.1603 399	1749.096 058	3	0	0
0:03:10	1621.651 917	58.37946 902	-0.11587 2774	426.4927 319	1765.428 45	3	0	0
0:03:11	1610.174 951	57.96629 825	-0.11476 9657	442.7092 51	1781.644 969	3	0	0
0:03:12	1598.806 657	57.55703 964	-0.11368 2946	458.8110 006	1797.746 718	3	0	0
0:03:13	1587.545 425	57.15163 53	-0.11261 2318	474.7990 671	1813.734 785	3	0	0
0:03:14	1576.389 679	56.75002 845	-0.11155 7458	490.6745 214	1829.610 239	3	0	0
0:03:15	1565.337 873	56.35216 344	-0.11051 8059	506.4384 182	1845.374 136	3	0	0
0:03:16	1554.388 491	55.95798 569	-0.10949 3821	522.0917 969	1861.027 515	3	0	0
0:03:17	1543.540 046	55.56744 166	-0.10848 4452	537.6356 818	1876.571 4	3	0	0
0:03:18	1532.791 079	55.18047 885	-0.10748 9668	553.0710 823	1892.006 8	3	0	0
0:03:19	1522.140 16	54.79704 578	-0.10650 9188	568.3989 931	1907.334 711	3	0	0
0:03:20	1511.585 886	54.41709 191	-0.10554 2741	583.6203 947	1922.556 113	3	0	0
0:03:21	1501.126 88	54.04056 769	-0.10459 0062	598.7362 535	1937.671 971	3	0	0
0:03:22	1490.761 791	53.66742 448	-0.10365 0891	613.7475 223	1952.683 24	3	0	0
0:03:23	1480.494 15	53.29778 939	-0.10267 6412	628.6551 402	1967.590 858	-2	0	0
0:03:24	1470.317 757	52.93143 924	-0.10176 3933	643.4600 817	1982.395 8	-2	0	0
0:03:25	1460.231 335	52.56832 808	-0.10086 4211	658.1632 593	1997.098 977	-2	0	0
0:03:26	1450.233 634	52.20841 084	-0.09997 7012	672.7655 727	2011.701 291	-2	0	0
0:03:27	1440.323 424	51.85164 326	-0.09910 2104	687.2679 09	2026.203 627	-2	0	0
0:03:28	1430.499 498	51.49798 191	-0.09823 9263	701.6711 432	2040.606 861	-2	0	0
0:03:29	1420.760 671	51.14738 414	-0.09738 8269	715.9761 382	2054.911 856	-2	0	0
0:03:30	1411.105 78	50.79980 808	-0.09654 8907	730.1837 449	2069.119 463	-2	0	0
0:03:31	1401.533 683	50.45521 26	-0.09572 0967	744.2948 027	2083.230 521	-2	0	0
0:03:32	1392.043 259	50.11355 732	-0.09490 4242	758.3101 396	2097.245 857	-2	0	0
0:03:33	1382.633 406	49.77480 26	-0.09409 8534	772.2305 722	2111.166 29	-2	0	0
0:03:34	1373.303 041	49.43890 948	-0.09330 3644	786.0569 062	2124.992 624	-2	0	0

0:03:35	1364.051 103	49.10583 971	-0.09251 9381	799.7899 366	2138.725 654	-2	0	0
0:03:36	1354.876 547	48.77555 57	-0.09174 5558	813.4304 477	2152.366 166	-2	0	0
0:03:37	1345.778 348	48.44802 054	-0.09098 1991	826.9792 131	2165.914 931	-2	0	0
0:03:38	1336.755 498	48.12319 794	-0.09022 8499	840.4369 966	2179.372 714	-2	0	0
0:03:39	1327.807 007	47.80105 227	-0.08948 4908	853.8045 516	2192.740 269	-2	0	0
0:03:40	1318.931 903	47.48154 85	-0.08875 1046	867.0826 217	2206.018 34	-2	0	0
0:03:41	1310.129 228	47.16465 222	-0.08802 6744	880.2719 407	2219.207 659	-2	0	0
0:03:42	1301.398 045	46.85032 96	-0.08731 1838	893.3732 33	2232.308 951	-2	0	0
0:03:43	1292.737 428	46.53854 74	-0.08660 6167	906.3872 134	2245.322 931	-2	0	0
0:03:44	1284.146 47	46.22927 294	-0.08590 9574	919.3145 877	2258.250 306	-2	0	0
0:03:45	1275.624 28	45.92247 409	-0.08522 1903	932.1560 524	2271.091 77	-2	0	0
0:03:46	1267.169 98	45.61811 927	-0.08454 3004	944.9122 952	2283.848 013	-2	0	0
0:03:47	1258.782 707	45.31617 744	-0.08387 273	957.5839 95	2296.519 713	-2	0	0
0:03:48	1250.461 613	45.01661 808	-0.08321 0935	970.1718 221	2309.107 54	-2	0	0
0:03:49	1242.205 865	44.71941 116	-0.08255 7478	982.6764 382	2321.612 156	-2	0	0
0:03:50	1234.014 643	44.42452 716	-0.08191 2221	995.0984 969	2334.034 215	-2	0	0
0:03:51	1225.887 141	44.13193 707	-0.08127 5026	1007.438 643	2346.374 361	-2	0	0
0:03:52	1217.822 564	43.84161 232	-0.08064 5763	1019.697 515	2358.633 233	-2	0	0
0:03:53	1209.820 135	43.55352 485	-0.08002 4299	1031.875 74	2370.811 458	-2	0	0
0:03:54	1201.879 084	43.26764 702	-0.07941 0507	1043.973 942	2382.909 66	-2	0	0
0:03:55	1193.998 658	42.98395 167	-0.07880 4263	1055.992 733	2394.928 45	-2	0	0
0:03:56	1186.178 113	42.70241 208	-0.07820 5444	1067.932 719	2406.868 437	-2	0	0
0:03:57	1178.416 72	42.42300 193	-0.07761 3929	1079.794 5	2418.730 218	-2	0	0
0:03:58	1170.713 76	42.14569 537	-0.07702 9601	1091.578 667	2430.514 385	-2	0	0
0:03:59	1163.068 526	41.87046 692	-0.07645 2346	1103.285 805	2442.221 523	-2	0	0
0:04:00	1155.480 321	41.59729 154	-0.07588 2049	1114.916 49	2453.852 208	-2	0	0
0:04:01	1147.948 461	41.32614 458	-0.07531 8601	1126.471 294	2465.407 011	-2	0	0
0:04:02	1140.472 271	41.05700 177	-0.07476 1893	1137.950 778	2476.886 496	-2	0	0

