

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 面向节能的单/多列车优化决策问题

摘 要：

本文建立了单列车的节能操控策略分析模型、单列车的节能优化最优控制模型、单列车的速度—路程计算模型、多列车的发车间隔优化分析模型、列车延误恢复优化控制模型。在上述模型基础上，对单列车的操控策略以及多列车的发车间隔进行了全面分析和优化。在模型求解方法上面，引入了贪心算法，二分法以及分治算法，提高了模型计算效率，有利于模型的推广。

针对问题一，提炼出了典型的最优控制问题模型。并利用 $E-T$ 曲线的单值对应关系，创新性地提出将时间约束问题转化为能量约束问题。

通过对哈密尔顿函数的分析，总结出单列车单路段最节能的五种操控策略。在模型求解方面，采用二分法对最小能量点进行求解。给出了单列车的速度—路程曲线以及各个时刻的速度、牵引力、消耗功率等数据结果。

对于单列车两路段的模型，是在单路段优化模型基础之上，进行两路段的时间分配再优化。对于该模型的求解，从 $E-T$ 函数曲线的性质进行求解，引入了贪心算法，不断迭代进行求解。给出了两路段的速度—距离计算曲线，以及各时刻速度、牵引力、消耗功率等数据结果。

针对问题二，将整个复杂的多变量约束优化问题分步优化。

第一步借鉴了第一题的求解思想，将单列车在全程路段中进行能耗的优化，使之在不考虑能量回收时，达到最小能耗的节能要求。

第二步是将多列列车的发车间隔作为优化控制变量。先对两相邻列车模型进行了优化分析，得出在两车间隔为 241s 秒为最优发车间隔；再对三相邻列车模型进行了优化分析，得出在题给约束条件下，三车分别相隔 660s 和 658s 时，三车的总能耗达到最低的结论。基于对两车优化和三车优化的分析，结合分治算法，提出了多列车的有约束能耗优化模型，该模型将复杂的多车优化分组优化，通过两车优化或者三车优化使问题得到简化，有利于求解。并且在模型的改进中，提出了将分组优化后的相邻车辆进再次等值成单列列车，反复使用两车优化和三车优化模型，最终得出全部列车的优化结果。

针对问题三，提出了一种后车“延赶结合”的列车延误调整控制策略。按照该策略控制列车，可以阻止延误的进一步传播，使列车延误范围最小化；并能使后车经过一站的时间，达到恢复正点运行的目的。并基于这种调整策略，建立以能量消耗最小为目标函数的列车延误优化控制模型。根据运行条件不同给出了两种求解算法。考虑到列车发生较大延误的情况，进一步将模型推广到多列车的优化控制，给出了一种考虑多列车的列车延误优化控制模型，并提出利用遗传算法求解该模型。

关键词： 单列车的节能优化最优控制模型 单列车的速度—路程计算模型 多列车的发车间隔优化分析模型 列车延误恢复优化控制模型

一、 问题重述

1.1 问题背景

轨道交通具有区间距离短、启动和制动频繁等特点，带来严重的能量消耗问题。在低碳环保、节能减排日益受到关注的情况下，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。因此从能耗的角度，研究列车运行操纵控制与组织调度的优化模型和算法，建立不同的描述列车运行控制的优化模型及算法，不仅在理论上丰富和拓展优化理论在列车运行控制中的应用，更可以在实践上加强对铁路运输系统轨道交通流的控制和管理，减少铁路运输企业的能耗成本支出，同时有助于建设资源节约型社会，促进铁路运输企业履行社会责任、落实可持续发展的理念。从而尽可能发挥轨道交通系统的最大效率和效益，为轨道交通系统在新形势下的发展提供理论基础和技术支持^[1]。

1.2 问题描述

本文分别针对单辆列车及多辆列车提出节能优化方法。针对单列列车控制，主要通过操纵控制，改变列车牵引、巡航、惰行和制动的状态在保证列车准点的前提下实现节能运行。对于多辆列车控制，除了通过操纵控制，还可以通过改变不同列车的发车间隔及停车时间等调度管理的方法，提高再生能量利用效率，实现节能运行。最后，针对可能出现的延误等极端现象提出相应的优化控制计算模型。

1.3 本文所需解决的问题

一、单列车节能运行优化控制问题

1) 建立计算速度距离曲线的数学模型，计算寻找一条列车从 A_6 站出发到达 A_7 站的最节能运行的速度距离曲线，其中两车站间的运行时间为 110 秒。

2) 建立计算速度距离曲线的数学模型，计算寻找一条列车从 A_6 站出发到达 A_8 站的最节能运行的速度距离曲线，其中要求列车在 A_7 车站停站 45 秒， A_6 站和 A_8 站间总运行时间规定为 220 秒。

二、多列车节能运行优化控制问题

1) 当 100 列列车以间隔 $H=\{h_1, \dots, h_{99}\}$ 从 A_1 站出发，追踪运行，依次经过 A_2, A_3, \dots 到达 A_{14} 站，中间在各个车站停站最少 D_{\min} 秒，最多 D_{\max} 秒。间隔 H 各分量的变化范围是 H_{\min} 秒至 H_{\max} 秒。建立优化模型并寻找使所有列车运行总能耗最低的间隔 H 。要求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为 $T_0=63900$ 秒，且从 A_1 站到 A_{14} 站的总运行时间不变，均为 2086 秒。

2) 接上问，如果高峰时间（早高峰 7200 秒至 12600 秒，晚高峰 43200 至 50400 秒）发车间隔不大于 2.5 分钟且不小于 2 分钟，其余时间发车间隔不小于 5 分钟，每天 240 列。制定运行图和相应的速度距离曲线。

三、列车延误后运行优化控制问题

接上问，若列车 i 在车站 A_j 延误 DT_j^i 发车，建立控制模型，找出在确保安全的前提下，首先使所有后续列车尽快恢复正点运行，其次恢复期间耗能最少的列车运行曲线。

二、模型的假设与符号说明

2.1 模型假设

假设一：忽略列车长度，用单质点模型表示列车；
假设二：不考虑列车的乘客及上座率，质量为列车车体质量；
假设三：列车的牵引力、制动力可以连续调节；
假设四：列车运行过程中的动能只考虑线性动能，而不考虑列车转弯过程中的转动动能；
假设五：在分组优化中，认为各单元的控制策略相同。

2.2 符号说明

表 2-1 符号说明

参数符号	符号说明
F_{\max}	牵引力的最大值（kN）
μ_f	实际输出的牵引力和最大牵引力的比值
w_0	单位基本阻力（N/kN）
w_1	单位附加阻力（N/kN）
i	线路坡度
R	轨道曲率半径（m）
B_{\max}	最大制动力（kN）
μ_b	实际输出的制动力与最大值动力的比值
M	列车质量（kg）
V_{\max}	线路限速（km/h）

E	列车的耗能 (kJ)
T	列车在区间的运行时间 (s)
D_{\max} 、 D_{\min}	列车的最大、最小停站时间 (s)
H_{\max} 、 H_{\min}	列车的最大、最小运行时间 (s)
L	列车运行总路程 (m)
h	两辆列车间的出发间隔 (s)
DT_j^i	列车 i 在车站 j 的延误时间 (s)
s	距离 (m)

三、 问题分析

3.1 针对问题一

问题一是针对单列列车的以耗能最小为目标的控制策略最优问题,属于典型优化控制问题。状态量为列车位置和速度,控制变量为列车的牵引力和制动力。初状态和末状态均已知,在约束条件下,选择合适的控制变量取值,使得系统目标函数达到最小。

对于该类问题的建模和求解方法常分为数值方法和随机方法。就本题来说,控制变量较少,数值方法在可行的基础上效率更高,所以本文将用数值方法进行求解。

3.2 针对问题二

问题二需考虑控制变量较多,包括每列车的发车间隔,每列车的停站时间以及每列车的控制策略,并且相关控制变量均有自身的约束条件,分析较为复杂,大规模的随机算法在计算效率上难以满足需要。在对复杂模型进行部分简化的基础上,再借助第一问的优化结论,可以将该复杂问题转化为一个两步优化的问题,先从能量角度对单列车进行优化,再将列车之间的发车间隔进行优化,具体描述如下:

第一步:关于单列车的控制方式,先不考虑能量回收的问题,在 A1-A14 站之间找到最优的运行方式。使得列车从电网中获取能量最小。

第二步:在单列车全程优化的基础之上,对多列列车的发车间隔 H 进行优化,使得从电网所需总能耗最小。

3.3 针对问题三

问题三是列车延误情况下恢复运行的控制策略问题，涉及恢复运行的安全距离、恢复调整时间以及恢复过程能耗问题。涉及变量较多，与前两题相比复杂程度进一步提升。基于此，本文提出了一种基于追赶结合策略的优化控制模型，并根据运行条件不同给出了两种求解算法。考虑到列车发生较大延误的情况，进一步将模型推广到多列车的优化控制，给出了一种考虑多列车的列车延误优化控制模型，并提出利用遗传算法求解该模型。

四、 问题一求解

4.1 数学模型

4.1.1 单路段的最优控制模型

在列车运行时间和距离确定的条件下，列车可以有多种操纵策略。问题一的核心在于寻一种操纵策略最小化列车在运行过程中的牵引耗能。列车正常运行状态时的动力学方程为：

$$\frac{dv(x)}{dt} = F(x) - B(x) - W(x) \quad (4-1)$$

式中, v 为列车运行速度； t 为列车运行时间； x 为列车运行位置； F 为列车施加的牵引力。 B 为列车施加的制动力； W 表示列车的总阻力，由基本阻力和附加阻力组成。由题目已知条件，牵引力 F 可以由式（4—2）给出，制动力 B 可以由式（4—3）得到，总阻力 W 由（4—4）给出：

$$F = \mu_f F_{\max} \quad (4-2)$$

$$B = \mu_b B_{\max} \quad (4-3)$$

$$W = (w_0 + w_1) \times g \times \frac{M}{1000} \quad (4-4)$$

其中，

$$F_{\max} = \begin{cases} 203 & 0 \leq v \leq 51.5 \text{ km/h} \\ -0.002032v^3 + 0.4928v^2 - 42.12v + 1343 & 51.5 < v \leq 80 \text{ km/h} \end{cases} \quad (4-5)$$

$$B_{\max} = \begin{cases} 166 & 0 \leq v \leq 77 \text{ km/h} \\ 0.1343v^2 - 25.07v + 1300 & 77 < v \leq 80 \text{ km/h} \end{cases} \quad (4-6)$$

$$\begin{cases} w_0 = A + Bv + cv^2 \\ w_1 = i + \frac{c}{R} \end{cases} \quad (4-7)$$

式中, μ_f 为实际输出的牵引力和最大牵引力的比值； μ_b 为实际输出的制动力与最大值动力的比值； w_0 为基本阻力； F_{\max} 为最大牵引力； B_{\max} 为最大制动力； A 、 B 、 C 为阻力

多项式系数，一般取经验值； w_1 为附加阻力； i 为线路坡度； c 为综合反映影响曲线阻力许多因素的经验系数，一般取 600； R 为曲率半径。

综上，以最节能运行作为优化目标，满足列车运行所需的时间约束、最大加速度约束、最大减速度约束、限速约束的数学模型为：

$$\left\{ \begin{array}{l} \min \quad E = \int_0^L F(x) dx \\ \text{s.t.} \quad \frac{dv(x)}{dt} = \mu_f(x) F_{\max}(v) - \mu_b(x) B_{\max}(v) - W(x, v) \\ \int_0^L \frac{1}{v(x)} dx = T \\ -1 < \frac{dv(x)}{dt} \leq 1 \\ v(0) = v(T) = 0, v(x) \leq V_{\max}(x) \\ 0 \leq \mu_f(x) \leq 1, 0 \leq \mu_b(x) \leq 1 \end{array} \right. \quad (4-8)$$

式中, E 为列车运行的牵引耗能； L 为列车的运行距离； T 为列车的运行总时间； $V_{\max}(x)$ 为列车在 x 处的限速。

4.1.2 两路段的最优控制模型

根据上述模型可以得出单路段的最优操控，问题一第二问要求在两路段进行优化。在上一问的基础上，对于两路段的优化提出如下数学模型：

$$\left\{ \begin{array}{l} \min \quad E = E_1 + E_2 \\ \text{s.t.} \quad E_1 = \int_0^{L_1} F(x) dx, E_2 = \int_0^{L_2} F(x) dx \\ \frac{dv(x)}{dt} = \mu_f(x) F_{\max}(v) - \mu_b(x) B_{\max}(v) - W(x, v) \\ \int_0^{L_1} \frac{1}{v(x)} dx + \int_0^{L_2} \frac{1}{v(x)} dx = T \\ -1 < \frac{dv(x)}{dt} \leq 1 \\ v(0) = v(T) = 0, v(x) \leq V_{\max}(x) \\ 0 \leq \mu_f(x) \leq 1, 0 \leq \mu_b(x) \leq 1 \end{array} \right. \quad (4-9)$$

式中, E_1 和 E_2 分别为路段 1 和路段 2 的耗能； L_1 和 L_2 为两路段路程； T 为两运行总用时（不包括停车时间）；其余变量含义不变。

4.2 模型求解方法

4.2.1 列车运行状态之间的转换分析

列车运行过程中，一般会有牵引、巡航、惰行、制动四种运行状态。在列车运行过程中，由于受到线路的限速等限制，这四种状态之间发生相互转化（如图 4.1 所示）。例如，当列车启动后加速运行到最高限速时，列车需要由牵引状态转化为巡航状态。虽然列车运行过程中会发生四种状态之间的相互转化，但是这种无序操纵序列转化显然不是最优的，频繁的加减速必然会使得能量消耗较大^[2]。

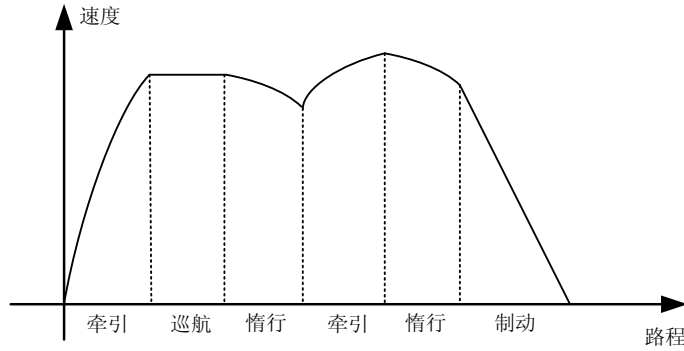


图 4.1 列车的操纵序列示意图

由庞德利亚金最大原则^[1]可知，最小化列车牵引能耗等价于最大化汉密尔顿函数 H ：

$$H = p_1[F(x) - B(x) - W(x)] - \frac{p_2}{v(x)} - F(x) \quad (4-10)$$

根据式 (4—10)，分析列车节能运行策略分为 5 种情况：

- 1) $p_1 > 1$ 时， $F = F_{\max}$ ， $B = 0$ ，最大加速；
- 2) $p_1 = 1$ 时， F 可变， $B = 0$ ，巡航；
- 3) $0 < p_1 < 1$ 时， $F = 0$ ， $B = 0$ ，惰行；
- 4) $p_1 = 0$ 时， $F = 0$ ， B 可变，巡航；
- 5) $p_1 < 0$ 时， $F = 0$ ， $B = B_{\max}$ ，最大制动。

