

轨道交通GPS数据约简算法研究 与比较



陈德旺

北京交通大学电子信息工程学院
轨道交通控制与安全国家重点实验室

Tel: 51687111

Office: SY1107

Email: dwchen@bjtu.edu.cn



北京交通大学

报告内容



- 列车控制发展概述
- GPS介绍
- 青藏线项目介绍
- GPS数据约简模型
- 算法与比较
- 讨论

列车控制发展概述



- 参考唐涛教授的ppt

- 国家自然科学基金重点项目，“列车运行控制及组织的基础理论与关键技术研究”，2007 - 2010，项目负责人。

GPS 介绍



- 参考蔡伯根教授的ppt

- 铁道部，青藏铁路GPS数据测试与验证，2005 - 2007，项目负责人

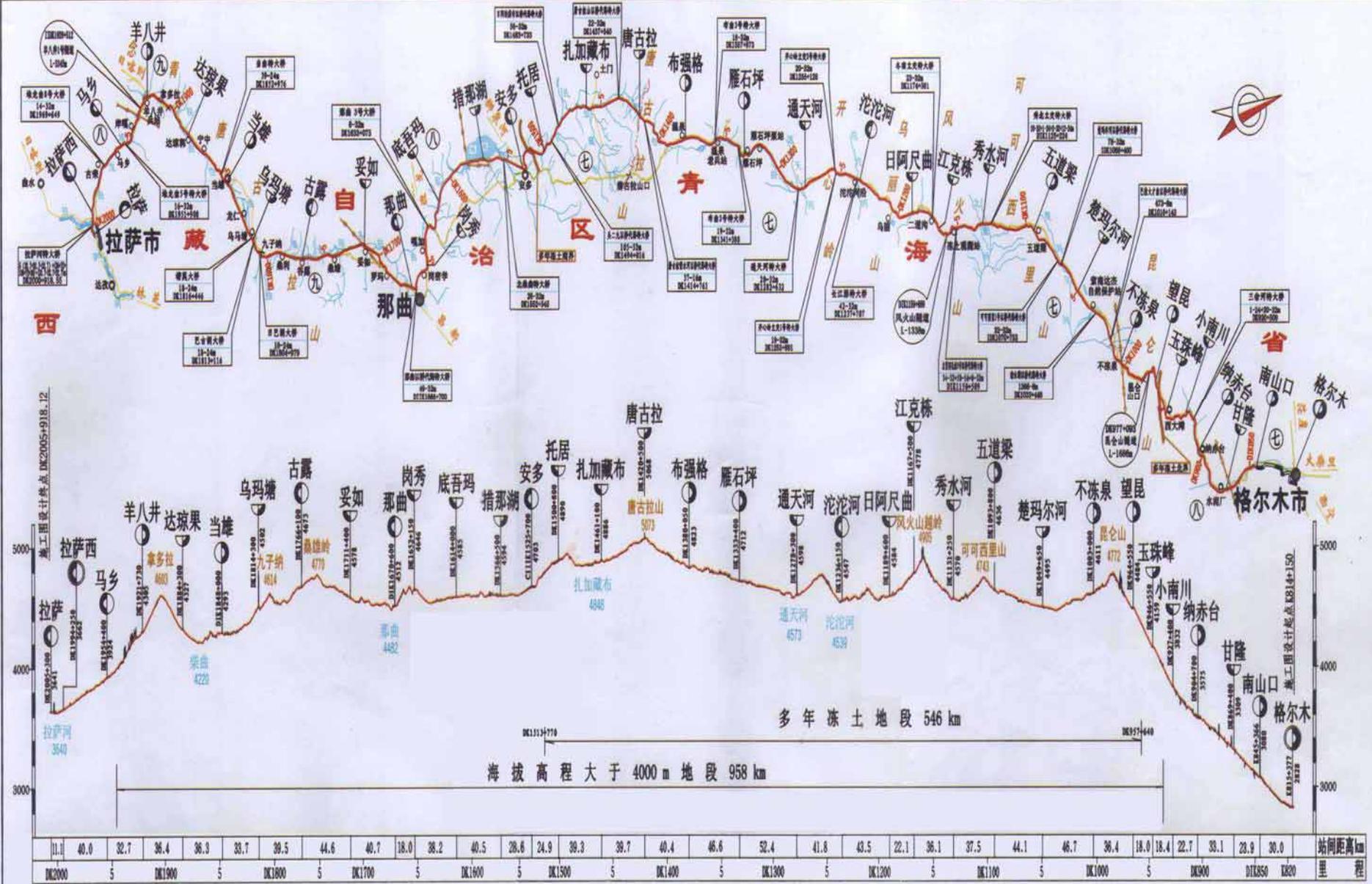
青藏铁路项目



- 青藏铁路是我国第一条采用GPS + GSM-R的铁路，第一条初步移动闭塞的铁路，也是目前我国最先进的列车控制系统。
- 我们项目组主要承担
 - 电子地图的制作
 - 所有45个站的关键点数据的地图匹配误差的验证