0:04:03	1133.051 09	40.78983 922	-0.07421 1818	1149.355 501	2488.291 219	-2	0	0
0:04:04	1125.684 262	40.52463 344	-0.07366 8272	1160.686 012	2499.621 73	-2	0	0
0:04:05	1118.371 147	40.26136 13	-0.07313 1152	1171.942 854	2510.878 572	-2	0	0
0:04:06	1111.111 111	40	-0.07260 036	1183.126 566	2522.062 284	-2	0	0
0:04:06	1111.111 111	40	0	1183.126 566	2522.062 284	0	0	0
0:04:07	1026.955 731	36.97040 633	-0.84155 3798	1194.237 677	2533.173 395	0	0	0
0:04:08	943.3979 687	33.96232 687	-0.83557 7627	1204.507 234	2543.442 952	0	0	0
0:04:09	860.4014 171	30.97445 102	-0.82996 5515	1213.941 214	2552.876 932	0	0	0
0:04:10	777.9305 822	28.00550 096	-0.82470 8349	1222.545 228	2561.480 946	0	0	0
0:04:11	695.9508 153	25.05422 935	-0.81979 7669	1230.324 534	2569.260 252	0	0	0
0:04:12	614.4282 516	22.11941 706	-0.81522 5637	1237.284 042	2576.219 76	0	0	0
0:04:13	533.3297 504	19.19987 102	-0.81098 5012	1243.428 325	2582.364 042	0	0	0
0:04:14	452.6228 383	16.29442 218	-0.80706 9122	1248.761 622	2587.697 34	0	0	0
0:04:15	372.2756 539	13.40192 354	-0.80347 1844	1253.287 85	2592.223 568	0	0	0
0:04:16	292.2568 955	10.52124 824	-0.80018 7583	1257.010 607	2595.946 325	0	0	0
0:04:17	212.5357 702	7.651287 729	-0.79721 1253	1259.933 176	2598.868 894	0	0	0
0:04:18	133.0819 445	4.790950 003	-0.79453 8257	1262.058 534	2600.994 252	0	0	0
0:04:19	53.86549 671	1.939157 881	-0.79216 4478	1263.389 353	2602.325 071	0	0	0
0:04:20	0	0	-0.53865 4967	1263.928 008	2602.863 726	0	0	0
0:04:21	292.2568 955	10.52124 824	-0.80018 7583	1257.010 607	2595.946 325	0	0	0
0:04:22	212.5357 702	7.651287 729	-0.79721 1253	1259.933 176	2598.868 894	0	0	0
0:04:23	133.0819 445	4.790950 003	-0.79453 8257	1262.058 534	2600.994 252	0	0	0
0:04:24	53.86549 671	1.939157 881	-0.79216 4478	1263.389 353	2602.325 071	0	0	0
0:04:25	0	0	-0.53865 4967	1263.928 008	2602.863 726	0	0	0

主要程序代码

注：只列出了每一小问的主函数，相关 m 子函数请查看附件文件

第一题第一问主程序：

```
function Q1_1()%µÛÖ»ÎâµÛÖ»ÎÊ
tic;
clc;close all;
global T;
si=13594;
se=12240;
L=si-se;
v0=0;
vmax=80;
ws=inf;
Vm=v0;
Vn=0;
T=1;
S1=0;S2=0;S3=0;S4=0;VData=[];U1=0;U2=0;
C=1e-3*3600;%m/s×a»»Îakm/hµ.ÄÎµÊÝ
tt=[];wt=[];
for u1=0.9:0.1:0.9
    for u2=0.9:0.1:0.9
        for vm=45:vmax

[sData1,tData1,vData1,aData1,WData1,VData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,si);%»ñµ
ÄÇ£ÖÝ¶ÎµÄÎ»ÖÆ/Ê±¼ä/ËÜ¶ÈÊÝ¾Ý
            W1=max(WData1);
            if(exitflag1<0)
                continue;
            end
            if(exitflag1==0)
                vm=vData1(length(vData1));
            end
            for vn=20:vm

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest_1(vn,u2,se); %»ñµÄÖÆ¶ÎÎµÄÎ»ÖÆ
/Ê±¼ä/ËÜ¶ÈÊÝ¾Ý

                if(exitflag4<0)
                    continue;
                end

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,se+max(sData4));
                    if(exitflag3<0)
                        continue;
                    end

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñµÄÑ201½¶ÎµÄÎ»ÖÆ/Ê±¼ä/ËÜ¶ÈÊÝ¾Ý
                    if(t2<0)
                        continue;
                    end
                    t=max(tData1)+t2+max(tData3)+max(tData4);
                    s2=t2*max(vData1)/C;
                    vData2=max(vData1)*ones(1,floor(t2/T));
                    aData2=zeros(1,floor(t2/T)-1);
                    tData2=T:t2;
                    sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,si-max(sData1));%»ñË;Ñ201½¶ÎËÜ×öµÄ¹
|

```



```

        if(exitflag2<0)
        continue;
    end
    W2=max(WData2);
    W=W1+W2;

    if(t<=111)
        tt=[tt,t];
        wtt=[wtt,W];
        if(ws>W)
            ws=W;
            Vm=vm
            Vn=vn
            S1=max(sData1);
            S2=s2;
            S3=max(sData3);
            S4=max(sData4);
            U1=u1;
            U2=u2;
            VData=[vData1,vData2,vData3,vData4];
            VLData=[VData1,vlData2,vlData3,vlData4];
            aData=[aData1,aData2,-aData3,-aData4];
            pData=[pData1,pData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
            fData=[fData1,fData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
            kData=[kData1,kData2,kData3,kData4];
            if(isempty(tData2))

tData=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
                else

tData=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)+max(
tData3)];

sData=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sData2)+m
ax(sData3)];

                end

wData=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(WD
ata2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];

                end
            end
        end
    end
end
if(isempty(VData))
    disp(['2»Äú×âÖªÇó']);
else
    disp(['×İĐİİû°ÄÄÜÄ¿îª',num2str(ws),'kJ'] );
    disp(['vm=',num2str(Vm),'vn=',num2str(Vn),'s1=',num2str(S1),'s2=',num2str(S2),'s3=',num2str(S3),'s4=',num
2str(S4),'u1=',num2str(U1),'u2=',num2str(U2)]);
    disp(['¹½³µÖ¾Êµ¼ÊÔÊÐÊ±¼ä',num2str(max(tData)),'s']);
    disp(['¹½³µÖ¾Êµ¼ÊÔÊÐÊ¾äÄë',num2str(max(sData)),'m']);
    figure
    plot(tData,VData);grid on;xlabel('Ê±¼ä/s'),ylabel('ËÜ¶Ê/km/h');hold on;plot(tData,VLData,'r');hold off;
    figure
    plot(sData,VData);grid on;xlabel('¾äÄë/m'),ylabel('ËÜ¶Ê/km/h');hold on;plot(sData,VLData,'r');hold off;