对于坡度不变的运行区间，其最优驾驶策略是由最大加速、巡航、惰行和最大制动组成的。文献^[4]从理论上证明了在坡度变化很小的线路，也存在的最优的操纵序列为“最大牵引、巡航、惰行、最大制动”，列车仅在牵引和巡航状态下消耗能量，其余状态不消耗能量，并且巡航仅出现在速度达到上限的情况。

在以上分析的基础之上，可以看出，对于一段存在恒定限速和至多包括一个起伏型坡道的简单线路，根据线路的限速和线路长度以及运行时间以及坡度的不同，列车操纵序列会存在临界状态，不一定包括所有的四种状态，有可能会缺失惰行、巡航以及制动其中的一种或几种^{[5][6]}。但可以肯定的是，这四种状态的转换方向不能改变（如图 4.2），综上对于一个简单路段的分析，可以给出最节能的五种操控策略，即：

- 策略 1：最大牵引—最大制动；
- 策略 2：最大牵引—巡航—最大制动；
- 策略 3：最大牵引—惰行—最大制动；
- 策略 4：最大牵引—巡航—惰行—最大制动；
- 策略 5：最大牵引—惰行。

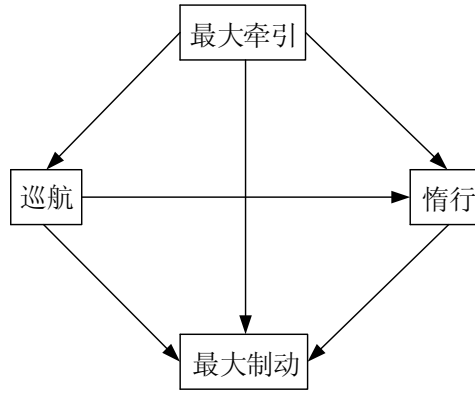


图 4.2 列车最优操纵状态转化

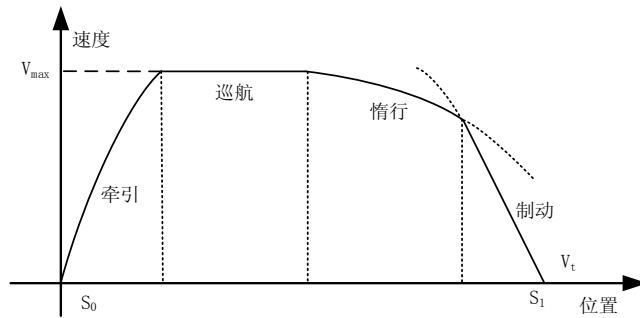


图 4.3 列车最优操纵模式

至于最终列车采用何种最节能操控策略（必为五种之一），这需要根据具体的路况以及给定的时间约束直接计算得出。从能量角度来讲，根据题中所给 E-T 函数曲线，运行时间越短，所需要的能耗越大。并且运行时间与最小能耗是一个单值函数关系，即根据时间可以确定最小能耗，根据最小能耗亦可以确定运行时间。所以，对同一路段，上面五种操控策略一般满足如下关系：

$$E_{\text{策略1}}(E_{\text{策略2}}) > E_{\text{策略4}} > E_{\text{策略3}} > E_{\text{策略5}}$$

由此可以看出，操控策略与能量有直接对应关系。另外，根据分析可知，策略 1 能否存在取决于路况和速度上限。例如，达到速度上限时的剩余路段大于最大刹车距离时，就只会有序列 2 存在，不会存在序列 1。并且序列 5 在有速度约束时，不一定能存在。能量过大，后程无法完成刹车，能量过小，列车无法按照策略 5 完成运行。所以对应于时间，每个路段均有一个最小运行时间（最大能量）和最大运行时间（最小能量）。

4.2.2 单列车最优操控求解算法

问题一第一问为单列车节能运行优化控制问题。显然，如模型所述，列车存在运行时间和运行距离的双重约束，直接解析方法难以求解，考虑用数值解法进行求解。

根据前文对操控状态的分析，最小耗能和具有单值对应关系。显然，如果给定时间，难以直接确定其最小能量。但是如果给定能量，基于只有加速和巡航两种状态会消耗能量，加之速度约束，可以计算出两状态的时间，至于何时惰行转为制动，则需要求出剩余路段能够完成的最大刹车速度，即图 4.3 惰行和制动的交点。如果恰好制动线与巡航结束点相交，即是前述的策略 2，为无惰行的临界状态。

在求解时，将整个过程按距离划分为 N 等份，当 N 足够大（本文取 N=100000）时，则每份路程内可以看成匀速运动，在分段点处速度发生变化。并且，每段中牵引力，制

动力以及阻力保持不变。对于给定时间条件下的最小能量的求解，采用二分法，不断逼近求得。

综上，给定时间约束前提下，单列车的速度-路程图线的求解流程图 4.4 所示：

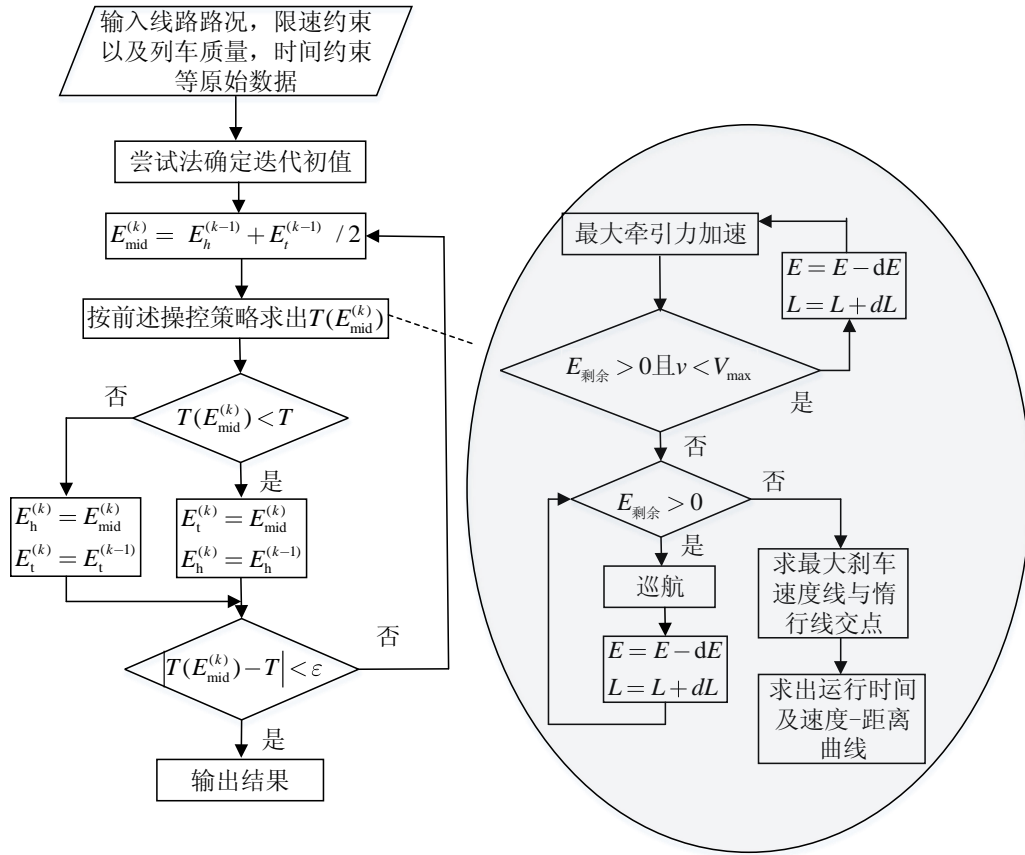


图 4.4 单列车最优操控求解算法流程图

4.2.3 单列车两路段能耗优化算法

在上一问基础之上，时间约束条件不增加，时间变量增加一维。对于该问题，借鉴第一问的方法枚举最小能量，求出两路段的 $E-T$ 曲线，然后将问题化为时间分配的优化问题。

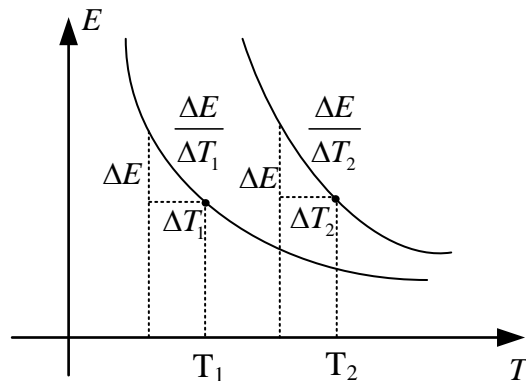


图 4.5 $E-T$ 曲线

如图 4.5 所示，本文提出一种基于贪心思想来分配两路段运行时间的方法。基本思想为：给定两路段初始运行时间和相应最小能量，使其大于运行时间上限；以一定能量步长增加能量，每步选择两段中对运行时间减少更有利（增加该能量后，运行时间减少

量更大) 的路段分配该能量, 直到总时间满足约束, 迭代结束。选择具体方法为:

Step1: 输入数据并完成初始化;

Step2: 分别求出两路段的 $E-T$ 曲线;

Step3: 给定两个路段的起始运行时间和对应最小能量;

Step4: 如果总时间不满足要求, 跳到第五步, 否则跳到第六步;

Step5: 能量增加 ΔE , 分别计算两路段对应的时间减少量, 比较得出 ΔE 应分配路段, 并修正变化后的能量以及运行时间, 跳到第四步;

Step6: 此时各路段对应的能量为列车最小耗能, 时间为最优分配。

4.3 结果及分析

4.3.1 单路段速度-距离曲线

根据上文所述算法, 通过 Matlab 编程求解, 具体结果如下。

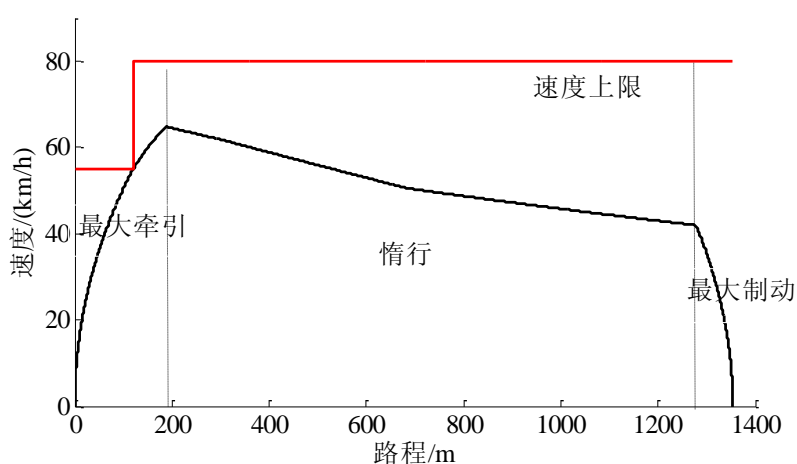


图 4.6 速度-距离曲线

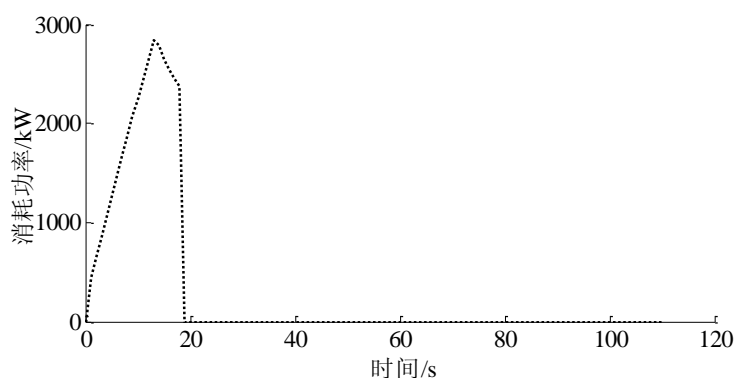


图 4.7 功率-时间曲线

如图 4.6 所示, 在该控制策略下, 列车能够以 110s 从 A6 站运行到 A7 站, 并且消耗能量达到最小值 34673kJ。

4.3.2 两路段速度—距离曲线

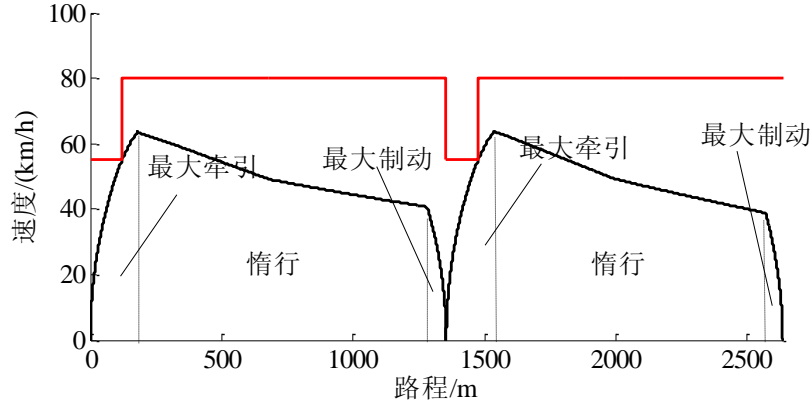


图 4.8 速度-距离曲线

如图 4.8 为两路段的最小能耗时的速度—距离曲线，在该操控策略下，A6-A7 运行时间为 112s，消耗能量为 33200kJ；A7-A8 运行时间为 108s，消耗能量为 33410kJ，总耗时 220s，消耗能量 66610kJ。具体的每时刻的速度，公里标，功率等见附录 C。

五、 问题二求解

5.1 数学模型

5.1.1 多列车能耗优化模型及求解算法

① 单列车的全程操控策略优化模型

如前文所述，任意相邻两站点之间路程中，均会存在一个特定的最小能耗 E 与运行时间的 T 的 $E-T$ 曲线。当列车的质量、速度约束、路程的长度以及各处路况给定时，该曲线必然是固定不变的，改变最小能耗，对应会给出一个运行时间。设想，当所有路段 $E-T$ 曲线已知时，若想最大程度减小能耗，则每段的运行时间即满足下式：

$$\begin{cases} E_1 = f_1(T_1) \\ E_2 = f_2(T_2) \\ \vdots \\ E_{13} = f_{13}(T_{13}) \\ T_1 + T_2 + \cdots + T_{13} + D_1 + D_2 + \cdots + D_{13} = 2086s \end{cases} \quad (5-1)$$