青藏铁路格尔木至拉萨段平纵断面缩图

平面 1:200000



铁道第一勘察设计院 2003.02



2006-11-22

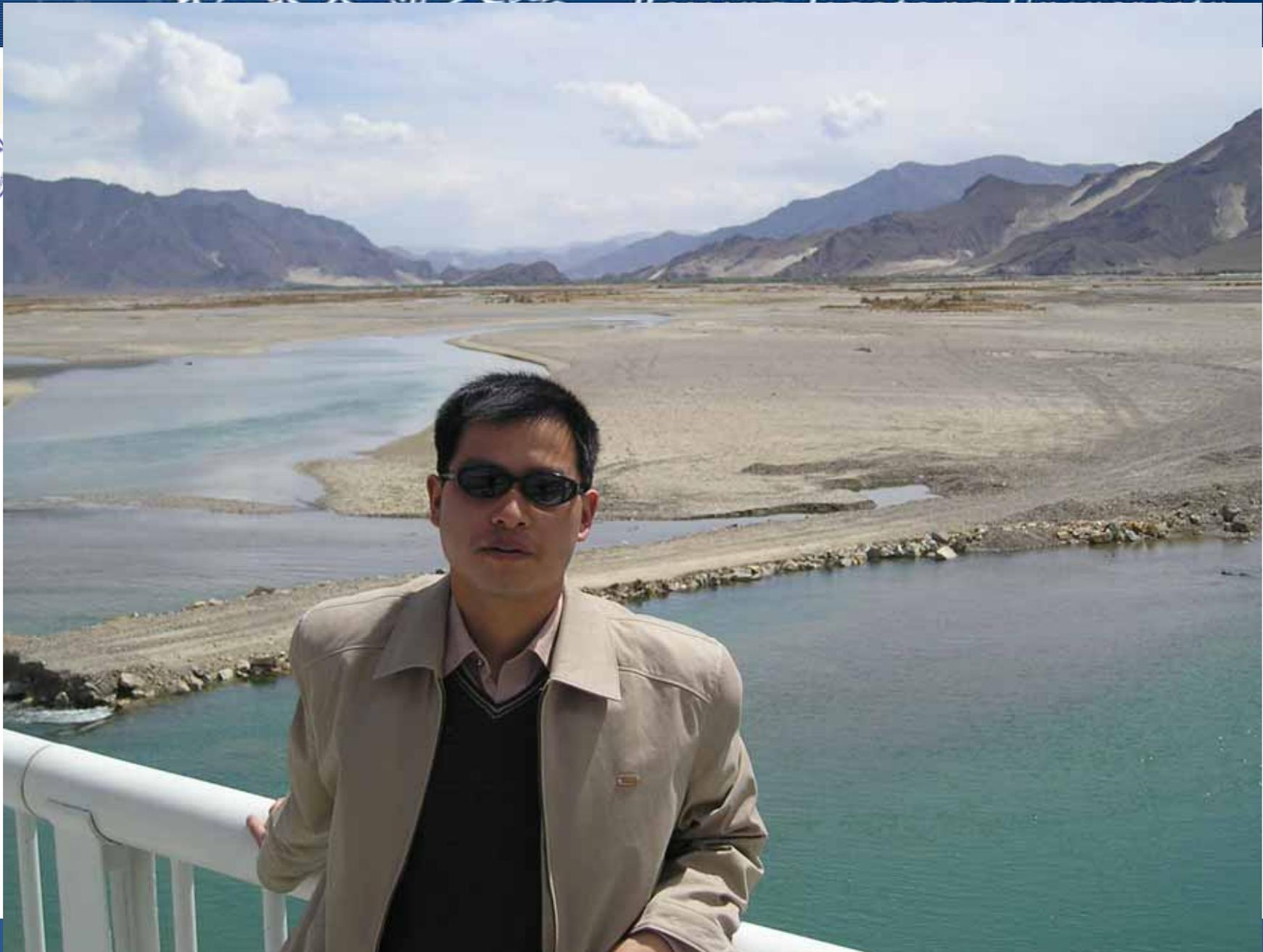
轨道交通控制与安全国家重点实验室

工作现场



青藏风光







数据约简问题



- 数据量太大，约50万组，如果不约简：
 - 存储空间太大
 - 地图匹配速度太慢
- 数据约简的要求
 - 约简率尽可能低
 - 误差满足一定的要求
 - 横向误差
 - 纵向误差

数据约简的基本思路



- 直线 + 曲线

- 多段直线

基于0 - 1规划的数据约简模型



基本假设

- 起始点为分段直线的起始点
- 分段点属于采集的数据点

$$s.t. \quad \forall d_{i,j} < \max Dist (x_i = 1, j = 1, 2, \dots, m_i) \quad (1)$$

$$s.t. \quad p_1 = beginPoint, p_n = endPoint \quad (2)$$

$$s.t. \quad x_i \in \{0, 1\}, \forall i = 2, \dots, n - 2;$$

$$\min(\sum_{i=2}^{n-1} x_i) \quad (3)$$

NP问题



- 所有的可能： $2^{(n-2)}$
 - N最大约等于50万（整条铁路）
 - N最小约等于8000（一个区间最少的数据）
- 难以在有限的时间求得最优解

基于非线性优化的模型



假设

- 起始点为分段直线的起始点
- 分段点不一定属于采集的数据点
- 连续优化问题
 - $segPoint=(segX, segY)$ 属于实数域
- 非线性优化问题
 - 点的投影公式是非线性的
- 对于增加一个优化的分段点的模型。