```



```

        continue;
    end
    for vn=35:35

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest_1(vn,u2,se); %»ñµÃÖÆ¶¶ÎµÃÎ»ÒÆ
/Ê±¼ä/ËÛ¶ËËý¾Ý
        if(exitflag4<0)
            continue;
        end
        %»ñË;Ç£Òý¶Îµ½ÖÆ¶¶ÎÖ®¼äµÃÂ·¿öÐÃĬ¢

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,se+max(sData4));
        if(exitflag3<0)
            continue;
        end

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñµÃÑ²º½¶ÎµÃÎ»ÒÆ/Ê±¼ä/ËÛ¶ËËý¾Ý
        if(t2<0)
            continue;
        end
        t=max(tData1)+t2+max(tData3)+max(tData4);
        s2=t2*max(vData1)/C;
        vData2=max(vData1)*ones(1,floor(t2/T));
        aData2=zeros(1,floor(t2/T)-1);
        tData2=T:T:t2;
        sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,se-max(sData1));%»ñË;Ñ²º½¶ÎËü×öµÃ
"
        if(exitflag2<0)
            continue;
        end
        W2=max(WData2);
        W=W1+W2;
        dt=abs(t-t1);
        if(dt<=1)
            if(ws>W)
                ws=W;
                Vm=vm
                Vn=vn
                S1=max(sData1);
                S2=s2;
                S3=max(sData3);
                S4=max(sData4);
                U1=u1;
                U2=u2;
                VData=[vData1,vData2,vData3,vData4];
                vlData_1=[VlData1,vlData2,vlData3,vlData4];
                aData=[aData1,aData2,-aData3,-aData4];
                pData_1=[pData1,pData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
                fData_1=[fData1,fData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
                kData_1=[kData1,kData2,kData3,kData4];

wData_1=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(W
Data2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
                if isempty(tData2))

tData=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

```

[illegible]

```

end
W2=max(WData2);
W=W1+W2;
t2=220-t1;
dt=abs(t-t2);
if(dt<=1)
    if(ws2>W)
        ws2=W;
        Vm2=vm;
        Vn2=vn;
        S21=max(sData1);
        S22=s2;
        S23=max(sData3);
        S24=max(sData4);
        U21=u1;
        U22=u2;
        VData_2=[vData1,vData2,vData3,vData4];
        vlData_2=[vlData1,vlData2,vlData3,vlData4];
        aData_2=[aData1,aData2,-aData3,-aData4];

wData_2=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(W
Data2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
pData_2=[pData1,pData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
fData_2=[fData1,fData2,zeros(1,length(tData3)),zeros(1,length(tData4))];
kData_2=[kData1,kData2,kData3,kData4];
if isempty(tData2))

tData_2=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData_2=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
else

tData_2=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)+ma
x(tData3)];

sData_2=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sData2)+
max(sData3)];

end

end

end
% end
end
end
end
end
w=ws+ws2;
if(wt>w)%»ñÊ;Á½Ö¾×iÓÂ°ÄÄÜμÄ×iĐ;Öμ
    wt=w;
    T1=t1;
    T2=220-t1;
    TData1=tData;
    TData2=tData_2;
    SData1=sData;
    SData2=sData_2;
    TData=[tData,max(tData)+(T:T:40),tData_2+max(tData)+40];
    SData=[sData,max(sData)*ones(1,40/T),sData_2+max(sData)];
    NSData=[sData,max(sData)*ones(1,40/T),sData_2];
    VData=[VData,zeros(1,40/T),VData_2];
    VLData=[vlData_1,zeros(1,40/T),vlData_2];

```

```

aDats=[aData,zeros(1,40/T),aData_2];
wData=[wData_1,max(wData_1)*ones(1,40/T),wData_2+max(wData_1)];
kData=[kData_1,max(kData_1)*ones(1,40/T),kData_2];
fData=[fData_1,zeros(1,40/T),fData_2];
pData=[pData_1,zeros(1,40/T),pData_2];

VM1=Vm;
VM2=Vm2;
VN1=Vn;
VN2=Vn2;
UU1=U1;
UU2=U2;
UU21=U21;
UU22=U22;

end
end
%IÖÊ¼ĐÄİç
disp(['×İĐİİâ°ÄÄÜÄ¿İª',num2str(wt),'kJ'] );
disp(['vm=',num2str(VM1),'vn=',num2str(VN1),'u1=',num2str(UU1),'u2=',num2str(UU2)]);
disp(['Ç°Á½³μÖ¼¹æ¶"ÔÊĐĐÊ±¼ä',num2str(max(T1)),'s','Ç°Á½³μÖ¼¹æμ¼ÊÔÊĐĐÊ±¼ä',num2str(max(TData
1)),'s']);
disp(['Ç°Á½³μÖ¼¹æ¶"ÔÊĐĐ¼àÀě',num2str(L),'m','Ç°Á½³μÖ¼¹æμ¼ÊÔÊĐĐ¼àÀě',num2str(max(SData1)),'m']
);
disp(['vm=',num2str(VM2),'vn=',num2str(VN2),'u1=',num2str(UU21),'u2=',num2str(UU22)]);
disp(['°óÁ½³μÖ¼¹æ¶"ÔÊĐĐÊ±¼ä',num2str(max(T2)),'s','°óÁ½³μÖ¼¹æμ¼ÊÔÊĐĐÊ±¼ä',num2str(max(TData2)
),'s']);
disp(['°óÁ½³μÖ¼¹æ¶"ÔÊĐĐ¼àÀě',num2str(L2),'m','°óÁ½³μÖ¼¹æμ¼ÊÔÊĐĐ¼àÀě',num2str(max(SData2)),'m']
);
%»æÖÆİ¼Đİ
figure
plot(TData,VDatas);grid on;xlabel('Ê±¼ä/s'),ylabel('ËÜ¶Ê/km/h');hold on;plot(TData,VLDatas,'r');hold off;
figure
plot(SData,VDatas);grid on;xlabel('¼àÀě/m'),ylabel('ËÜ¶Ê/km/h');hold on;plot(SData,VLDatas,'r');hold off;
figure
plot(0:T:(length(aDats)-1)*T,aDats);grid on;xlabel('Ê±¼ä/s'),ylabel('¼ÓËÜ¶Ê/m/s^2');
figure
plot(SData,wData);grid on;xlabel('¼àÀě/m'),ylabel('ËÜ¶Ê/km/h');
figure
plot(TData,wData);grid on;xlabel('Ê±¼ä/s'),ylabel('ËÜ¶Ê/km/h');

%½«Êý¼Ý°,ñÊ½Đ'½øxIsİÄμμ
len=length(TData);
adata=[0,diff(VDatas)/(3.6*T)];
for i=1:len
    m=floor(TData(i)/60);
    s=mod(TData(i),60);
    if(s<10)
        tsData{i}=['00:0',num2str(m),':0',num2str(s)];
    else
        tsData{i}=['00:0',num2str(m),':',num2str(s)];
    end
end
end
str=['A8:A',num2str(7+len)];
xlswrite('μÜÖ»İâμÜ¶İÊÊäÊëÊý¼ÝÄ°â.xlsx',tsData',2,str);
writeData=[zeros(len,5),VDatas'*1e2/3.6,VDatas',adata',NSData',SData',zeros(len,3),kData',zeros(len,1),1e3*fD
ata',pData',zeros(len,1)];
str=['B8:S',num2str(7+len)];
xlswrite('μÜÖ»İâμÜ¶İÊÊäÊëÊý¼ÝÄ°â.xlsx',writeData,2,str);
disp(['³İĐòÔÊĐĐÊ±¼ä',num2str(toc),'s']);