根据 $E-T$ 曲线性质可知，运行时间越少，需要能量越大。最大能量临界状态即是没有惰行的临界点，即时间最小点。所以这里出现第一个优化，即认为所有站的停车时间取最小值 30 秒。该优化问题的数学模型可改写为：

$$\begin{aligned}
& \min \sum_{i=1}^{13} E_i \\
& \text{s.t.} \quad \begin{cases} E_1 = E_1(T_1) \\ E_2 = E_2(T_2) \\ \vdots \\ E_{13} = E_{13}(T_{13}) \\ T_1 + T_2 + \cdots + T_{13} = 1726s \end{cases}
\end{aligned} \tag{5—2}$$

对该问题的求解,将第一题的方法进行推广,基本思想为:在不小于最小能量的前提下,给出较小的能量方案 $E=[E_1, E_2 \cdots E_{13}]$ 和相应的各段运行时间 $T=[T_1, T_2 \cdots T_{13}]$, 通过增加能量 ΔE 来不断减小运行时间,最终使得运行时间满足 1726s 的约束。求解该问题的关键有三点:一是初值给出合理;二是 $E-T$ 曲线足够精确;三是 ΔE 在 $E-T$ 曲线精确的基础上足够小。具体流程如下:

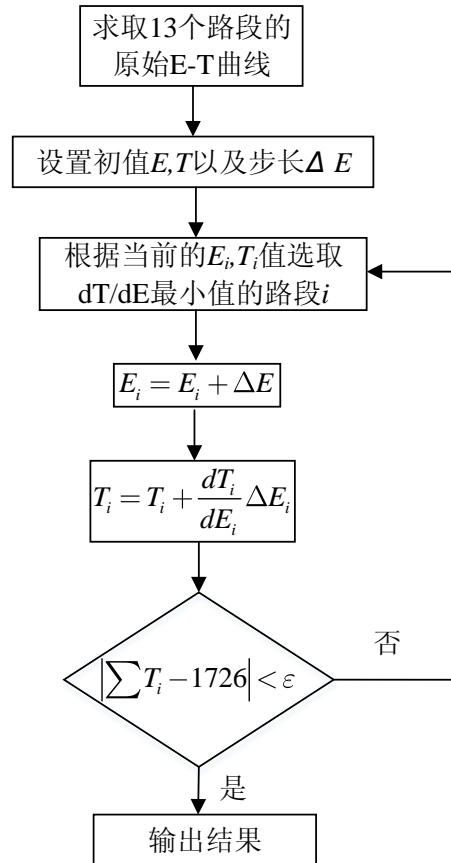


图 5.1 单列车的全程操控优化算法流程

需要注意的是,在每一步的迭代过程中,总是选择增加单位能量可以减少运行时间最多的路段,以求尽可能少的增加能量就可以满足时间的约束。这是推广算法的核心,也是第二个优化。

② 考虑能量回收的多车运行策略优化模型

由题可知,再生能量的计算公式为:

$$E_{\text{reg}} = (E_{\text{mech}} - E_f) \times 95\% \quad (5-3)$$

由于列车加速与其他列车刹车有时间重叠时，能量才可以回收利用，并且与重叠时间成正比，所以被利用的再生能量为：

$$E_{\text{used}} = E_{\text{reg}} \times t_{\text{overlap}} / t_{\text{brake}} \quad (5-4)$$

根据上面方法求得最佳运行结果 $E_{\text{min}} = [E_1, E_2 \cdots E_{13}]$ ，单列车的控制策略唯一确定，在该控制策略下，若不考虑能量回收，可以达到最小能耗。下面将基于相邻两列列车对发车间隔进行优化建模。

根据题目要求，相邻两辆车的间隔 $120s \leq h \leq 660s$ ，所以可以通过枚举法求得任意间隔下的总耗能。在进行耗能计算时，以每个单位时间内（1秒内，共 $2086+h$ 秒），记录两列车向电网吸取的净能量 $E_{\text{net}}(t)$ ，所以需将每个 1s 内的加速吸收能量 $E_{\text{in}}(t)$ 和刹车放出能量 $E_{\text{out}}(t)$ 分别记录，通过下式计算该 1s 时间内从电网吸收的净能量：

$$E_{\text{net}}(t) = \begin{cases} 0, & E_{\text{in}}(t) \leq E_{\text{out}}(t) \\ E_{\text{in}}(t) - E_{\text{out}}(t), & E_{\text{in}}(t) > E_{\text{out}}(t) \end{cases} \quad (5-5)$$

另外，根据题意，刹车放出的能量与两车加减速重叠时间成正比，所以整个刹车过程的产生的能量必然均匀分配在整个刹车时段内，并且这个能量是否能被吸收一是取决于是否与另一列车加速时间重合，二是该单位时间内刹车产生的能量是否能被全部吸收。具体算法流程如下：

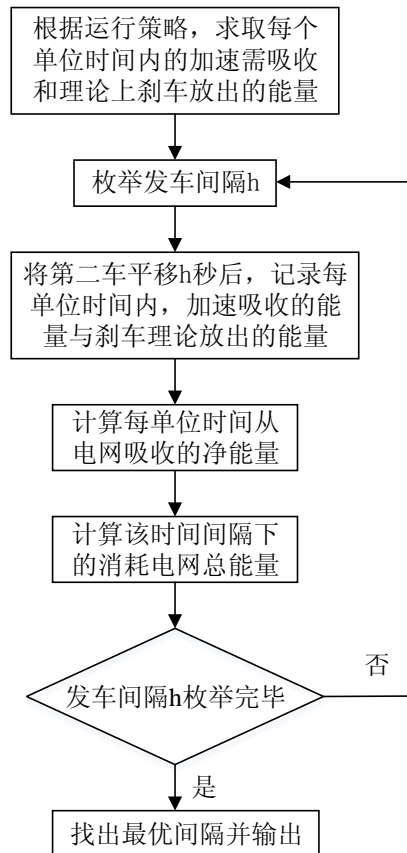


图 5.2 两车间隔优化算法流程

基于上述方法，同样可以完成对三相邻列车的两个间隔优化。但当列车数量增加以

及总时间有约束时，对于 100 列车的发车间隔优化无法再采用枚举方法。所以，这里考虑采用分治策略来进行分组优化。如图 5.3 所示，每 3 列车分为一组，100 列车共分 33 组，最后一列车单独分组。将问题进一步简化每一组的发车间隔策略相同（ h_1 和 h_2 ，黑色和蓝色），组间发车间隔（红色， h_3 ）相同。

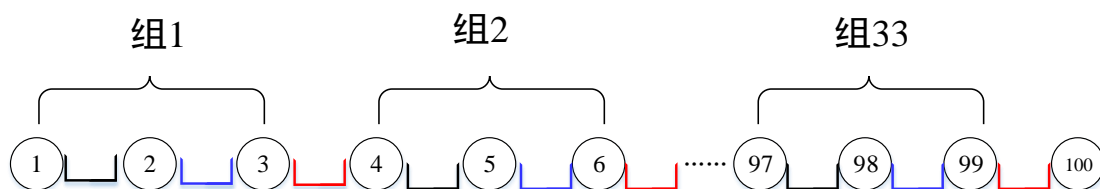


图 5.3 分治算法示意图

根据总体时间约束：

$$\begin{cases} 120s \leq h_3 = \frac{63900 - 33 \times (h_1 + h_2)}{33} \leq 660s \\ h_1 + h_2 \leq 2 \times 660s \end{cases} \quad (5-6)$$

可以解得

$$1277s \leq h_1 + h_2 \leq 1320s \quad (5-7)$$

所以，将该问题转化为一个三车优化问题，通过上文方法解出 h_1 、 h_2 和 h_3 ，完成对 100 列列车的发车间隔优化。

5.1.2 考虑高峰时间约束的列车间隔优化模型及求解算法

在上一问模型的基础之上，同样采用分治策略，进行分组优化。如图 5.4 所示，首先按照发车间隔约束不同分为 5 段，在每一段的约束条件下进行单段优化，在段间相邻处进行适当修正满足总体的时间约束。

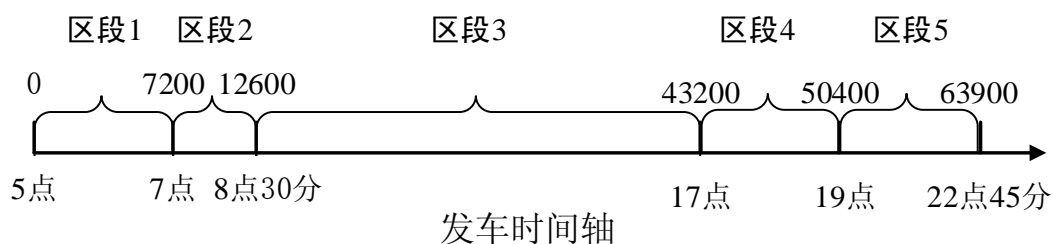


图 5.4 区段示意图

对于区段 1、区段 3 和区段 5，发车间隔约束为：

$$300s \leq h \leq 660s \quad (5-8)$$

对于区段 2 和区段 4，发车间隔约束为：

$$120s \leq h \leq 150s \quad (5-9)$$

在每一段的发车间隔优化过程中，利用上一问的方法转化为两车或者三车优化，这里以两车优化为例进行分组优化。根据约束条件得出每一段列车的数量范围为如下表 5.1 所示。该问题的数学模型即为 5.1.1 节中的数学模型与表 5.1 的数量约束的结合。

表 5.1 列车数量约束

区段	1	2	3	4	5
列车数量约束	[11,24]	[36,45]	[47,102]	[48,60]	[21,45]
总数量约束	240				

5.2 结果及分析

5.2.1 多列车能耗优化结果

首先，按一定步长，求解所有 13 条路段的 E-T 曲线，求解结果如下图所示：

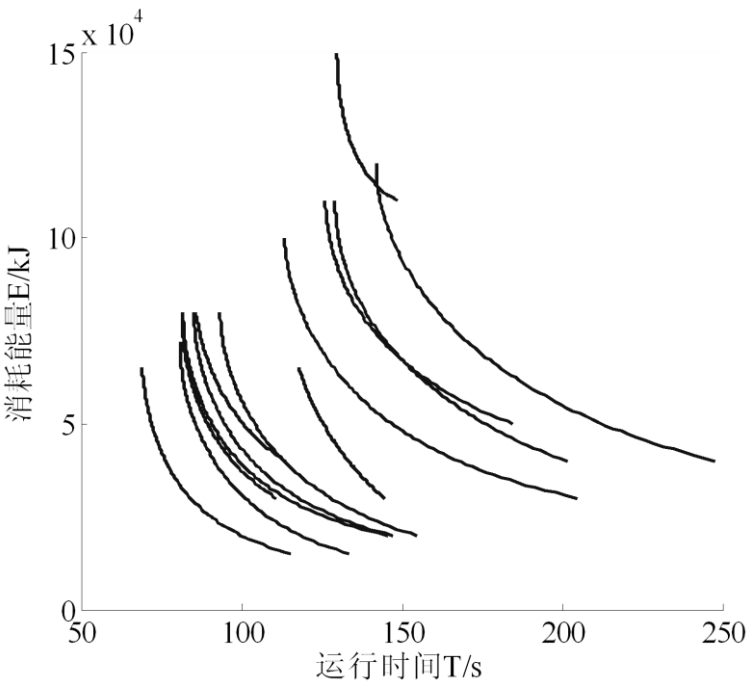


图 5.5 E-T 曲线

单列车在 A1-A14 时间分配为：

表 5.2 区段运行时间及耗能

车站区间	运行时间(s)	耗能 (kJ)
A1-A2	113	40400
A2-A3	109	31000
A3-A4	145	30000
A4-A5	160	60800
A5-A6	169	54000
A6-A7	114	32000
A7-A8	110	32000
A8-A9	125	31400
A9-A10	93	23500

A10-A11	149	48670
A11-A12	146	111200
A12-A13	109	24690
A13-A14	184	63200
A1-A14	1726	582860

单列车全程的速度—路程关系如图 5.6 所示：

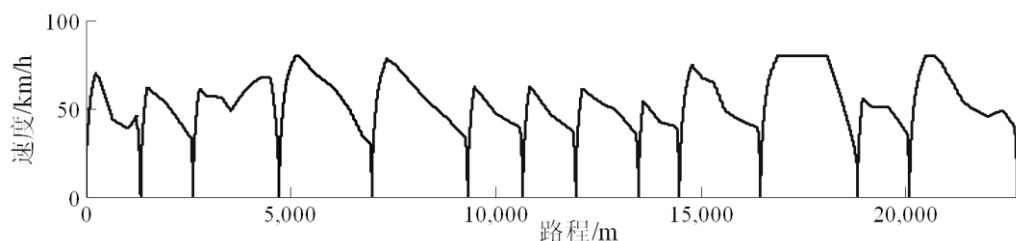


图 5.6 单列车全程速度—路程曲线

对于两车模型进行计算后，在约束范围内，发车时间间隔与两车总耗能的关系如图 5.7 所示：

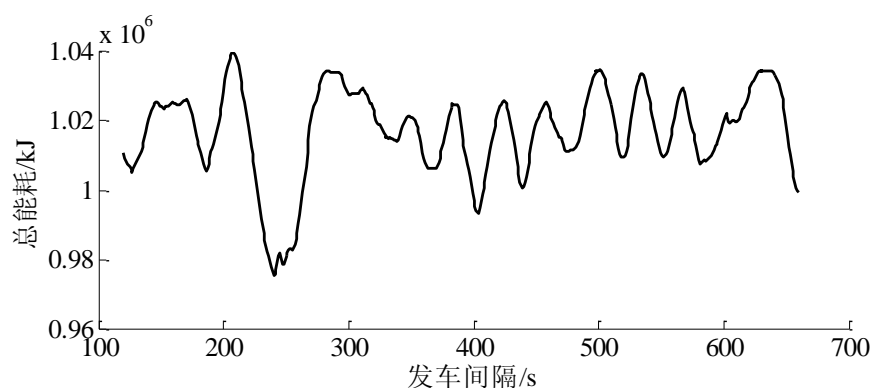


图 5.7 两列车发车间隔与总耗能关系曲线

对于三车模型进行计算后，在约束范围内，发车时间间隔与两车总耗能的关系如图 5.8 所示：

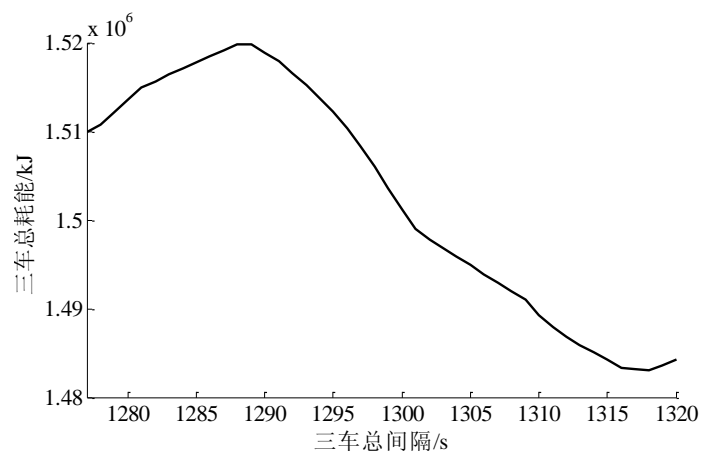


图 5.8 三列车总能耗与发车间隔关系

根据三车优化结果可知，当三车总间隔 $h_1 = 660s$ ， $h_2 = 658s$ ， $h_1 + h_2$ 取 1318s 时，单

组列车的能耗最小,继而算出 $h_3 = 618s$ 。经计算,100 辆列车全天的能耗为 $4.84 \times 10^7 kJ$ 。
100 列列车发车时刻表 5.3 所示:

表 5.3 100 列列车发车时刻表

列车编号	发车时刻	列车编号	发车时刻	列车编号	发车时刻	列车编号	发车时刻
1	0	26	16148	51	32294	76	48400
2	660	27	16806	52	32912	77	49060
3	1318	28	17424	53	33572	78	49718
4	1936	29	18084	54	34230	79	50336
5	2596	30	18742	55	34848	80	50996
6	3254	31	19360	56	35508	81	51654
7	3872	32	20020	57	36166	82	52272
8	4532	33	20678	58	36784	83	52932
9	5190	34	21296	59	37444	84	53590
10	5808	35	21956	60	38102	85	54208
11	6468	36	22614	61	38720	86	54868
12	7126	37	23232	62	39380	87	55526
13	7744	38	23892	63	40038	88	56144
14	8404	39	24550	64	40656	89	56804
15	9062	40	25168	65	41316	90	57462
16	9680	41	25828	66	41974	91	58080
17	10340	42	26486	67	42592	92	58740
18	10998	43	27104	68	43252	93	59398
19	11616	44	27764	69	43910	94	60016
20	12276	45	28422	70	44528	95	60676
21	12934	46	29040	71	45188	96	61334
22	13552	47	29700	72	45846	97	61952
23	14212	48	30358	73	46464	98	62612
24	14870	49	30976	74	47124	99	63270
25	15488	50	31636	75	47782	100	63900

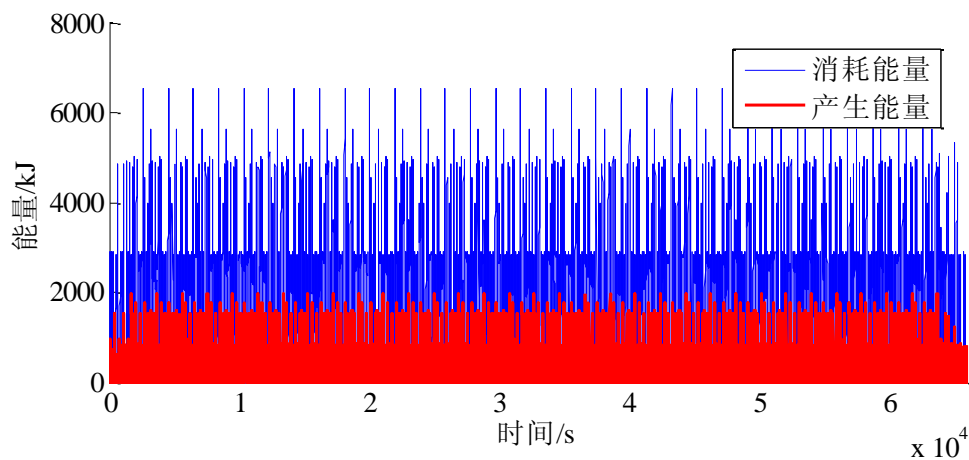


图 5.9 100 列列车每时刻的耗能与产能

5.2.2 考虑高峰时间发车约束的多列车能耗优化结果

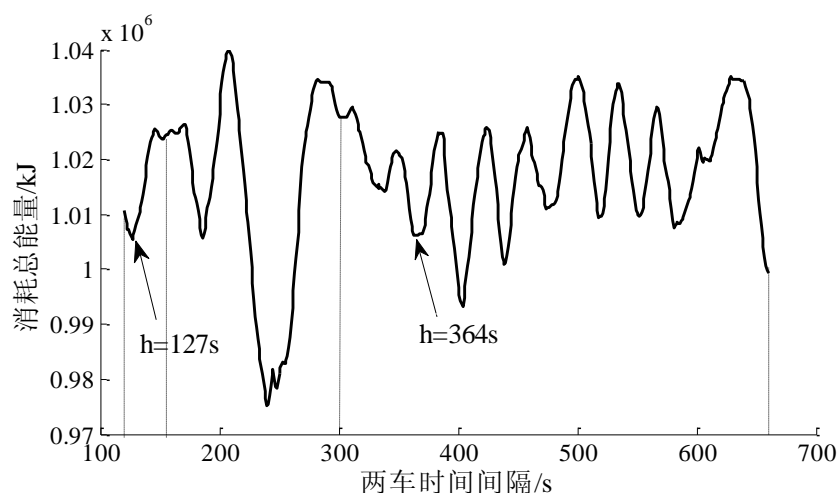


图 5.10 两车总能耗与发车间隔关系

首先，对区段 2 和区段 4 进行发车间隔优化（ $7200s \leq t < 12600s$ ， $43200s \leq t < 50400s$ ）。根据图 5.10 可知，在约束范围内（ $120s \leq h \leq 150s$ ）取 127s，两车总耗能最小。所以在区段 2 内按照 127s 间隔发车，共发 42 趟；在区段 4 内按 127s 的间隔共发 55 趟。

其次，对区段 1、区段 3 以及区段 5 进行优化（ $0 \leq t < 7200s$ ， $12600s \leq t < 43200s$ 以及 $43200s < t \leq 63900s$ ）。在区段 2 和区段 4 优化完成后，剩余可用间隔时间为：

$$t = 63900 - 127 \times 97 = 51581s \quad (5-10)$$

剩余 143 辆列车平均间隔约为 360s。由图 5.10 可知，在 360s 附近取得能耗最小值的发车间隔约为 364s，所以剩余车辆按 364 秒进行发车，在区段连接附近通过适当调整个别列车发车时间来满足总时间 63900s 的要求。240 辆列车的发车分布区段如下表 5.4 所示：

表 5.4 区段列车数

区段	1	2	3	4	5
列车数	20	42	84	55	39

由此可计算得，一天中，240 辆列车的发车时刻表 5.5 所示：

表 5.5 240 列列车发车时刻表

列车编号	发车时刻	列车编号	发车时刻	列车编号	发车时刻	列车编号	发车时刻
1	0	61	12360	121	33963	181	47745
2	364	62	12487	122	34327	182	47872
3	728	63	12851	123	34691	183	47999
4	1092	64	13215	124	35055	184	48126
5	1456	65	13579	125	35419	185	48253
6	1820	66	13943	126	35783	186	48380
7	2184	67	14307	127	36147	187	48507
8	2548	68	14671	128	36511	188	48634
9	2912	69	15035	129	36875	189	48761
10	3276	70	15399	130	37239	190	48888
11	3640	71	15763	131	37603	191	49015
12	4004	72	16127	132	37967	192	49142

13	4368	73	16491	133	38331	193	49269
14	4732	74	16855	134	38695	194	49396
15	5096	75	17219	135	39059	195	49523
16	5460	76	17583	136	39423	196	49650
17	5824	77	17947	137	39787	197	49777
18	6188	78	18311	138	40151	198	49904
19	6552	79	18675	139	40515	199	50031
20	6916	80	19039	140	40879	200	50158
21	7280	81	19403	141	41243	201	50285
22	7407	82	19767	142	41607	202	50412
23	7534	83	20131	143	41971	203	50539
24	7661	84	20495	144	42335	204	50903
25	7788	85	20859	145	42699	205	51267
26	7915	86	21223	146	43063	206	51631
27	8042	87	21587	147	43427	207	51995
28	8169	88	21951	148	43554	208	52359
29	8296	89	22315	149	43681	209	52723
30	8423	90	22679	150	43808	210	53087
31	8550	91	23043	151	43935	211	53451
32	8677	92	23407	152	44062	212	53815
33	8804	93	23771	153	44189	213	54179
34	8931	94	24135	154	44316	214	54543
35	9058	95	24499	155	44443	215	54907
36	9185	96	24863	156	44570	216	55271
37	9312	97	25227	157	44697	217	55635
38	9439	98	25591	158	44824	218	55999
39	9566	99	25955	159	44951	219	56363
40	9693	100	26319	160	45078	220	56727
41	9820	101	26683	161	45205	221	57091
42	9947	102	27047	162	45332	222	57455
43	10074	103	27411	163	45459	223	57819
44	10201	104	27775	164	45586	224	58183
45	10328	105	28139	165	45713	225	58547
46	10455	106	28503	166	45840	226	58911
47	10582	107	28867	167	45967	227	59275
48	10709	108	29231	168	46094	228	59639
49	10836	109	29595	169	46221	229	60003
50	10963	110	29959	170	46348	230	60367
51	11090	111	30323	171	46475	231	60731
52	11217	112	30687	172	46602	232	61095
53	11344	113	31051	173	46729	233	61459
54	11471	114	31415	174	46856	234	61823
55	11598	115	31779	175	46983	235	62187
56	11725	116	32143	176	47110	236	62551

57	11852	117	32507	177	47237	237	62915
58	11979	118	32871	178	47364	238	63279
59	12106	119	33235	179	47491	239	63589
60	12233	120	33599	180	47618	240	63900

经计算，240 辆列车全天的能耗为 1.0315×10^8 kJ。

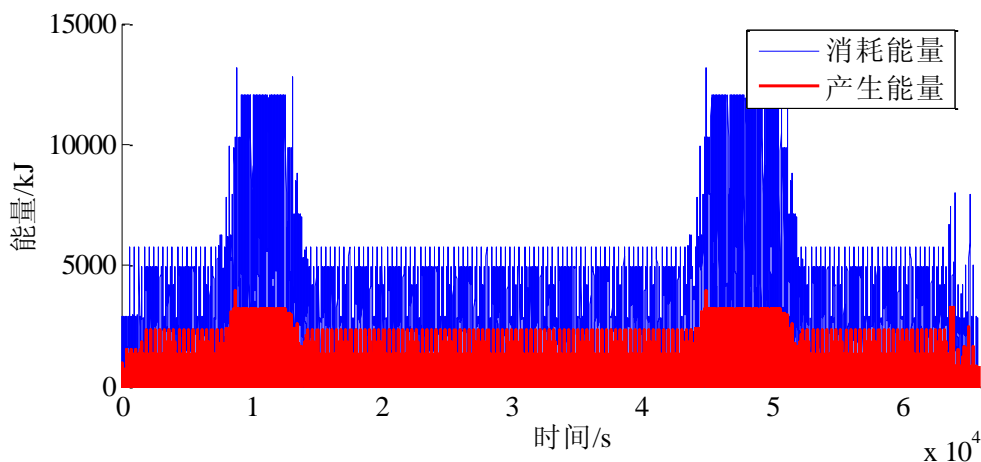


图 5.11 240 列列车每时刻的耗能与产能

由上图 5.11 可知，列车消耗的能量远大于回收的能量，另外，两个发车高峰期内能耗需求明显。

六、 问题三求解

6.1 数学模型

6.1.1 列车延误分析及控制策略

轨道交通一般具有行车密度大，列车追踪时间短的特点。如果发生列车延误，极易造成延误传播，使后续车辆相继出现延误现象，直接扰乱列车车间追踪间隔。因此列车延误运行调整对于列车运行图的完整实现尤为重要。

现行的列车延误调整策略一般采用晚点列车的“赶点”方式,即一般会通过减少晚点列车在下一区间的运行时间以及下一车站的停靠时间来调整。但是这种调整措施存在明显的弊端：（1）仅用于前车（2）仅用于前车晚点结束后（3）对前车晚点期间后车“紧随到站”现象不予调整，这样会导致前后两车之间的间隔过小，导致后车欠载，不利于列车之间对于时段客流的合理分担。

上述调整策略只是针对于晚点列车本身，本文提出一种基于后车“延赶结合”的列车延误调整控制策略。“延赶结合”运行调整策略是在前车“赶点”基础上，扩充两项策略：

1) 在“晚点发生”期间, 延长后车的前一区间运行时分, 降低后车“紧随到站/站外停车”的可能，称为“延点”运行；

2) 后车在下一区间“赶点”运行。“延赶结合”运行调整策略作为均衡发到间隔的一种可行策略，在一个区间内消除延误传播根源，并通过后车“延点”能耗节省弥补前车“赶点”能耗浪费。“延赶结合”运行调整与传统的赶点的策略对比如图6.1和图6.2

所示。

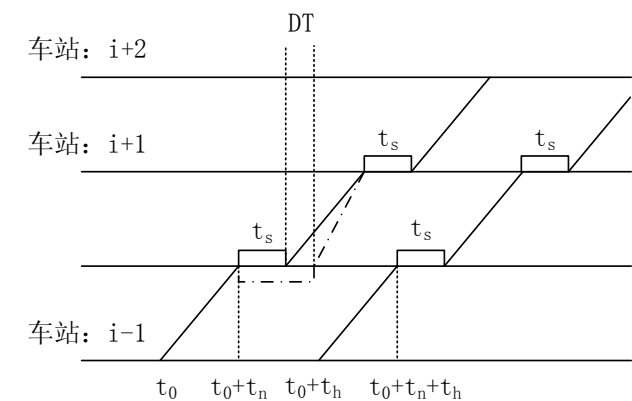


图 6.1 现行调整方式

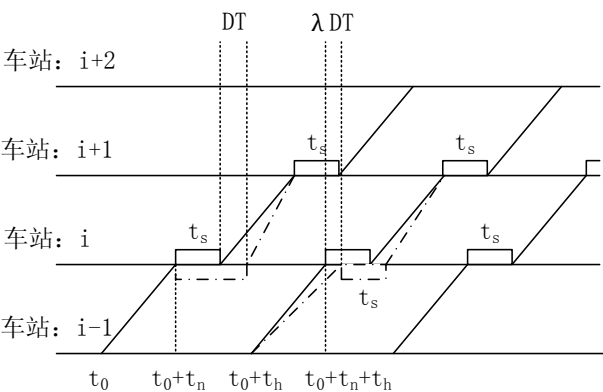


图 6.2 后车“延赶结合”调整方式

经过后车的“延赶结合”，可以使后二车以及以后的车辆完全不受前车的延误影响，而后车经过两段区间的调整，在下一车站能够实现准点运行。这样使得前车后车之间的追踪间隔比预定时间间隔减少 $(1-\lambda)DT_j^i$ ， $0\leq\lambda\leq1$ ，后车与后二车之间的追踪间隔也减少 λDT_j^i ($0\leq x\leq1$)，因此就将延误时间在后两车之间分散开来。依据此方法，还可以将列车延误时间在后车之间分散开来。