$$error_{1,j} = \max(error1)$$

$$(x1 = (segX, segY), j = 1, 2, \dots, m_1) \quad (1)$$

$$error_{2,j} = \max(error2)$$

$$(x1 = (segX, segY), j = 1, 2, \dots, m_2) \quad (2)$$

$$s.t. p_1 = beginPoint, p_n = endPoint \quad (3)$$

$$\max Error = \max(error1, error2); \quad (4)$$

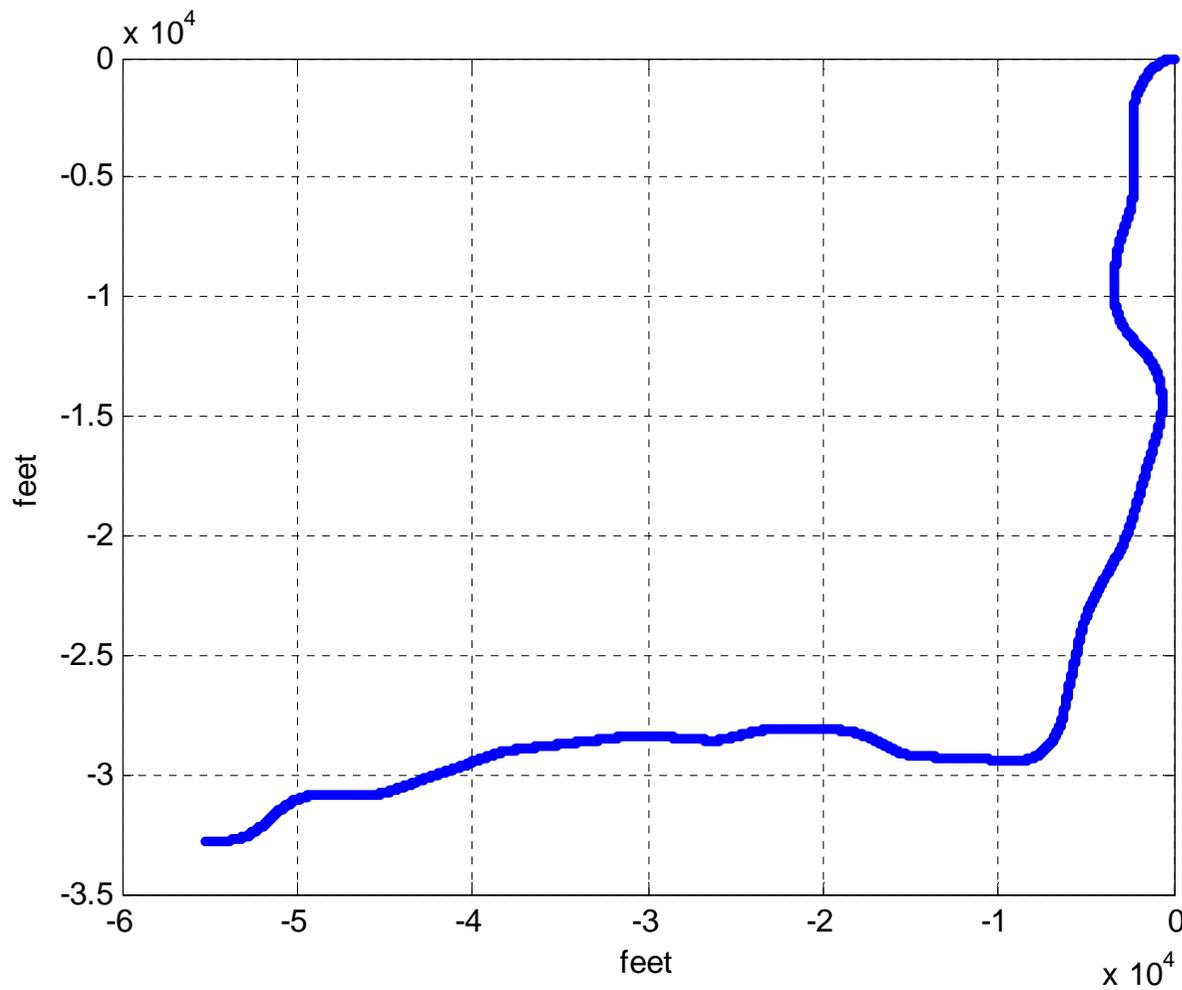
$$\min(\max Error)$$