```

第二题第一问

```
function h=Q2_1()% ,ù¼ÝÁÐ³μÔÚ ,÷¶ÎμÄÔËÐÐÊ±¼ä£¬È·¶Á½Á¾³μ²È«ÔËÐÐÊ±¼ä
global T;
E=0;
tic;
sis=[22903 21569 20283 18197 15932 13594 12240 10960 9422 8429 6447
4081 2806];
ses=[21569 20283 18197 15932 13594 12240 10960 9422 8429 6447 4081
2806 175];
v0=0;close all;
vmax=80;g=9.8;
T=5;
C=1e-3*3600;%m/s×ª»»Îªkm/hμÄÏμËÝ
ds=sis-ses;
Vm=zeros(1,length(ds));
Vn=zeros(1,length(ds));
U1=zeros(1,length(ds));
U2=zeros(1,length(ds));
eGenerate=zeros(1,length(ds));
wdata=zeros(1,length(ds));
td=zeros(1,length(ds));
swdata=[];%¼ÇÄ¼Ö»¬μÄ¹ºÄ±ä»¬ÇË
m=194.295;
TData=[];
VDatas=[];
SDatas=[];
WData=[];
tData=[];
tMin=inf;
VM=load('Vm.mat');
VN=load('Vn.mat');
VM=VM.Vm;
VN=VN.Vn;
UU1=load('UU1.mat');
UU2=load('UU2.mat');
tRun=load('×;ÓÄÔËÐÐÊ±¼ä.mat');
tRun=tRun.tmean;
UU1=UU1.U1;
UU2=UU2.U2;
st=0;
VLdata=[];
for ti=1:length(ds)
    ws=inf;
    for u1=UU1(ti):UU1(ti)
        for u2=UU2(ti):UU2(ti)
            for vm=VM(ti):VM(ti)

[sData1,tData1,vData1,aData1,WData1,vlData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,sis(ti));%»
ñμÄÇ£ÖÝ¶ÎμÄÎ»ÖÆ/Ê±¼ä/ËÜ¶ËËÝ¾Ý
                W1=max(WData1);
                if(exitflag1<0)
                    continue;
                end
                for vn=VN(ti):VN(ti)

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest_1(vn,u2,ses(ti)); %»ñμÄÖÆ¶ÎμÄÎ»
ÖÆ/Ê±¼ä/ËÜ¶ËËÝ¾Ý
                    if(exitflag4<0)
                        continue;
                    end
```

```

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,ses(ti)+max(sData4))
;
    if(exitflag3<0)
        continue;
    end
    L=sis(ti)-ses(ti);

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñμÃÑ²º½¶îμÃÎ»ÔÆ/Ê±¼ä/ËÙ¶ËËý³¼Ý
    if(t2<0)
        continue;
    end
    t=max(tData1)+t2+max(tData3)+max(tData4);
    s2=t2*max(vData1)/C;
    vData2=max(vData1)*ones(1,floor(t2/T));
    aData2=zeros(1,floor(t2/T)-1);
    tData2=T:T:t2;
    sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,ses(ti)-max(sData1));%»ñÊ;Ñ²º½¶îËù×ö
μÃ¹|
    if(exitflag2<0)
        continue;
    end
    W2=max(WData2);
    W=W1+W2;
    ddt=abs(t-tRun(ti));
    if(ddt<=T)
        if(ws>W)
            ti
            ws=W;
            U1(ti)=u1;
            U2(ti)=u2;
            Vm(ti)=vm;
            Vn(ti)=vn;
            S1=max(sData1);
            S2=s2;
            S3=max(sData3);
            S4=max(sData4);

WData{ti}=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(
WData2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
            VData{ti}=[vData1,vData2,vData3,vData4];
            vlData{ti}=[vlData1,vlData2,vlData3,vlData4];
%            VLdata=[VLdata,vld];
            aData{ti}=[aData1,aData2,-aData3,-aData4];

            if(isempty(tData2))
                tmax=max(tData4)+max(tData1)+max(tData3);

tData{ti}=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData{ti}=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
                else
                    tmax=max(tData4)+max(tData1)+max(tData3)+max(tData2);

tData{ti}=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)+
max(tData3)];

sData{ti}=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sData2)]

```



```

+max(sData3)];

end
if(ti>1)
    sData{ti}=sData{ti}+max(sData{ti-1});
    WData{ti}=WData{ti}+max(WData{ti-1});
end

wdata(ti)=ws;
st=st+tmax;
td(ti)=tmax;
%±£'æ,Ã¶ÎµÄÖÆ¶Ê±¼ä¶Î½Úµã/ÔÙÊúÄÜÁ|ÒÔ¼°¹|°ÄÊ±¼ä
if(ti>1)
    tData{ti}=tData{ti}+max(tData{ti-1});
    tDrag{ti}=[tmax-max(tData4)+st,tmax+st];
    tLoss{ti}=[min(tData1)+st,max(tData1)+max(tData2)+st];
else
    tDrag{ti}=[tmax-max(tData4),tmax];
    tLoss{ti}=[min(tData1),max(tData1)+max(tData2)];
end
[vl,r,k]=getRoadCon(ses(ti)+max(sData4));%»ñÊ;ÖÆ¶¶ÎµÄÆÂ¶Ê

Emesh=0.5*m*min(vData3)^2/3.6^2-m*g*max(sData4)*sin(pi*k/180);%¼ÆËäÖÆ¶¶ÎÁÐ³µ»úÐµÄÜµÄ¼δÐ
;Á¿£¬×¢Òà·ù°Ä

eGenerate(ti)=(Emesh-Ef)*0.95;

end
end
end
end
end
end
if(ti==length(ds))
    WData=[WData,WData{ti}];
    SData=[SData,sData{ti}];
    VData=[VData,VData{ti}];
    %    tData=[tData,tData{ti}];
    VLdata=[VLdata,vld{ti}];

    d=tData{ti}+35*(ti-1);
    tData=[tData,d];
else
    SData=[SData,sData{ti},max(sData{ti})*ones(1,35/T-1)];
    WData=[WData,WData{ti},max(WData{ti})*ones(1,35/T-1)];
    VData=[VData,VData{ti},zeros(1,35/T-1)];
    VLdata=[VLdata,vld{ti},zeros(1,35/T-1)];
    a=T:T:30;
    b=max(tData{ti}+35*(ti-1))*ones(1,30/T);
    c=a+b;
    d=tData{ti}+35*(ti-1);
    tData=[tData,d,c];
end
end
disp(['³ÐòÔËÐÊ±¼ä',num2str(toc),'s']);
disp(['×ÜÄÜ°Ä',num2str(sum(wdata)),'kJ']);

figure
plot(tData)
figure