6.1.2 数学模型

① 两车间的延误优化控制模型及求解方法

基于以上对于列车延误问题的讨论，只考虑两车的延误优化控制模型如(6—1)所述：

$$\begin{cases}
\min E_{i+1}(T) = E_{i+1}(T_{j-1}) + E_{i+1}(T_j) = \int_0^{T_{j-1} + \lambda DT_j^i} u_f(t)v(t)F(t)dt + \int_{T_{j-1} + \lambda DT_j^i}^{T_j + (1-\lambda)DT_j^i} u_f(t)v(t)F(t)dt \\
s.t. \\
\frac{dv(t)}{dt} = u_f(t)F(v) - u_b(t)B(v) - \omega_0(v) - \omega_1(x) \\
L_{j-1} = \int_0^{T_{j-1} + \lambda DT_j^i} v(t)dt \\
L_j = \int_{T_{j-1} + \lambda DT_j^i}^{T_j + (1-\lambda)DT_j^i} v(t)dt \\
v(0) = v(T_{j-1} + \lambda DT_j^i) = v[T_j + (1-\lambda)DT_j^i] = 0 \\
v \leq \min(\bar{v}, 2\sqrt{sB_e}) \\
0 \leq u_f(t) \leq 1 \\
0 \leq u_b(t) \leq 1 \\
t_s - \lambda DT_j^i \geq \Delta T_{\min} \\
t_s - (1-\lambda)DT_j^i \geq \Delta T_{\min} \\
0 \leq \lambda \leq 1
\end{cases} \quad (6-1)$$

式中, ΔT_{\min} 为最小追踪间隔; B_e 为最大制动加速度。

前车晚点时间时刻与后车发车时间之间先后顺序不同, 本模型的求解思路不同:

1) 列车延误结束后车尚未发车, 此时前车的延误时间以及经确定。所以可以按照最大加速牵引-巡航-惰行-制动的操纵序列在每段区间进行优化, 再通过合理分配 DT_j^i 在两区间的配比, 即可求出最小能耗。

2) 前车延误时刻晚于后车发车时刻, 在后车发车到得知前车发生延误的时段内, 后车按照既定的操纵序列操纵行进, 在得知列车延误到列车延误结束时段内, 根据列车所处的运行状态转入牵引调整或惰行调整, 最后前车晚点结束后, 以此时的速度为初速度继续进行操纵序列的优化 (如图6.3所示)。

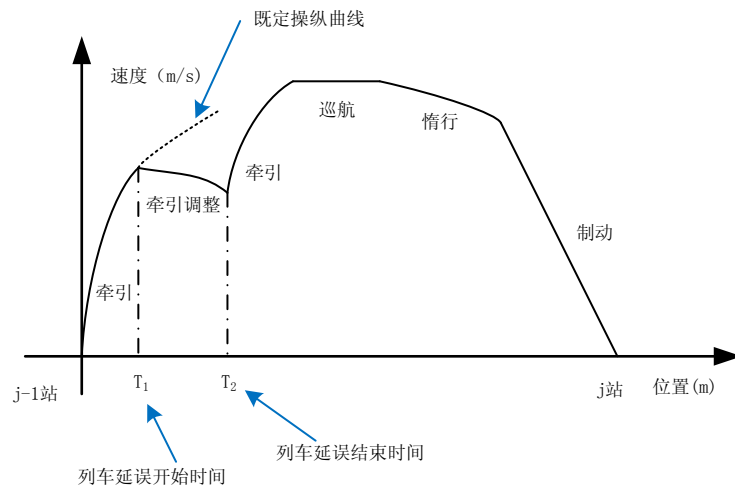


图 6.3 后车调整方式图

牵引调整是指牵引工况下, 提前切断(直流/交流)电机功率输送, 实现牵引末速的降低。该方法通过降低牵引末速进而间接降低制动初速, 促成区间平均速度降低, 从而延长

区间运行总时分。

惰行调整是指惰行工况下,以空气制动(闸瓦/盘式)或电磁制动(磁轨涡流/磁轨转子)辅助电气制动,通过调速制动,实现制动初速的降低。

② 考虑多车的延误优化控制模型及求解方法

当列车延误时较长,使无论怎样调整延误时间在前车后一车,后一车后二车之间的分配,都导致列车的跟踪时间小于最小跟踪时间,此时需要将延误时间在后车、后一车、后两车或随后的车之间分配,对于这类问题,本文给出以下最优控制模型:

$$\left\{ \begin{array}{l} \min \sum_{p=i+1}^{i+n} E_p(T) = \sum_{p=i+1}^{i+n} E_p(T_{j-1}) + \sum_{p=i+1}^{i+n} E_p(T_j) \\ s.t \\ \sum_1^n \lambda_i = 1 \\ 0 \leq \lambda_i \leq 1 \\ \frac{dv(t)}{dt} = u_f(t)F(v) - u_b(t)B(v) - \omega_0(v) - \omega_1(x) \\ L_{j-1} = \int_0^{T_{j-1} + \lambda DT_j^i} v(t)dt \\ L_j = \int_{T_{j-1} + \lambda DT_j^i}^{T_j + (1-\lambda)DT_j^i} v(t)dt \\ v(0) = v(T_{j-1} + \lambda DT_j^i) = v[T_j + (1-\lambda)DT_j^i] = 0 \\ v \leq \min(\bar{v}, 2\sqrt{sB_e}) \\ 0 \leq u_f(t) \leq 1 \\ 0 \leq u_b(t) \leq 1 \\ t_s - \lambda DT_j^i \geq \Delta T_{\min} \\ t_s - (1-\lambda)DT_j^i \geq \Delta T_{\min} \end{array} \right. \quad (6-2)$$

由于该模型控制变量过多,难以用数值方法求解。这里,提出采用遗传算法进行求解。遗传算法是模拟达尔文生物进化论的自然选择和遗传学机理的生物进化过程的计算模型,是一种通过模拟自然进化过程搜索最优解的方法。遗传算法体现的是随机和迭代过程对种群中个体的作用。每个个体由一个或多个染色体组成,并且每个个体代表一种可行解。每一个个体根据对求解目标的适应性赋予权值,适应度高的个体将有更高的可能通过染色体复制传给下一代,并通过交叉和变异保持种群的多样性。经过多次迭代后的个体将有对目标解的更高适应度^{[11][12]}。遗传算法的流程如图 6.4 所示:

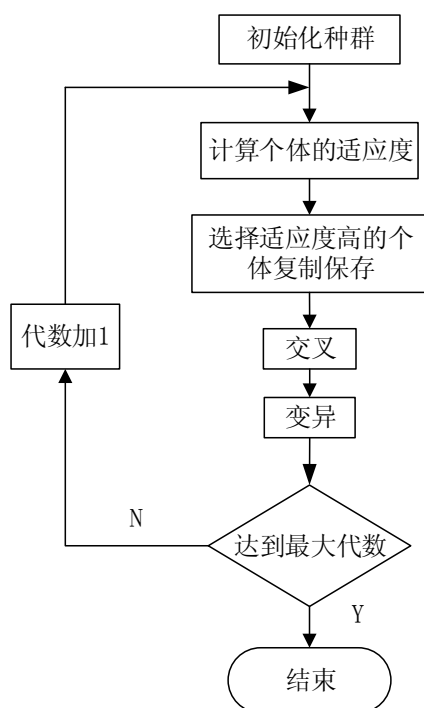


图 6.4 遗传算法

七、 模型的评价与推广

7.1 模型评价

7.1.1 模型的优点

①第一题中，首先对四种运行状态以及转换关系进行分析，总结出最节能的 5 种列车操控策略，不仅有实际意义，而且这对于无法解析求解的数学模型是极大地简化。通过简单的数值迭代即可求取速度—路程曲线以及最小能耗。在模型求解方法方面，提出引入二分法来准确寻找最小能量点，几步迭代后，即可求得较精确解，大大提高了计算效率。在求解两路段最小能量的问题时，引入了贪心算法，将无法解析的问题变得可以数值求解，相比枚举法来看，提高计算效率。

②第二题中，将一个复杂模型分两步进行优化，基于单车全程优化以及多车间隔优化，在优化过程中，进一步简化，使得问题可解，并且有利于实际的控制实现。在求解方法方面引入了分治策略，将多车分组优化，反复使用两车以及三车优化的结论当作参考进行全局优化。

③第三题中，比较之前的两题又引入了列车延误问题，分别提出了针对两车厌恶和多车延误的调整策略的优化模型。

7.1.2 模型的缺点

本文的所提出模型在一定程度上为简化计算，对问题作了部分简化。例如，问题二中，分组优化假定每个组的发车策略一致，并假定所有列车整体运行策略取一致。虽然简化了分析，但是不利于全局的最优。另外，部分模型的求解最优性缺乏严格数学证明，不利于模型推广。

7.2 模型改进

针对问题的分治策略，可以进行如下改进：对于多列车的发车间隔优化问题，可以在上文基础上将分治策略进一步深入，不仅仅是单步的分治优化，而是将分治策略不断进行。如图 7.1 所示，将所有列车两两分组优化，两两优化结束后，可将一个单位组再次等值成一列列车，该列车的运行时间长度变成 $h+T$ 。如此往复循环，直到只剩下一个间隔需要优化，将所有列车最终等值成一列列车。另外，如果某次待优化组数为奇数，则根据上文方法进行三车优化。

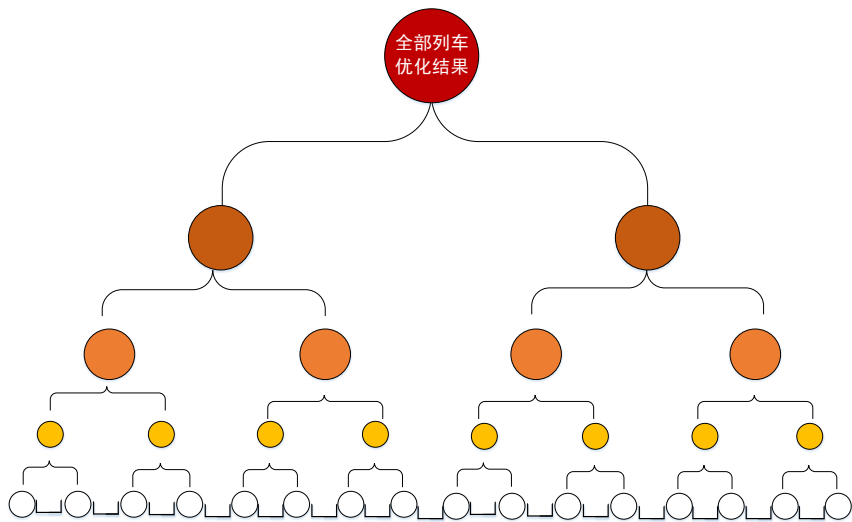


图 7.1 分治优化过程示意图

7.3 模型推广

本文模型是针对牵引交通系统，对于工程中具有相同最优控制策略特点的问题，本文模型及求解方法同样有一定适用性。

八、 参考文献

[1] 付印平. 列车追踪运行与节能优化建模与模拟研究[D]. 北京交通大学, 2009, 北京.

[2] 顾青. 城市轨道交通列车节能优化驾驶研究[D]. 北京交通大学, 2014, 北京.

[3] Ross I M. A Primer on Pontryagin’s Principle in Optimal Control[M]. Collegiate Publishers, 2009.

[4] 金炜东, 王自力, 李崇维, 苟先太, 靳蕃. 列车节能操纵优化方法研究[J]. 铁道学报, 1997, 19(6): 58-62.

[5] Su S, Tang T, Li X, et al. Optimization of multitrain operations in a subway system[J]. Intelligent Transportation Systems, IEEE Transactions on, 2014, 15(2): 673-684.

[6] 宿帅, 唐涛. 城市轨道交通 ATO 的节能优化研究[J]. 铁道学报, 2014, 36(12):50-55.

[7] E. Khmelnitsky, On an optimal control problem of train operation[J], IEEE Transactions on Automatic Control, 45(7): 1257 – 1266.

[8] 李坤妃. 多列车协同控制节能优化方法的研究[D]. 北京交通大学, 2014, 北京.

[9] Ning Zh, Clive Roberts, Stuart Hillmansen, Gemma Nicholson, et al. A Multiple Train

Trajectory Optimization to Minimize Energy Consumption and Delay[J]. Intelligent Transportation Systems, IEEE Transactions on, 2015, pp: 1-10.

- [10] 吴洋, 罗霞. 一种晚点地铁列车实时调整策略及其动态速控模式[J]. 中国铁道科学, 2005, 26(6): 113-118.
- [11] 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[J]. 铁道学报, 1998, 20(5): 47-52.
- [12] 陈荣武, 刘莉, 郭进. 基于遗传算法的列车运行能耗优化算法[J]. 交通运输工程学报, 2012, 12(1): 108-114.

九、 附录

附录 A 主要 Matlab 程序代码

1. 单列车单路段最小能量求解

```
clear;
clc;
w11=load('附加阻力.txt');%读入按公里标的处理好附加阻力
for i=12240:13594
    w1(13594-i+1)=-w11(i);
end
N=100000;%整个过程按距离分成N段
m=194.295*1000;
dL=(13594-12240)/N;
vm=80/3.6;
g=9.8;
step=0;
E1=30000e3;%二分法初始头指针
E2=40000e3;%二分法初始尾指针
%二分法迭代
while 1
    v=[];p=[];q=[];sp=[];sq=[];s=[];
    s(1)=1;v(1)=0;
    i=1;
    v(N+1)=0;
    w0(1)=2.031+0.0622*(v(1)*3.6)+0.001807*(v(1)*3.6)^2;
    F=[];
    E=(E1+E2)/2;
    EE=E;
    %最大牵引过程
    while 1
        i=i+1;
        if v(i-1)<=51.5/3.6 F(i-1)=203;end
        if v(i-1)>51.5/3.6
```

```

F(i-1)=-0.002032*(v(i-1)*3.6)^3+0.4928*(v(i-1)*3.6)^2-42.13*v(i-1)*3.6+13
43;end
w0(i-1)=2.031+0.0622*v(i-1)*3.6+0.001807*(v(i-1)*3.6)^2;
w(i-1)=w0(i-1)+w1(floor(s(i-1)));
a=(F(i-1)*1000-w(i-1)*g*m/1000)/m;
v(i)=sqrt(v(i-1)^2+2*a*dL);
s(i)=s(i-1)+dL;
E=E-F(i-1)*1000*dL;
    if v(i)>=vm v(i)=vm;break;end
if E<=0 break; end
end
%匀速过程
while E>0
    i=i+1;
    v(i)=v(i-1);
    s(i)=s(i-1)+dL;
    w0(i-1)=2.031+0.0622*v(i-1)*3.6+0.001807*(v(i-1)*3.6)^2;
    w(i-1)=w0(i-1)+w1(floor(s(i-1)));
    E=E-w(i-1)*m*g/1000*dL;
end
%惰行过程
k=i;
p(k)=v(k);
sp(k)=s(k);
while k<N+1
    wp0(k)=2.031+0.0622*p(k)*3.6+0.001807*(p(k)*3.6)^2;
    wp(k)=wp0(k)+w1(floor(sp(k)));
    p(k+1)=sqrt(p(k)^2-2*dL*wp(k)*g/1000);
    sp(k+1)=sp(k)+dL;
    k=k+1;
end
%最大制动
j=N+1;
q(N+1)=0;
sq(N+1)=13594-12240;
while j>i
    if q(j)>77/3.6 B(j)=0.1343*(q(j)*3.6)^2-25.07*q(j)*3.6+1300;end
    if q(j)<=77/3.6 B(j)=166;end
    wq0(j)=2.031+0.0622*q(j)*3.6+0.001807*(q(j)*3.6)^2;
    a=(B(j)*1000+(wq0(j)+w1(ceil(sq(j))))*m*g/1000)/m;
    q(j-1)=sqrt(q(j)^2+2*a*dL);
    sq(j-1)=sq(j)-dL;
    j=j-1;
end

```

```

%求解最大刹车线与惰行线交点
while i<N+1
    i=i+1;
    v(i)=min(q(i),p(i));
end
T=0;
for i=2:N
    T=T+dL/v(i);
end
if T>110 E1=EE;end
if T<110 E2=EE;end
if abs(T-110)<0.001 break;end %判断时间是否满足条件
end
i=0:N;
plot(i*dL,v*3.6);%输出v-s曲线
disp(EE/1000);%输出最小能量，单位：kJ

```

2. 单列车两路段最小耗能求解程序（与单列车全程 A1-A14 的优化为同一程序，仅是处理路段差异）

```

clear;clc;
aa=load('A1A14ET.txt');%读入求出的十四个路段的E-T曲线
E=[20000 20000]; %给出E迭代初值
T=[147.0769089 145.8077719]; %给出T迭代初值
a=aa(:,11:14);
%求E-T曲线各段斜率
for i=1:2
    for j=1:99
        dt=a(j,i*2)-a(j+1,i*2);
        de=a(j,i*2-1)-a(j+1,i*2-1);
        b(i,j)=dt/de;
    end
end
dtde=b(:,1);
sumt=sum(T);
sume=sum(E);
dE=1; %能量迭代步长
k=0;
%开始迭代过程
while 1
    k=k+1;
    sume=sume+dE;
    [mindtde,sec]=min(dtde);
    T(sec)=T(sec)+mindtde*dE;
    E(sec)=E(sec)+dE;
end

```

```

sumt
for j=1:99
    if a(j,sec*2-1)<=E(sec) & a(j+1,sec*2-1)>E(sec) break; end
end
    dtde(sec)=b(sec,j); %更新斜率
    ssec(k,1)=sec;
    ssec(k,2)=j;
    sumt=sum(T);
    if (sumt-220)<0.001 break;end %判断时间是否符合要求 ,精确到0.001s
end
disp(T);

```

3. 两相邻列车最佳发车间隔求解程序

```

Clear;
Clc;
p1=load('P-t.txt'); %读入处理好的单列车全程功率数据,牵引耗电以及刹车放电
Em=0;
E=[];
ef=[];
for i=1:2086
    if p1(i)<0
        Em=Em-p1(i);
    end
end
for i=2086:2086+660 p1(i)=0;end
for h=120:660
    p2=[];
    for j=h+1:1:2086+h
        p2(j)=p1(j-h);
    end
    for t=1:2086+h
        es(h,t)=0;
        ef(h,t)=0;
        psum(h,t)=0;
        if p1(t)>0 es(h,t)=es(h,t)+p1(t);end
        if p1(t)<0 ef(h,t)=ef(h,t)-p1(t);end
        if p2(t)>0 es(h,t)=es(h,t)+p2(t);end
        if p2(t)<0 ef(h,t)=ef(h,t)-p2(t);end
        if es(h,t)>=ef(h,t) psum(h,t)=es(h,t)-ef(h,t);end
        if es(h,t)<ef(h,t) psum(h,t)=0;end
    end
    Esum(h)=sum(psum(h,:));
end
h=120:660;
plot(h,Esum(120:660));

```

```
min(Esum(120:660));
```

4. 三相邻列车最佳发车间隔求解程序

```
clear;
p=load('P-t.txt'); %读入处理好的单列车全程功率数据，牵引耗电以及刹车放电
E=[];
Em=0;
sce=[];
cas=[];
num=0;
for i=1:2086+1320
    p1(i)=0;
    p2(i)=0;
    p3(i)=0;
end
for i=1:2086
    if p(i)<0
        Em=Em-p(i);
    end
end
for H=1277:1320
    for i=1:2086+1320
        p2(i)=0;
        p3(i)=0;
    end
    num=0;
    for h1=H-660:660
        num=num+1;
        sce(H,num)=0;
        h2=H-h1;
        cas(H,num,1)=h1;
        cas(H,num,2)=h2;
        p1=p;
        for i=2087:2086+1320
            p1(i)=0;
        end
        for j=1:1:2086
            p2(j+h1)=p1(j);
            p3(j+h1+h2)=p1(j);
        end
        for t=1:2086+H
            ef=0;es=0;
            psum=0;
            if p1(t)<0 ef=ef-p1(t);end
            if p2(t)<0 ef=ef-p2(t);end
```

```

        if p3(t)<0 ef=ef-p3(t);end
        if p1(t)>0 es=es+p1(t);end
        if p2(t)>0 es=es+p2(t);end
        if p2(t)>0 es=es+p2(t);end
        if es>=ef psum=es-ef;end
        if es<ef psum=0;end
        sce(H,num)=sce(H,num)+psum;
    end
end
end
for H=1277:1320
    [E(H),Nmin(H)]=min(sce(H,1:1320-H+1));
end
E=E;
H=1277:1320;
plot(H,E(H));
for H=1277:1320
    best(H,1)=cas(H,Nmin(H),1);
    best(H,2)=cas(H,Nmin(H),2);
    best(H,3)=E(H);
end

```

5. 100列列车的耗能计算（与240列耗能计算同程序）

```

clear;clc;
p=load('P-t.txt');
h=load('ncar.txt');
psum=p;
t=0;
for i=1:63900+2086
    es(i)=0;ef(i)=0;
end
for i=1:2086
    if p(i)>0
        es(i)=p(i);
    end
    if p(i)<0
        ef(i)=-p(i);
    end
end
for k=2:240
    t=t+h(k-1);
    pk=[];
    for j=1:2086
        pk(j+t)=p(j);
        if pk(j+t)>0
            es(j+t)=es(j+t)+pk(j+t);

```



```

end
if pk(j+t)<0
    ef(j+t)=ef(j+t)-pk(j+t);
end
end
end
for i=1:63900+2086
    if ef(i)>=es(i) psum(i)=0;end
    if ef(i)<es(i) psum(i)=es(i)-ef(i);end
end
sum(psum)
hold on;
plot(es);
plot(ef,'r','linewidth',2);

```

附录 B 问题一（1）计算结果

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	13594	0	0	0
0:00:01	0.021682278	7.805619981	2.168227773	2.31534	13591.68466	0	203000	440.1502378
0:00:02	0.031781393	11.44130153	1.009911541	2.65384	13589.03082	0	203000	645.1622807
0:00:03	0.04184694	15.06489844	1.006554698	3.6558	13585.37502	0	203000	849.4928844
0:00:04	0.051932724	18.69578061	1.008578379	4.68484	13580.69018	0	203000	1054.234295
0:00:05	0.061945093	22.30023363	1.001236952	5.67326	13575.01692	0	203000	1257.485396
0:00:06	0.071916879	25.8900765	0.997178574	6.67522	13568.3417	0	203000	1459.912647
0:00:07	0.081846515	29.46474533	0.992963563	7.67718	13560.66452	0	203000	1661.48425
0:00:08	0.091717149	33.01817376	0.987063455	8.6656	13551.99892	0	203000	1861.858132
0:00:09	0.101530609	36.55101917	0.981345945	9.65402	13542.3449	0	203000	2061.071359
0:00:10	0.111275131	40.05904699	0.974452172	10.6289	13531.716	0	203000	2258.885149
0:00:11	0.12094205	43.53913796	0.966691938	11.59024	13520.12576	0	203000	2455.123613
0:00:12	0.130544522	46.99602804	0.960247244	12.56512	13507.56064	0	203000	2650.053803
0:00:13	0.140063474	50.42285051	0.95189513	13.51292	13494.04772	0	203000	2843.288515
0:00:14	0.149210968	53.71594861	0.914749473	14.46072	13479.587	0	186929.6676	2789.195672
0:00:15	0.157255966	56.61214771	0.804499748	15.31374	13464.27326	0	168639.755	2651.960755
0:00:16	0.164402629	59.18494638	0.714666298	16.07198	13448.20128	0	154479.5333	2539.684138
0:00:17	0.17083934	61.5021623	0.643671088	16.76252	13431.43876	0	143227.6961	2446.892503
0:00:18	0.176687336	63.60744095	0.584799624	17.35828	13414.08048	0	134106.5422	2369.492768
0:00:19	0.179213198	64.51675111	0.252586156	17.8728	13396.20768	0	0	0
0:00:20	0.177891363	64.04089054	-0.13218349	17.84572	13378.36196	0	0	0
0:00:21	0.176583437	63.57003717	-0.1307926	17.71032	13360.65164	0	0	0
0:00:22	0.175289298	63.10414734	-0.12941384	17.57492	13343.07672	0	0	0
0:00:23	0.174007835	62.64282073	-0.12814628	17.45306	13325.62366	0	0	0
0:00:24	0.172739929	62.18637453	-0.12679061	17.31766	13308.306	0	0	0
0:00:25	0.171484477	61.73441175	-0.12554522	17.1958	13291.1102	0	0	0

0:00:26	0.170076794	61.22764589	-0.1407683	17.07394	13274.03626	1.8	0	0
0:00:27	0.168672311	60.72203179	-0.14044836	16.925	13257.11126	1.8	0	0
0:00:28	0.167282473	60.22169016	-0.13898379	16.77606	13240.3352	1.8	0	0
0:00:29	0.165904928	59.72577391	-0.13775451	16.6542	13223.681	1.8	0	0
0:00:30	0.164541808	59.23505078	-0.13631198	16.50526	13207.17574	1.8	0	0
0:00:31	0.163191887	58.74907941	-0.13499205	16.36986	13190.80588	1.8	0	0
0:00:32	0.161855063	58.26782273	-0.13368241	16.23446	13174.57142	1.8	0	0
0:00:33	0.160530122	57.79084386	-0.13249413	16.1126	13158.45882	1.8	0	0
0:00:34	0.15921808	57.31850887	-0.13120416	15.9772	13142.48162	1.8	0	0
0:00:35	0.157918841	56.85078288	-0.12992389	15.8418	13126.63982	1.8	0	0
0:00:36	0.15663231	56.38763176	-0.12865309	15.7064	13110.93342	1.8	0	0
0:00:37	0.155357288	55.92862359	-0.12750227	15.58454	13095.34888	1.8	0	0
0:00:38	0.154093686	55.47372707	-0.12636014	15.46268	13079.8862	1.8	0	0
0:00:39	0.152841421	55.02291152	-0.12522654	15.34082	13064.54538	1.8	0	0
0:00:40	0.151601512	54.57654415	-0.12399094	15.20542	13049.33996	1.8	0	0
0:00:41	0.150372771	54.13419745	-0.12287408	15.08356	13034.2564	1.8	0	0
0:00:42	0.149155118	53.69584264	-0.12176523	14.9617	13019.2947	1.8	0	0
0:00:43	0.147948476	53.26145153	-0.1206642	14.83984	13004.45486	1.8	0	0
0:00:44	0.146752768	52.83099653	-0.11957083	14.71798	12989.73688	1.8	0	0
0:00:45	0.14556682	52.40405514	-0.11859483	14.60966	12975.12722	1.8	0	0
0:00:46	0.144391659	51.98099707	-0.11751613	14.4878	12960.63942	1.8	0	0
0:00:47	0.143227212	51.56179649	-0.11644461	14.36594	12946.27348	1.8	0	0
0:00:48	0.142073412	51.14642814	-0.1153801	14.24408	12932.0294	1.8	0	0
0:00:49	0.140929091	50.73447288	-0.11443202	14.13576	12917.89364	1.8	0	0
0:00:50	0.140020965	50.40754748	-0.09081261	14.02744	12903.8662	-3.5	0	0
0:00:51	0.139410846	50.18790454	-0.06101193	13.95974	12889.90646	-3.5	0	0
0:00:52	0.138805659	49.97003719	-0.06051871	13.90558	12876.00088	-3.5	0	0
0:00:53	0.138205954	49.75414357	-0.05997045	13.83788	12862.163	-3.5	0	0
0:00:54	0.13761169	49.54020841	-0.05942643	13.77018	12848.39282	-3.5	0	0
0:00:55	0.137022243	49.32800757	-0.05894468	13.71602	12834.6768	-3.5	0	0
0:00:56	0.13643758	49.11752867	-0.05846636	13.66186	12821.01494	-3.5	0	0
0:00:57	0.135857665	48.90875943	-0.05799145	13.6077	12807.40724	-3.5	0	0
0:00:58	0.135283039	48.7018941	-0.05746259	13.54	12793.86724	-3.5	0	0
0:00:59	0.13471309	48.49671244	-0.05699491	13.48584	12780.3814	-3.5	0	0
0:01:00	0.134147785	48.29320244	-0.05653055	13.43168	12766.94972	-3.5	0	0
0:01:01	0.133587656	48.0915561	-0.05601287	13.36398	12753.58574	-3.5	0	0
0:01:02	0.133031536	47.891353	-0.05561197	13.32336	12740.26238	-3.5	0	0
0:01:03	0.132480523	47.69298834	-0.05510129	13.25566	12727.00672	-3.5	0	0
0:01:04	0.131933459	47.49604525	-0.05470641	13.21504	12713.79168	-3.5	0	0
0:01:05	0.131391433	47.30091576	-0.05420264	13.14734	12700.64434	-3.5	0	0
0:01:06	0.130853296	47.10718649	-0.05381368	13.10672	12687.53762	-3.5	0	0
0:01:07	0.130320129	46.91524635	-0.05331671	13.03902	12674.4986	-3.5	0	0
0:01:08	0.129790793	46.72468541	-0.05293359	12.9984	12661.5002	-3.5	0	0
0:01:09	0.12926636	46.53588948	-0.05244332	12.9307	12648.5695	-3.5	0	0