```

```

H=[];Es=[];
h=[605    660    660    660    640];
E=GetE(0,h,5);
Es=[Es,E];
H=[H,h];
h=[645    660    645    660    615];
E=GetE(0,h,5);
Es=[Es,E];

```

```

H=[H;h];
h=[645 645 645 645 645];
E=GetE(0,h,5);
Es=[Es,E];
H=[H;h];
h0=[645 645 645 645 645];
%Ëæ»ú²úÊú·ç³μ¼ä,ôÊý¾Ý
for i=1:18
    a=ceil(rand(1,2)*5);
    t=ceil(rand()*15);
    h(a(1))=h0(a(1))+t;
    h(a(2))=h0(a(2))-t;
    E=GetE(0,h,5);
    Es=[Es,E];
    H=[H;h];
end
H=[H,Es'];
xlswrite('ÄÖë·ç³μ¼ä,ô.xls',H);

```

第二题第二问主程序

```

function Q2_2()
global T;
clc;
tic;
sis=[22903 21569 20283 18197 15932 13594 12240 10960 9422 8429 6447
4081 2806];
ses=[21569 20283 18197 15932 13594 12240 10960 9422 8429 6447 4081
2806 175];
v0=0;close all;
vmax=80;g=9.8;
T=5;
C=1e-3*3600;%m/s×a»»Îªkm/hμÄÏμÊý
ds=sis-ses;
Vm=zeros(1,length(ds));
Vn=zeros(1,length(ds));
U1=zeros(1,length(ds));
U2=zeros(1,length(ds));
eGenerate=zeros(1,length(ds));
wdata=zeros(1,length(ds));
td=zeros(1,length(ds));
swdata=[];%¼ÇÂ¼ÖÂ»¯μÄ¹ºÄ±ä»Çéö
m=194.295;
TData=[];
VData=[];
SDatas=[];
WData=[];
tData=[];
tMin=inf;
VM=load('Vm.mat');
VN=load('Vn.mat');
VM=VM.Vm;
VN=VN.Vn;
UU1=load('UU1.mat');
UU2=load('UU2.mat');
tRun=load('×îÖÂÖÊÐÊ±¼ä.mat');
tRun=tRun.tmean;
UU1=UU1.U1;
UU2=UU2.U2;
st=0;

```

```

VLdata=[];
for ti=1:length(ds)
    ws=inf;
    for u1=UU1(ti):UU1(ti)
        for u2=UU2(ti):UU2(ti)
            for vm=VM(ti):VM(ti)

[sData1,tData1,vData1,aData1,WData1,vlData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,ses(ti));%»
ñµÃÇÖý¶ÎµÃÎ»ÒÆ/Ê±¼ä/ËÜ¶ËËË¼Ý
                W1=max(WData1);
                if(exitflag1<0)
                    continue;
                end
                for vn=VN(ti):VN(ti)

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest(vn,u2,ses(ti)); %»ñµÃÖÆ¶ÎµÃÎ»Ò
Æ/Ê±¼ä/ËÜ¶ËËË¼Ý
                    if(exitflag4<0)
                        continue;
                    end

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,ses(ti)+max(sData4))
;

                    if(exitflag3<0)
                        continue;
                    end
                    L=ses(ti)-ses(ti);

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñµÃÑ²º½¶ÎµÃÎ»ÒÆ/Ê±¼ä/ËÜ¶ËËË¼Ý
                    if(t2<0)
                        continue;
                    end
                    t=max(tData1)+t2+max(tData3)+max(tData4);
                    s2=t2*max(vData1)/C;
                    vData2=max(vData1)*ones(1,floor(t2/T));
                    aData2=zeros(1,floor(t2/T)-1);
                    tData2=T:T:t2;
                    sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,ses(ti)-max(sData1));%»ñËÑ²º½¶ÎËù×ö
µÃ¹|
                    if(exitflag2<0)
                        continue;
                    end
                    W2=max(WData2);
                    W=W1+W2;
                    ddt=abs(t-tRun(ti));
                    if(ddt<=T)
                        if(ws>W)
                            ti
                            ws=W;
                            U1(ti)=u1;
                            U2(ti)=u2;
                            Vm(ti)=vm;
                            Vn(ti)=vn;
                            S1=max(sData1);
                            S2=s2;
                            S3=max(sData3);
                            S4=max(sData4);

```

```

WData{ti}=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(
WData2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
VData{ti}=[vData1,vData2,vData3,vData4];
vld{ti}=[vlData1,vlData2,vlData3,vlData4];
%
VLdata=[VLdata,vld];
aData{ti}=[aData1,aData2,-aData3,-aData4];

if(isempty(tData2))
    tmax=max(tData4)+max(tData1)+max(tData3);

tData{ti}=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData{ti}=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
else
    tmax=max(tData4)+max(tData1)+max(tData3)+max(tData2);

tData{ti}=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)+
max(tData3)];

sData{ti}=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sData2)
+max(sData3)];

end
if(ti>1)
    sData{ti}=sData{ti}+max(sData{ti-1});
    WData{ti}=WData{ti}+max(WData{ti-1});
end

wdata(ti)=ws;
st=st+tmax;
td(ti)=tmax;
%±£'æ,Ä¶ÎµÄÖÆ¶Ê±¼ä¶Î½Úµã/ÔÛÉúÄÜÁ!ÒÔ¼°¹ºÄÊ±¼ä
if(ti>1)
    tData{ti}=tData{ti}+max(tData{ti-1});
    tDrag{ti}=[tmax-max(tData4)+st,tmax+st];
    tLoss{ti}=[min(tData1)+st,max(tData1)+max(tData2)+st];
else
    tDrag{ti}=[tmax-max(tData4),tmax];
    tLoss{ti}=[min(tData1),max(tData1)+max(tData2)];
end
[vl,r,k]=getRoadCon(ses(ti)+max(sData4));%»ñË;ÖÆ¶¶ÎµÄÆÂ¶Ë

Emesh=0.5*m*min(vData3)^2/3.6^2-m*g*max(sData4)*sin(pi*k/180);%¼ÆËäÖÆ¶¶ÎÁÐ³µ»úÐµÄÜµÄ¼öÐ
;ÄÇ£¬×çÒâ·û°Ä

eGenerate(ti)=(Emesh-Ef)*0.95;

end
end
end
end
end
end
if(ti==length(ds))
    WDatas=[WDatas,WData{ti}];
    SDatas=[SDatas,sData{ti}];
    VDatas=[VDatas,VData{ti}];
%
    tDatas=[tDatas,tData{ti}];
    VLdata=[VLdata,vld{ti}];

    d=tData{ti}+35*(ti-1);
    tDatas=[tDatas,d];