0:01:10	0.1287457	46.34845204	-0.05206596	12.89008	12635.67942	-3.5	0	0
0:01:11	0.128229334	46.16256014	-0.05163664	12.83592	12622.8435	-3.5	0	0
0:01:12	0.12771723	45.97820282	-0.05121037	12.78176	12610.06174	-3.5	0	0
0:01:13	0.12720882	45.79517508	-0.05084104	12.74114	12597.3206	-3.5	0	0
0:01:14	0.126705153	45.61385512	-0.05036665	12.67344	12584.64716	-3.5	0	0
0:01:15	0.126205124	45.43384478	-0.05000287	12.63282	12572.01434	-3.5	0	0
0:01:16	0.125709243	45.25532749	-0.04958813	12.57866	12559.43568	-3.5	0	0
0:01:17	0.125216949	45.07810168	-0.04922939	12.53804	12546.89764	-3.5	0	0
0:01:18	0.124728748	44.90234938	-0.04882008	12.48388	12534.41376	-3.5	0	0
0:01:19	0.124244611	44.72806014	-0.04841368	12.42972	12521.98404	-3.5	0	0
0:01:20	0.123763986	44.55503488	-0.04806257	12.3891	12509.59494	-3.5	0	0
0:01:21	0.123286849	44.38326558	-0.0477137	12.34848	12497.24646	-3.5	0	0
0:01:22	0.122814219	44.21311868	-0.04726303	12.28078	12484.96568	-3.5	0	0
0:01:23	0.122344507	44.04402237	-0.0469712	12.2537	12472.71198	-3.5	0	0
0:01:24	0.121879245	43.87652821	-0.04652615	12.186	12460.52598	-3.5	0	0
0:01:25	0.121416858	43.71006898	-0.04623868	12.15892	12448.36706	-3.5	0	0
0:01:26	0.120958355	43.54500765	-0.04585037	12.10476	12436.2623	-3.5	0	0
0:01:27	0.120503197	43.38115078	-0.0455158	12.06414	12424.19816	-3.5	0	0
0:01:28	0.120051871	43.21867354	-0.04513257	12.00998	12412.18818	-3.5	0	0
0:01:29	0.119603845	43.05738407	-0.04480263	11.96936	12400.21882	-3.5	0	0
0:01:30	0.119159097	42.89727482	-0.04447479	11.92874	12388.29008	-3.5	0	0
0:01:31	0.118718108	42.73851895	-0.04409885	11.87458	12376.4155	-3.5	0	0
0:01:32	0.118280353	42.58092694	-0.04377556	11.83396	12364.58154	-3.5	0	0
0:01:33	0.117845809	42.42449137	-0.04345432	11.79334	12352.7882	-3.5	0	0
0:01:34	0.117414458	42.26920491	-0.04313513	11.75272	12341.03548	-3.5	0	0
0:01:35	0.116986278	42.11506025	-0.04281796	11.7121	12329.32338	-3.5	0	0
0:01:36	0.116561742	41.96222728	-0.0424536	11.65794	12317.66544	-3.5	0	0
0:01:37	0.109009404	39.24338553	-0.75523382	11.31944	12306.346	-3.5	0	0
0:01:38	0.100132278	36.04762022	-0.88771259	10.45288	12295.89312	-3.5	0	0
0:01:39	0.09120047	32.83216905	-0.89318088	9.5457	12286.34742	0	0	0
0:01:40	0.082108737	29.55914522	-0.90917329	8.65206	12277.69536	0	0	0
0:01:41	0.073073569	26.30648469	-0.90351681	7.74488	12269.95048	0	0	0
0:01:42	0.064080262	23.06889432	-0.89933066	6.85124	12263.09924	0	0	0
0:01:43	0.055150771	19.85427744	-0.89294913	5.94406	12257.15518	0	0	0
0:01:44	0.046247579	16.64912842	-0.89031917	5.06396	12252.09122	0	0	0
0:01:45	0.03739752	13.4631072	-0.8850059	4.17032	12247.9209	0	0	0
0:01:46	0.028579774	10.28871865	-0.8817746	3.29022	12244.63068	0	0	0
0:01:47	0.019802292	7.128825273	-0.87774816	2.41012	12242.22056	0	0	0
0:01:48	0.011106605	3.998377678	-0.86956878	1.53002	12240.69054	0	0	0
0:01:49	0.002665494	0.959577933	-0.84411104	0.66346	12240.02708	0	0	0
0:01:50	0	0	0	0.04062	12240	0	0	0

附录 C 问题一（2）计算结果

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	13594	0	0	0
0:00:01	0.0216823	7.80562	2.1682278	2.31534	13591.685	0	203000	440.15024
0:00:02	0.0317814	11.441302	1.0099115	2.65384	13589.031	0	203000	645.16228
0:00:03	0.0418469	15.064898	1.0065547	3.6558	13585.375	0	203000	849.49288
0:00:04	0.0519327	18.695781	1.0085784	4.68484	13580.69	0	203000	1054.2343
0:00:05	0.0619451	22.300234	1.001237	5.67326	13575.017	0	203000	1257.4854
0:00:06	0.0719169	25.890076	0.9971786	6.67522	13568.342	0	203000	1459.9126
0:00:07	0.0818465	29.464745	0.9929636	7.67718	13560.665	0	203000	1661.4843
0:00:08	0.0917171	33.018174	0.9870635	8.6656	13551.999	0	203000	1861.8581
0:00:09	0.1015306	36.551019	0.9813459	9.65402	13542.345	0	203000	2061.0714
0:00:10	0.1112751	40.059047	0.9744522	10.6289	13531.716	0	203000	2258.8851
0:00:11	0.120942	43.539138	0.9666919	11.59024	13520.126	0	203000	2455.1236
0:00:12	0.1305445	46.996028	0.9602472	12.56512	13507.561	0	203000	2650.0538
0:00:13	0.1400635	50.422851	0.9518951	13.51292	13494.048	0	203000	2843.2885
0:00:14	0.149211	53.715949	0.9147495	14.46072	13479.587	0	186929.67	2789.1957
0:00:15	0.157256	56.612148	0.8044997	15.31374	13464.273	0	168639.76	2651.9608
0:00:16	0.1644026	59.184946	0.7146663	16.07198	13448.201	0	154479.53	2539.6841
0:00:17	0.1708393	61.502162	0.6436711	16.76252	13431.439	0	143227.7	2446.8925
0:00:18	0.176279	63.460423	0.5439614	17.35828	13414.08	0	0	0
0:00:19	0.1749867	62.995209	-0.129226	17.56138	13396.519	0	0	0
0:00:20	0.1737091	62.535266	-0.127762	17.41244	13379.107	0	0	0
0:00:21	0.172444	62.079835	-0.126508	17.29058	13361.816	0	0	0
0:00:22	0.1711913	61.628879	-0.125265	17.16872	13344.647	0	0	0
0:00:23	0.169951	61.182361	-0.124033	17.04686	13327.601	0	0	0
0:00:24	0.1687229	60.740241	-0.122811	16.925	13310.676	0	0	0
0:00:25	0.1675079	60.302837	-0.121501	16.7896	13293.886	0	0	0
0:00:26	0.1661688	59.820763	-0.13391	16.68128	13277.205	1.8	0	0
0:00:27	0.164803	59.329091	-0.136576	16.53234	13260.672	1.8	0	0
0:00:28	0.1634505	58.842177	-0.135254	16.39694	13244.275	1.8	0	0
0:00:29	0.1621111	58.359986	-0.133942	16.26154	13228.014	1.8	0	0
0:00:30	0.1607836	57.882079	-0.132752	16.13968	13211.874	1.8	0	0
0:00:31	0.1594701	57.409224	-0.131349	15.99074	13195.883	1.8	0	0
0:00:32	0.1581683	56.940583	-0.130178	15.86888	13180.015	1.8	0	0
0:00:33	0.1568781	56.476125	-0.129016	15.74702	13164.267	1.8	0	0
0:00:34	0.1556017	56.016614	-0.127642	15.59808	13148.669	1.8	0	0
0:00:35	0.1543356	55.560822	-0.126609	15.48976	13133.18	1.8	0	0
0:00:36	0.153082	55.109514	-0.125363	15.35436	13117.825	1.8	0	0
0:00:37	0.1518396	54.662263	-0.124236	15.2325	13102.593	1.8	0	0
0:00:38	0.1506084	54.219038	-0.123118	15.11064	13087.482	1.8	0	0
0:00:39	0.1493884	53.779811	-0.122008	14.98878	13072.493	1.8	0	0

0:00:40	0.1481793	53.344554	-0.120905	14.86692	13057.626	1.8	0	0
0:00:41	0.1469812	52.913237	-0.11981	14.74506	13042.881	1.8	0	0
0:00:42	0.145794	52.485835	-0.118723	14.6232	13028.258	1.8	0	0
0:00:43	0.1446176	52.062321	-0.117643	14.50134	13013.757	1.8	0	0
0:00:44	0.1434508	51.642274	-0.11668	14.39302	12999.364	1.8	0	0
0:00:45	0.1422946	51.226065	-0.115614	14.27116	12985.093	1.8	0	0
0:00:46	0.141148	50.813273	-0.114664	14.16284	12970.93	1.8	0	0
0:00:47	0.1400119	50.404269	-0.113612	14.04098	12956.889	1.8	0	0
0:00:48	0.1388851	49.998637	-0.112676	13.93266	12942.956	1.8	0	0
0:00:49	0.1377676	49.596353	-0.111746	13.82434	12929.132	1.8	0	0
0:00:50	0.1366605	49.197789	-0.110712	13.70248	12915.429	1.8	0	0
0:00:51	0.1358732	48.914339	-0.078736	13.6077	12901.822	-3.5	0	0
0:00:52	0.1352985	48.707451	-0.057469	13.54	12888.282	-3.5	0	0
0:00:53	0.1347279	48.50204	-0.057058	13.49938	12874.782	-3.5	0	0
0:00:54	0.1341631	48.298713	-0.05648	13.41814	12861.364	-3.5	0	0
0:00:55	0.1336023	48.09684	-0.056076	13.37752	12847.987	-3.5	0	0
0:00:56	0.1330462	47.896615	-0.055618	13.32336	12834.663	-3.5	0	0
0:00:57	0.1324945	47.698027	-0.055163	13.2692	12821.394	-3.5	0	0
0:00:58	0.131948	47.501264	-0.054656	13.2015	12808.193	-3.5	0	0
0:00:59	0.1314053	47.305913	-0.054264	13.16088	12795.032	-3.5	0	0
0:01:00	0.1308677	47.112363	-0.053764	13.09318	12781.939	-3.5	0	0
0:01:01	0.1303339	46.920203	-0.053378	13.05256	12768.886	-3.5	0	0
0:01:02	0.1298051	46.72982	-0.052884	12.98486	12755.901	-3.5	0	0
0:01:03	0.12928	46.540805	-0.052504	12.94424	12742.957	-3.5	0	0
0:01:04	0.1287593	46.353348	-0.052072	12.89008	12730.067	-3.5	0	0
0:01:05	0.1282429	46.167436	-0.051642	12.83592	12717.231	-3.5	0	0
0:01:06	0.1277307	45.983059	-0.051216	12.78176	12704.449	-3.5	0	0
0:01:07	0.1272223	45.800012	-0.050846	12.74114	12691.708	-3.5	0	0
0:01:08	0.1267185	45.618672	-0.050372	12.67344	12679.035	-3.5	0	0
0:01:09	0.1262185	45.438643	-0.050008	12.63282	12666.402	-3.5	0	0
0:01:10	0.125722	45.259914	-0.049647	12.5922	12653.809	-3.5	0	0
0:01:11	0.1252302	45.082861	-0.049181	12.5245	12641.285	-3.5	0	0
0:01:12	0.1247414	44.906899	-0.048878	12.49742	12628.788	-3.5	0	0
0:01:13	0.1242572	44.732592	-0.048419	12.42972	12616.358	-3.5	0	0
0:01:14	0.1237765	44.559549	-0.048068	12.3891	12603.969	-3.5	0	0
0:01:15	0.1232999	44.387949	-0.047666	12.33494	12591.634	-3.5	0	0
0:01:16	0.1228267	44.217596	-0.04732	12.29432	12579.339	-3.5	0	0
0:01:17	0.1223569	44.048482	-0.046976	12.2537	12567.086	-3.5	0	0
0:01:18	0.1218911	43.880785	-0.046583	12.19954	12554.886	-3.5	0	0
0:01:19	0.1214291	43.714494	-0.046192	12.14538	12542.741	-3.5	0	0
0:01:20	0.1209706	43.549415	-0.045855	12.10476	12530.636	-3.5	0	0
0:01:21	0.1205154	43.385541	-0.045521	12.06414	12518.572	-3.5	0	0
0:01:22	0.120064	43.223046	-0.045137	12.00998	12506.562	-3.5	0	0
0:01:23	0.1196159	43.061739	-0.044807	11.96936	12494.593	-3.5	0	0