```

```

else
    SDatas=[SDatas,sData{ti},max(sData{ti})*ones(1,35/T-1)];
    WData=[WData,WData{ti},max(WData{ti})*ones(1,35/T-1)];
    VData=[VData,VData{ti},zeros(1,35/T-1)];
    VLdata=[VLdata,vld{ti},zeros(1,35/T-1)];
    a=T:T:30;
    b=max(tData{ti}+35*(ti-1))*ones(1,30/T);
    c=a+b;
    d=tData{ti}+35*(ti-1);
    tDatas=[tDatas,d,c];
end
end

%¼ÆËàÔÙÉúÄÜÁ¿£¬·ÖÎâ¶Î½øÐÐ¼ÆËà
nn=[8,15,5,15,5];
h{1}=342*ones(1,21);
h{2}=131*ones(1,41);
h{3}=355*ones(1,86);
h{4}=133*ones(1,54);
h{5}=355*ones(1,38);
E=[];
for i=1:5
    ht=0;
    for j=1:i-1
        ht=ht+sum(h{j});
    end
    e=GetE(0,h{i},nn(i));
    E=[E,e];
end

WE=240*sum(wdata)-sum(E);%¼ÆËàÊµ¼ÊÏû°Ä¹°Ä
%²úÉúËæ»úÊ±¼ä¶ÎµÄÔÙÉúÄÜÁ¿
cNum=[21,41,86,54,38];
tNum=[7200 5400 30600 7200 13500];

hn=round(tNum./cNum);
wdatas=[sum(E)];LN=cNum;
for ii=1:18
    a=ceil(rand(1,2)*5);
    n=ceil(rand()*5);
    cNum(a(1))=cNum(a(1))+n;
    cNum(a(2))=cNum(a(2))-n;
    hn=round(tNum./(cNum));
    for cn=1:length(cNum)
        h{cn}=hn(cn)*ones(1,cNum(cn));
    end
    E=0;
    for i=1:length(cNum)
        ht=0;
        for j=1:i-1
            ht=ht+sum(h{j});
        end
        e=GetE(0,h{i},nn(i));
        E=E+e;
    end
    wdatas=[wdatas,E];
    LN=[LN;cNum];
end
LN=[LN,wdatas'];
xlswrite('¹°ÄÖ³µËý.xls',LN);

```

```

disp(['iDòÔËÐÐÊ±¼ä',num2str(toc),'s']);
disp(['ÔÛÉúÄÜÁç',num2str(sum(E)),'kJ']);
disp(['Ê¼ÊÄÛ°Ä',num2str(WE),'kJ']);

```

第三题第一问主要程序

```

function Q3_1()
clc;close all;
NUM=3;%Ö,¶ÑÓ³Û³μ±à°Ä¼°Ô³¼ì°Ä
SNUM=4;
h=[645 660 645 660 615];%·ç³μ¼ä,ô
global T;
E=0;
tic;
sis=[22903 21569 20283 18197 15932 13594 12240 10960 9422 8429 6447
4081 2806];
ses=[21569 20283 18197 15932 13594 12240 10960 9422 8429 6447 4081
2806 175];

v0=0;close all;
vmax=80;g=9.8;
T=1;
C=1e-3*3600;%m/s×ª»»Îªkm/hμÄÏμÊý
ds=sis-ses;
Vm=zeros(1,length(ds));
Vn=zeros(1,length(ds));
U1=zeros(1,length(ds));
U2=zeros(1,length(ds));
eGenerate=zeros(1,length(ds));
wdata=zeros(1,length(ds));
td=zeros(1,length(ds));
swdata=[];%¼ÇÄ¼ÖÄ»μÄ¹°Ä±ä»Çéçö
m=194.295;
TData=[];
VDatas=[];
SDatas=[];
WDatas=[];
tDatas=[];
tMin=inf;
VM=load('Vm.mat');
VN=load('Vn.mat');
VM=VM.Vm;
VN=VN.Vn;
UU1=load('UU1.mat');
UU2=load('UU2.mat');
tRun=load('×iÓÄÔËÐÐÊ±¼ä.mat');
tRun=tRun.tmean;
UU1=UU1.U1;
UU2=UU2.U2;
st=0;SMAX1=0;
VLdata=[];
WT=inf;
U1=zeros(1,2);U2=zeros(1,2);S1=zeros(1,2);S2=zeros(1,2);S3=zeros(1,2);S4=zeros(1,2);
%¼ÆËä³ôÊ¼½á¹û
vvm=[72,64];
vvn=[35,38];
for ti=0:1
    ws=inf;
    for u1=0.9:0.1:0.9
        for u2=0.9:0.1:1
            for vm=vvm(ti+1):vvm(ti+1)

```

```

[sData1,tData1,vData1,aData1,WData1,vlData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,ses(ti+SNUM));
W1=max(WData1);
if(exitflag1<0)
    continue;
end
for vn=vvn(ti+1):vvn(ti+1)

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest(vn,u2,ses(ti+SNUM));
if(exitflag4<0)
    continue;
end

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,ses(ti+SNUM)+max(sData4));
if(exitflag3<0)
    continue;
end
L=ses(ti+SNUM)-ses(ti+SNUM);

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1);
if(t2<0)
    continue;
end
t=max(tData1)+t2+max(tData3)+max(tData4);
s2=t2*max(vData1)/C;
vData2=max(vData1)*ones(1,floor(t2/T));
aData2=zeros(1,floor(t2/T)-1);
tData2=T:T:t2;
sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,ses(ti+SNUM)-max(sData1));
if(exitflag2<0)
    continue;
end
W2=max(WData2);
W=W1+W2;
ddt=abs(t-tRun(ti+SNUM));
if(ddt<=T)
    if(ws>W)
        U1(ti+1)=u1;
        U2(ti+1)=u2;
        Vm(ti+1)=vm;
        Vn(ti+1)=vn;
        ws=W;
        S1(ti+1)=max(sData1);
        S2(ti+1)=s2;
        S3(ti+1)=max(sData3);
        S4(ti+1)=max(sData4);

WData{ti+1}=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+max(WData2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
VData{ti+1}=[vData1,vData2,vData3,vData4];
vld{ti+1}=[vlData1,vlData2,vlData3,vlData4];
VLdata=[VLdata,vld];
aData{ti+1}=[aData1,aData2,-aData3,-aData4];

```



```

        if isempty(tData2)
            tmax=max(tData4)+max(tData1)+max(tData3);

tData{ti+1}=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData{ti+1}=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
        else
            tmax=max(tData4)+max(tData1)+max(tData3)+max(tData2);

tData{ti+1}=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)
+max(tData3)];

sData{ti+1}=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sData
a2)+max(sData3)];

        end
        if(ti>0)
            sData{ti+1}=sData{ti+1}+max(sData{ti+1-1});
            WData{ti+1}=WData{ti+1}+max(WData{ti+1-1});
        end

        wdata(ti+1)=ws;
        st=st+tmax;
        %±£'æ,Ä¶ÎµÄÖÆ¶Ê±¼ä¶Î½Úµã/ÔÛÉúÄÜÁ|ÒÔ¼°¹ºÄÊ±¼ä
        if(ti>0)
            tData{ti+1}=tData{ti+1}+max(tData{ti+1-1});
            tDrag{ti+1}=[tmax-max(tData4)+st,tmax+st];
            tLoss{ti+1}=[min(tData1)+st,max(tData1)+max(tData2)+st];
        else
            tDrag{ti+1}=[tmax-max(tData4),tmax];
            tLoss{ti+1}=[min(tData1),max(tData1)+max(tData2)];
        end
        end

[vl,r,k]=getRoadCon(ses(ti+SNUM)+max(sData4));%»ñÈ;ÖÆ¶ÎÎµÄÆÄ¶È

Emesh=0.5*m*min(vData3)^2/3.6^2-m*g*max(sData4)*sin(pi*k/180);%¼ÆËäÖÆ¶ÎÎÎÄ³µ»úÐµÄÜµÄ¼øÐ
;ÁÇ£¬×¢Òà·ù°Á

        eGenerate(ti+1)=(Emesh-Ef)*0.95;

        end
    end
end
end
end
end
end
end
w=sum(wdata);
if(WT>w)
    WT=w;
    vdata1=[VData{1},VData{2}];
    sdata1=[sData{1},sData{2}];
    tdata1=[tData{1},tData{2}];
    wdata1=[WData{1},WData{2}];

disp(['vm=',num2str(Vm(1)),'vn=',num2str(Vn(1)),'s1=',num2str(S1(1)),'s2=',num2str(S2(1)),'s3=',num2str(S3
(1)),'s4=',num2str(S4(1)),'u1=',num2str(U1(1)),'u2=',num2str(U2(1))]);

disp(['vm=',num2str(Vm(2)),'vn=',num2str(Vn(2)),'s1=',num2str(S1(2)),'s2=',num2str(S2(2)),'s3=',num2str(S3
(2)),'s4=',num2str(S4(2)),'u1=',num2str(U1(2)),'u2=',num2str(U2(2))]);

    end
    disp(['¹³ÐòÔËÐÐÊ±¼ä',num2str(toc),'s']);
    disp(['Ö³£ÐÐÊ»¹ºÄ',num2str(WT),'kJ']);

```

```

%¼ÆËäÖÅ»½á¹û
vvm=[79,69];
vvn=[38,38];
WT=inf
for dt=7:1:7
    dt
    tic;
    tti=[-dt -10+dt];
    for ti=0:1
        ws=inf;
        for u1=0.9:0.1:0.9
            for u2=1:0.1:1
                %for vm=vvm(ti+1):vvm(ti+1)
                for vm=45:vmax

[sData1,tData1,vData1,aData1,WData1,vlData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,ses(ti+SNUM));%»ñµÃÇ£Öý¶ÎµÄÎ»ÒÆ/Ê±¼ä/ËÜ¶ËËÿ¼Ý
                W1=max(WData1);
                if(exitflag1<0)
                    continue;
                end
                for vn=20:vm
                    %
                    for vn=vvn(ti+1):vvn(ti+1)

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest(vn,u2,ses(ti+SNUM)); %»ñµÃÖÆ¶ÎÎµÄÎ»ÒÆ/Ê±¼ä/ËÜ¶ËËÿ¼Ý
                    if(exitflag4<0)
                        continue;
                    end

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,ses(ti+SNUM))+max(sData4));
                    if(exitflag3<0)
                        continue;
                    end
                    L=sis(ti+SNUM)-ses(ti+SNUM);

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñµÃÑ²º½¶ÎµÄÎ»ÒÆ/Ê±¼ä/ËÜ¶ËËÿ¼Ý
                    if(t2<0)
                        continue;
                    end
                    t=max(tData1)+t2+max(tData3)+max(tData4);
                    s2=t2*max(vData1)/C;
                    vData2=max(vData1)*ones(1,floor(t2/T));
                    aData2=zeros(1,floor(t2/T)-1);
                    tData2=T:T:t2;
                    sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2,ses(ti+SNUM)-max(sData1));%»ñÈ;Ñ²º½¶ÎËÜ×öµÄ¹
                    if(exitflag2<0)
                        continue;
                    end
                    W2=max(WData2);
                    W=W1+W2;
                    ddt=abs(t-tRun(ti+SNUM)-tti(ti+1));
                    if(ddt<=T)
                        if(ws>W)
                            ws=W;
                            U1(ti+1)=u1;

```



```

w=sum(wdata);
if(WT>w)
    WT=w;
    TT=tti;
    vdata2=[VData{1},VData{2}];
    sdata2=[sData{1},sData{2}];
    SMAX1=max(sData{1});
    tdata2=[tData{1},tData{2}];
    wdata2=[WData{1},WData{2}];

disp(['vm=',num2str(Vm(1)),'vn=',num2str(Vn(1)),'s1=',num2str(S1(1)),'s2=',num2str(S2(1)),'s3=',num2str(S3(1)),'s4=',num2str(S4(1)),'u1=',num2str(U1(1)),'u2=',num2str(U2(1))]);

disp(['vm=',num2str(Vm(2)),'vn=',num2str(Vn(2)),'s1=',num2str(S1(2)),'s2=',num2str(S2(2)),'s3=',num2str(S3(2)),'s4=',num2str(S4(2)),'u1=',num2str(U1(2)),'u2=',num2str(U2(2))]);
end
disp(['µ¥'îĐòÔĖĐĖ±¼ă',num2str(toc),'s']);
end
disp(['Ė«¶ĭÑÓÊ±ĐĖÊ¹ºĂ',num2str(WT),'kJ']);

disp(['³]ĐòÔĖĐĖ±¼ă',num2str(toc),'s']);
%¼ĖĖâµ¥¶ĭÓĂ»¹½ă¹û
WT=inf
for dt=10:1:10
    dt
    tic;
    tti=[-dt -10+dt];
    for ti=0:1
        ws=inf;
        for u1=0.9:0.1:0.9
            for u2=0.9:0.1:1
                %for vm=vvm(ti+1):vvm(ti+1)
                for vm=45:vmax

[sData1,tData1,vData1,aData1,WData1,vlData1,pData1,fData1,kData1,exitflag1]=getAccst(v0,vm,u1,ses(ti+SNUM));%»ñµĂÇŁÔŸ¶ĭµĂĭ»ÔĖ/Ê±¼ă/ĖÛ¶ĖĖŸ³Ÿ
                W1=max(WData1);
                if(exitflag1<0)
                    continue;
                end
                for vn=20:vm
                    %
                    for vn=vvn(ti+1):vvn(ti+1)

[sData4,tData4,vData4,aData4,Ef,vlData4,kData4,exitflag4]=getReducest(vn,u2,ses(ti+SNUM)); %»ñµĂŒĖ¶ĭĭµĂĭ»ÔĖ/Ê±¼ă/ĖÛ¶ĖĖŸ³Ÿ
                    if(exitflag4<0)
                        continue;
                    end

[sData3,tData3,vData3,aData3,vlData3,kData3,exitflag3]=getBaseDrag_2(max(vData1),vn,ses(ti+SNUM))+max(sData4));
                    if(exitflag3<0)
                        continue;
                    end
                    L=ses(ti+SNUM)-ses(ti+SNUM);