0:01:24	0.1191711	42.901613	-0.04448	11.92874	12482.664	-3.5	0	0
0:01:25	0.1187296	42.742659	-0.044154	11.88812	12470.776	-3.5	0	0
0:01:26	0.1182918	42.585051	-0.04378	11.83396	12458.942	-3.5	0	0
0:01:27	0.1178572	42.428599	-0.043459	11.79334	12447.148	-3.5	0	0
0:01:28	0.1174258	42.273297	-0.04314	11.75272	12435.396	-3.5	0	0
0:01:29	0.1169976	42.119136	-0.042822	11.7121	12423.684	-3.5	0	0
0:01:30	0.116573	41.966287	-0.042458	11.65794	12412.026	-3.5	0	0
0:01:31	0.1161511	41.814388	-0.042194	11.63086	12400.395	-3.5	0	0
0:01:32	0.1157327	41.663785	-0.041834	11.5767	12388.818	-3.5	0	0
0:01:33	0.1153175	41.514293	-0.041525	11.53608	12377.282	-3.5	0	0
0:01:34	0.1149053	41.365906	-0.041219	11.49546	12365.787	-3.5	0	0
0:01:35	0.1144962	41.218617	-0.040914	11.45484	12354.332	-3.5	0	0
0:01:36	0.1140901	41.072418	-0.040611	11.41422	12342.918	-3.5	0	0
0:01:37	0.1136865	40.927131	-0.040358	11.38714	12331.53	-3.5	0	0
0:01:38	0.1132864	40.783094	-0.04001	11.33298	12320.197	-3.5	0	0
0:01:39	0.1110944	39.993976	-0.219199	11.27882	12308.919	-3.5	0	0
0:01:40	0.1022183	36.798604	-0.887603	10.64244	12298.276	-3.5	0	0
0:01:41	0.0933332	33.599967	-0.88851	9.77588	12288.5	0	0	0
0:01:42	0.0842382	30.32576	-0.909502	8.85516	12279.645	0	0	0
0:01:43	0.0751851	27.066639	-0.905311	7.96152	12271.684	0	0	0
0:01:44	0.0661904	23.828532	-0.899474	7.05434	12264.629	0	0	0
0:01:45	0.0572399	20.606348	-0.895051	6.1607	12258.469	0	0	0
0:01:46	0.0483365	17.401152	-0.890332	5.26706	12253.202	0	0	0
0:01:47	0.0394885	14.215876	-0.884799	4.37342	12248.828	0	0	0
0:01:48	0.0306776	11.043931	-0.881096	3.49332	12245.335	0	0	0
0:01:49	0.021861	7.86997	-0.881656	2.62676	12242.708	0	0	0
0:01:50	0.0131623	4.7384165	-0.869876	1.73312	12240.975	0	0	0
0:01:51	0.0043535	1.5672626	-0.880876	0.8801	12240.095	0	0	0
0:01:52	0	0	0	0.10832	12240	0	0	0
0:01:53	0	0	0	0	12240	0	0	0
0:01:54	0	0	0	0	12240	0	0	0
0:01:55	0	0	0	0	12240	0	0	0
0:01:56	0	0	0	0	12240	0	0	0
0:01:57	0	0	0	0	12240	0	0	0
0:01:58	0	0	0	0	12240	0	0	0
0:01:59	0	0	0	0	12240	0	0	0
0:02:00	0	0	0	0	12240	0	0	0
0:02:01	0	0	0	0	12240	0	0	0
0:02:02	0	0	0	0	12240	0	0	0
0:02:03	0	0	0	0	12240	0	0	0
0:02:04	0	0	0	0	12240	0	0	0
0:02:05	0	0	0	0	12240	0	0	0
0:02:06	0	0	0	0	12240	0	0	0
0:02:07	0	0	0	0	12240	0	0	0

0:02:08	0	0	0	0	12240	0	0	0
0:02:09	0	0	0	0	12240	0	0	0
0:02:10	0	0	0	0	12240	0	0	0
0:02:11	0	0	0	0	12240	0	0	0
0:02:12	0	0	0	0	12240	0	0	0
0:02:13	0	0	0	0	12240	0	0	0
0:02:14	0	0	0	0	12240	0	0	0
0:02:15	0	0	0	0	12240	0	0	0
0:02:16	0	0	0	0	12240	0	0	0
0:02:17	0	0	0	0	12240	0	0	0
0:02:18	0	0	0	0	12240	0	0	0
0:02:19	0	0	0	0	12240	0	0	0
0:02:20	0	0	0	0	12240	0	0	0
0:02:21	0	0	0	0	12240	0	0	0
0:02:22	0	0	0	0	12240	0	0	0
0:02:23	0	0	0	0	12240	0	0	0
0:02:24	0	0	0	0	12240	0	0	0
0:02:25	0	0	0	0	12240	0	0	0
0:02:26	0	0	0	0	12240	0	0	0
0:02:27	0	0	0	0	12240	0	0	0
0:02:28	0	0	0	0	12240	0	0	0
0:02:29	0	0	0	0	12240	0	0	0
0:02:30	0	0	0	0	12240	0	0	0
0:02:31	0	0	0	0	12240	0	0	0
0:02:32	0	0	0	0	12240	0	0	0
0:02:33	0	0	0	0	12240	0	0	0
0:02:34	0	0	0	0	12240	0	0	0
0:02:35	0	0	0	0	12240	0	0	0
0:02:36	0	0	0	0	12240	0	0	0
0:02:37	0	0	0	0	12240	0	0	0
0:02:38	0.0216324	7.7876602	2.1632389	2.304	12237.696	0	203000	439.1375
0:02:39	0.0317339	11.424212	1.0101534	2.6496	12235.046	0	203000	644.19864
0:02:40	0.0418231	15.056328	1.0089209	3.6608	12231.386	0	203000	849.00958
0:02:41	0.0518887	18.679932	1.0065569	4.672	12226.714	0	203000	1053.3406
0:02:42	0.0619245	22.292805	1.0035758	5.6832	12221.03	0	203000	1257.0665
0:02:43	0.071908	25.886896	0.9983587	6.6816	12214.349	0	203000	1459.7333
0:02:44	0.0818267	29.457613	0.9918656	7.6672	12206.682	0	203000	1661.082
0:02:45	0.0916996	33.011847	0.9872874	8.6656	12198.016	0	203000	1861.5014
0:02:46	0.1015121	36.544362	0.9812541	9.6512	12188.365	0	203000	2060.696
0:02:47	0.1112541	40.051474	0.9741977	10.624	12177.741	0	203000	2258.4581
0:02:48	0.1209281	43.534105	0.9673978	11.5968	12166.144	0	203000	2454.8398
0:02:49	0.1305256	46.989206	0.95975	12.5568	12153.587	0	203000	2649.6691
0:02:50	0.1400486	50.417491	0.9523015	13.5168	12140.07	0	203000	2842.9863
0:02:51	0.1491928	53.709416	0.9144235	14.4512	12125.619	0	186973.95	2789.5171

0:02:52	0.1572445	56.608037	0.8051725	15.3216	12110.298	0	168663.89	2652.1477
0:02:53	0.1643948	59.182137	0.7150279	16.0768	12094.221	0	154494	2539.8015
0:02:54	0.1708299	61.498765	0.6435079	16.7552	12077.466	0	143243.23	2447.0227
0:02:55	0.1766826	63.605728	0.5852673	17.3696	12060.096	0	134113.57	2369.5531
0:02:56	0.1755954	63.214359	-0.108713	17.6	12042.496	0	0	0
0:02:57	0.1743108	62.751879	-0.128467	17.4848	12025.011	0	0	0
0:02:58	0.1730391	62.294094	-0.127163	17.3568	12007.654	0	0	0
0:02:59	0.1716174	61.782267	-0.142174	17.2288	11990.426	3	0	0
0:03:00	0.1700816	61.229363	-0.153584	17.0624	11973.363	3	0	0
0:03:01	0.1685599	60.681573	-0.152164	16.9216	11956.442	3	0	0
0:03:02	0.1670535	60.139266	-0.150641	16.768	11939.674	3	0	0
0:03:03	0.1655622	59.602401	-0.149129	16.6144	11923.059	3	0	0
0:03:04	0.1640848	59.070522	-0.147744	16.4736	11906.586	3	0	0
0:03:05	0.1626222	58.544004	-0.146255	16.32	11890.266	3	0	0
0:03:06	0.1611733	58.022393	-0.144892	16.1792	11874.086	3	0	0
0:03:07	0.1597379	57.505652	-0.143539	16.0384	11858.048	3	0	0
0:03:08	0.1583171	56.994156	-0.142082	15.8848	11842.163	3	0	0
0:03:09	0.1569096	56.487456	-0.14075	15.744	11826.419	3	0	0
0:03:10	0.1555142	55.985107	-0.139541	15.616	11810.803	3	0	0
0:03:11	0.154133	55.487897	-0.138114	15.4624	11795.341	3	0	0
0:03:12	0.1527638	54.99497	-0.136924	15.3344	11780.006	3	0	0
0:03:13	0.1514075	54.506705	-0.135629	15.1936	11764.813	3	0	0
0:03:14	0.1500629	54.022659	-0.134457	15.0656	11749.747	3	0	0
0:03:15	0.1487311	53.543212	-0.13318	14.9248	11734.822	3	0	0
0:03:16	0.1474109	53.067925	-0.132024	14.7968	11720.026	3	0	0
0:03:17	0.1461033	52.597178	-0.130763	14.656	11705.37	3	0	0
0:03:18	0.1448059	52.130121	-0.129738	14.5408	11690.829	3	0	0
0:03:19	0.143521	51.667549	-0.128492	14.4	11676.429	3	0	0
0:03:20	0.1422473	51.209024	-0.127368	14.272	11662.157	3	0	0
0:03:21	0.1409836	50.754109	-0.126365	14.1568	11648	3	0	0
0:03:22	0.1397322	50.3036	-0.125141	14.016	11633.984	3	0	0
0:03:23	0.1384907	49.856651	-0.124153	13.9008	11620.083	3	0	0
0:03:24	0.1372601	49.413649	-0.123056	13.7728	11606.31	3	0	0
0:03:25	0.1363027	49.068954	-0.095748	13.6576	11592.653	-2	0	0
0:03:26	0.1355779	48.80804	-0.072476	13.5808	11579.072	-2	0	0
0:03:27	0.1348588	48.549157	-0.071912	13.5168	11565.555	-2	0	0
0:03:28	0.1341459	48.292533	-0.071284	13.44	11552.115	-2	0	0
0:03:29	0.1334393	48.038153	-0.070661	13.3632	11538.752	-2	0	0
0:03:30	0.1327382	47.78576	-0.070109	13.2992	11525.453	-2	0	0
0:03:31	0.1320433	47.535581	-0.069494	13.2224	11512.23	-2	0	0
0:03:32	0.1313538	47.287359	-0.06895	13.1584	11499.072	-2	0	0
0:03:33	0.1306703	47.041323	-0.068343	13.0816	11485.99	-2	0	0
0:03:34	0.1299916	46.796977	-0.067874	13.0304	11472.96	-2	0	0
0:03:35	0.1293195	46.555028	-0.067208	12.9408	11460.019	-2	0	0

0:03:36	0.1286521	46.314744	-0.066746	12.8896	11447.13	-2	0	0
0:03:37	0.1279899	46.076351	-0.06622	12.8256	11434.304	-2	0	0
0:03:38	0.1273335	45.840075	-0.065632	12.7488	11421.555	-2	0	0
0:03:39	0.1266824	45.605665	-0.065114	12.6848	11408.87	-2	0	0
0:03:40	0.1260364	45.373108	-0.064599	12.6208	11396.25	-2	0	0
0:03:41	0.1253949	45.142158	-0.064153	12.5696	11383.68	-2	0	0
0:03:42	0.1247591	44.913272	-0.063579	12.4928	11371.187	-2	0	0
0:03:43	0.1241283	44.686203	-0.063075	12.4288	11358.758	-2	0	0
0:03:44	0.1235026	44.46094	-0.062573	12.3648	11346.394	-2	0	0
0:03:45	0.1228812	44.237239	-0.062139	12.3136	11334.08	-2	0	0
0:03:46	0.1222654	44.015553	-0.061579	12.2368	11321.843	-2	0	0
0:03:47	0.1216539	43.795407	-0.061152	12.1856	11309.658	-2	0	0
0:03:48	0.1210473	43.577021	-0.060663	12.1216	11297.536	-2	0	0
0:03:49	0.1204449	43.360156	-0.06024	12.0704	11285.466	-2	0	0
0:03:50	0.1198479	43.145259	-0.059694	11.9936	11273.472	-2	0	0
0:03:51	0.1192552	42.931861	-0.059277	11.9424	11261.53	-2	0	0
0:03:52	0.1186665	42.719953	-0.058863	11.8912	11249.638	-2	0	0
0:03:53	0.1180833	42.50998	-0.058326	11.8144	11237.824	-2	0	0
0:03:54	0.1175035	42.301249	-0.057981	11.776	11226.048	-2	0	0
0:03:55	0.116929	42.094433	-0.057449	11.6992	11214.349	-2	0	0
0:03:56	0.1163579	41.888842	-0.057109	11.6608	11202.688	-2	0	0
0:03:57	0.1157921	41.685144	-0.056583	11.584	11191.104	-2	0	0
0:03:58	0.1152296	41.482654	-0.056247	11.5456	11179.558	-2	0	0
0:03:59	0.1146717	41.281812	-0.055789	11.4816	11168.077	-2	0	0
0:04:00	0.1141177	41.082386	-0.055396	11.4304	11156.646	-2	0	0
0:04:01	0.1135683	40.884589	-0.054944	11.3664	11145.28	-2	0	0
0:04:02	0.1130228	40.68819	-0.054555	11.3152	11133.965	-2	0	0
0:04:03	0.1124811	40.49318	-0.054169	11.264	11122.701	-2	0	0
0:04:04	0.1119432	40.299552	-0.053786	11.2128	11111.488	-2	0	0
0:04:05	0.1114098	40.107517	-0.053343	11.1488	11100.339	-2	0	0
0:04:06	0.1108795	39.916626	-0.053025	11.1104	11089.229	-2	0	0
0:04:07	0.1103536	39.727311	-0.052587	11.0464	11078.182	-2	0	0
0:04:08	0.1098315	39.539345	-0.052213	10.9952	11067.187	-2	0	0
0:04:09	0.1093131	39.35272	-0.05184	10.944	11056.243	-2	0	0
0:04:10	0.1087984	39.167427	-0.05147	10.8928	11045.35	-2	0	0
0:04:11	0.1081886	38.947893	-0.060982	10.8416	11034.509	0	0	0
0:04:12	0.1073049	38.629772	-0.088367	10.7648	11023.744	0	0	0
0:04:13	0.0981102	35.319685	-0.919469	10.2528	11013.491	0	0	0
0:04:14	0.0889707	32.02944	-0.913957	9.344	11004.147	0	0	0
0:04:15	0.0798867	28.759223	-0.908394	8.4352	10995.712	0	0	0
0:04:16	0.0708608	25.509873	-0.902597	7.5264	10988.186	0	0	0
0:04:17	0.0618982	22.283338	-0.89626	6.6176	10981.568	0	0	0
0:04:18	0.0529668	19.068055	-0.893134	5.7344	10975.834	0	0	0
0:04:19	0.0440893	15.872141	-0.887754	4.8384	10970.995	0	0	0

0:04:20	0.0352488	12.689564	-0.884049	3.9552	10967.04	0	0	0
0:04:21	0.0264532	9.5231418	-0.879562	3.072	10963.968	0	0	0
0:04:22	0.0176676	6.3603202	-0.878562	2.2016	10961.766	0	0	0
0:04:23	0.0089833	3.2339915	-0.868425	1.3184	10960.448	0	0	0
0:04:24	0.0051242	0.1232131	-0.898331	0.4608	10959.987	0	0	0
0:04:25	0	0	0	0.0128	10960	0	0	0