```

```

t2=(L-max(sData1)-max(sData3)-max(sData4))*C/max(vData1); %»ñµÃÑ²º½¶îµÄî»ÒÆ/Ê±¼ä/ËÙ¶ÈÊý¾Ý
    if(t2<0)
        continue;
    end
    t=max(tData1)+t2+max(tData3)+max(tData4);
    s2=t2*max(vData1)/C;
    vData2=max(vData1)*ones(1,floor(t2/T));
    aData2=zeros(1,floor(t2/T)-1);
    tData2=T:T:t2;
    sData2=max(vData1)*tData2/C;

[WData2,vlData2,pData2,fData2,kData2,exitflag2]=getConstW(vData2, sis(ti+SNUM)-max(sData1)); %»ñÈ;Ñ²º
½¶îÊù×öµÄ¹
    if(exitflag2<0)
        continue;
    end
    W2=max(WData2);
    W=W1+W2;
    ddt=abs(t-tRun(ti+SNUM)-tti(ti+1));
    if(ddt<=T)
        if(ws>W)
            ws=W;
            U1(ti+1)=u1;
            U2(ti+1)=u2;
            Vm(ti+1)=vm;
            Vn(ti+1)=vn;
            ws=W;
            S1(ti+1)=max(sData1);
            S2(ti+1)=s2;
            S3(ti+1)=max(sData3);
            S4(ti+1)=max(sData4);

WData{ti+1}=[WData1,max(WData1)+WData2(2:length(WData2)),ones(1,length(sData3))*(max(WData1)+ma
x(WData2)),ones(1,length(sData4))*(max(WData1)+max(WData2))];
            VData{ti+1}=[vData1,vData2,vData3,vData4];
            vld{ti+1}=[vlData1,vlData2,vlData3,vlData4];
            % VLdata=[VLdata,vld];
            aData{ti+1}=[aData1,aData2,-aData3,-aData4];

            if isempty(tData2)
                tmax=max(tData4)+max(tData1)+max(tData3);

tData{ti+1}=[tData1,tData2,tData3+max(tData1),tData4+max(tData1)+max(tData3)];

sData{ti+1}=[sData1,sData2,sData3+max(sData1),sData4+max(sData1)+max(sData3)];
            else
                tmax=max(tData4)+max(tData1)+max(tData3)+max(tData2);

tData{ti+1}=[tData1,tData2+max(tData1),tData3+max(tData1)+max(tData2),tData4+max(tData1)+max(tData2)
+max(tData3)];

sData{ti+1}=[sData1,sData2+max(sData1),sData3+max(sData1)+max(sData2),sData4+max(sData1)+max(sDat
a2)+max(sData3)];

            end
            if(ti>0)
                sData{ti+1}=sData{ti+1}+max(sData{ti+1-1});
                WData{ti+1}=WData{ti+1}+max(WData{ti+1-1});
            end

```

```

wdata(ti+1)=ws;
st=st+tmax;
%±£'æ,Ä¶ÎµÄÖÆ¶Ê±¼ä¶Î½Úµã/ÔÛÉúÄÜÁ|ÒÔ¼°¹°ÄÊ±¼ä
if(ti>0)
    tData{ti+1}=tData{ti+1}+max(tData{ti+1-1});
    tDrag{ti+1}=[tmax-max(tData4)+st,tmax+st];
    tLoss{ti+1}=[min(tData1)+st,max(tData1)+max(tData2)+st];
else
    tDrag{ti+1}=[tmax-max(tData4),tmax];
    tLoss{ti+1}=[min(tData1),max(tData1)+max(tData2)];
end
end

[vl,r,k]=getRoadCon(ses(ti+SNUM)+max(sData4));%»ñÊ;ÖÆ¶¶ÎµÄÆÂÊÈ

Emesh=0.5*m*min(vData3)^2/3.6^2-m*g*max(sData4)*sin(pi*k/180);%¼ÆËÄÖÆ¶¶ÎÎÁÐ³µ»úÐµÄÜµÄ¼øÐ
;ÁÇ£¬×çÒâ·û°Á

eGenerate(ti+1)=(Emesh-Ef)*0.95;
end
end
end
end
end
end
w=sum(wdata);
if(WT>w)
    WT=w;

vdata3=[VData{1},VData{2}];
sdata3=[sData{1},sData{2}];
SMAx1=max(sData{1});
tdata3=[tData{1},tData{2}];
wdata3=[WData{1},WData{2}];

disp(['vm=',num2str(Vm(1)),',vn=',num2str(Vn(1)),',s1=',num2str(S1(1)),',s2=',num2str(S2(1)),',s3=',num2str(S3
(1)),',s4=',num2str(S4(1)),',u1=',num2str(U1(1)),',u2=',num2str(U2(1))]);

disp(['vm=',num2str(Vm(2)),',vn=',num2str(Vn(2)),',s1=',num2str(S1(2)),',s2=',num2str(S2(2)),',s3=',num2str(S3
(2)),',s4=',num2str(S4(2)),',u1=',num2str(U1(2)),',u2=',num2str(U2(2))]);
end
disp(['µ¥Î³ÎÐôÔËÐÐÊ±¼ä',num2str(toc),'s']);
end
disp(['µ¥¶¶ÎÎÓÊ±ÐÐÊ»¹°Ä',num2str(WT),'kJ'];
tRun(SNUM:SNUM+1)=tRun(SNUM:SNUM+1)+TT;
figure
plot(sdata1,vdata1);grid on;xlabel('¼àÄË/m'),ylabel('ËÜ¶Ê/km/h');hold on;plot(sdata2,vdata2,'r');grid
on;xlabel('¼àÄË/m'),ylabel('ËÜ¶Ê/km/h');plot(sdata3,vdata3,'y');grid
on;xlabel('¼àÄË/m'),ylabel('ËÜ¶Ê/km/h');hold off;
legend('Ö³£ÐÐÊ»Â·Î³','Ë«¶Î»Ö','ÎÓÊ±','µ¥¶¶Î»Ö','ÎÓÊ±');
% figure
% plot(sDatas,VDatas);grid on;xlabel('¼àÄË/m'),ylabel('ËÜ¶Ê/km/h');

save('ÔËÐÐÊ±¼ä.mat','tRun');
disp(['³ÎÐôÔËÐÐÊ±¼ä',num2str(toc),'s']);
disp(['×ÜÄÜ°Ä',num2str(sum(wdata)),',kJ']);

```