基于非线性优化的模型

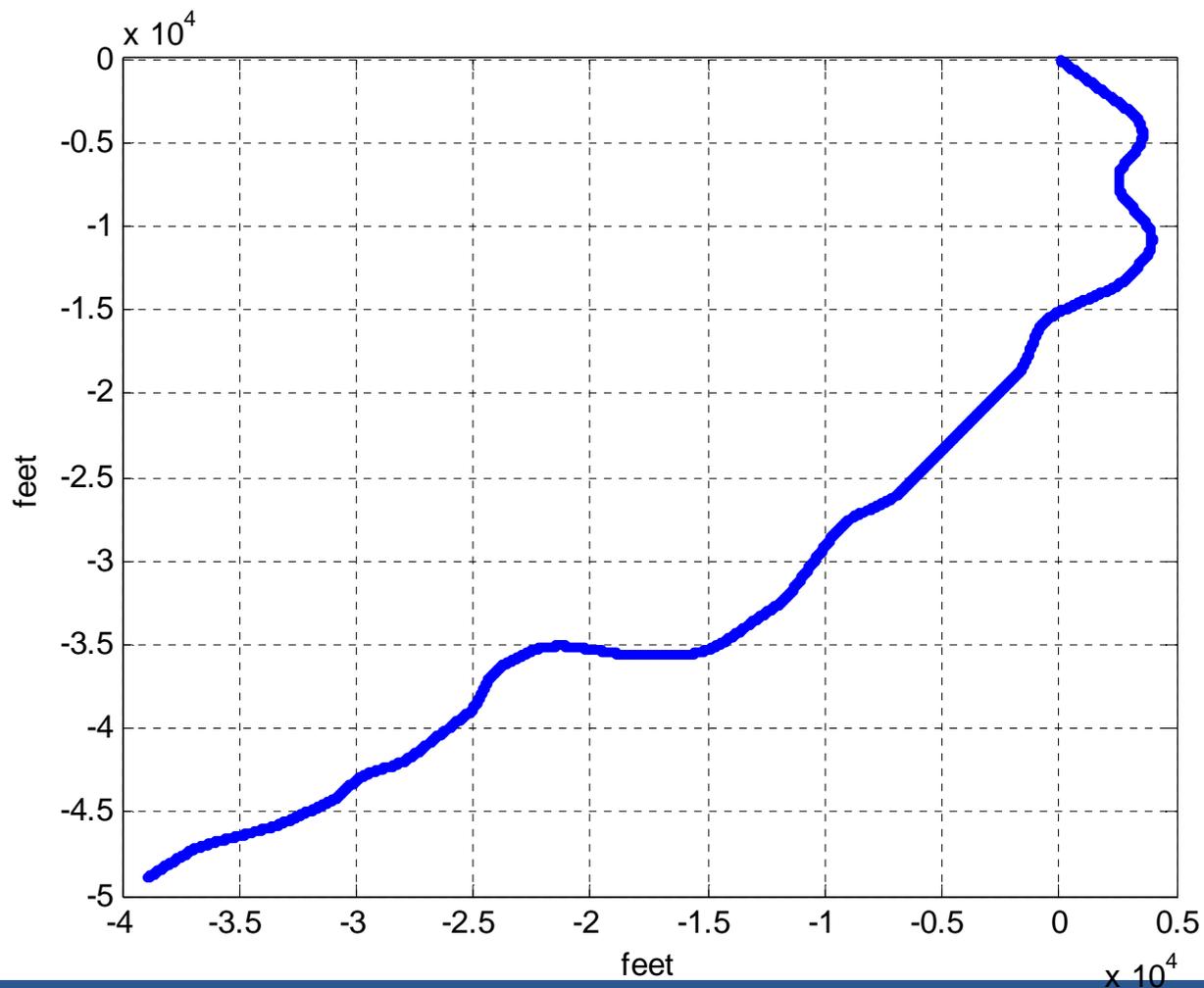
优化目标：

- 每增加一个优化点，使得最大的误差最小
- 在满足误差要求的前提下，优化点最少。
- 求最少的分段：连续和离散的混合问题
 - Step1: $N=1$
 - Step2: 增加一个优化点
 - 可把投影距离最大值点作为优化的起始点。
 - Step3: $N = N + 1$;
 - Step4: 最大投影误差 $\max \text{Error} < \text{setValue}$ ，结束；否则 goto Step2

数据描述：某铁路站1 (9935组数据)



数据描述：某铁路站2 (8452组数据)



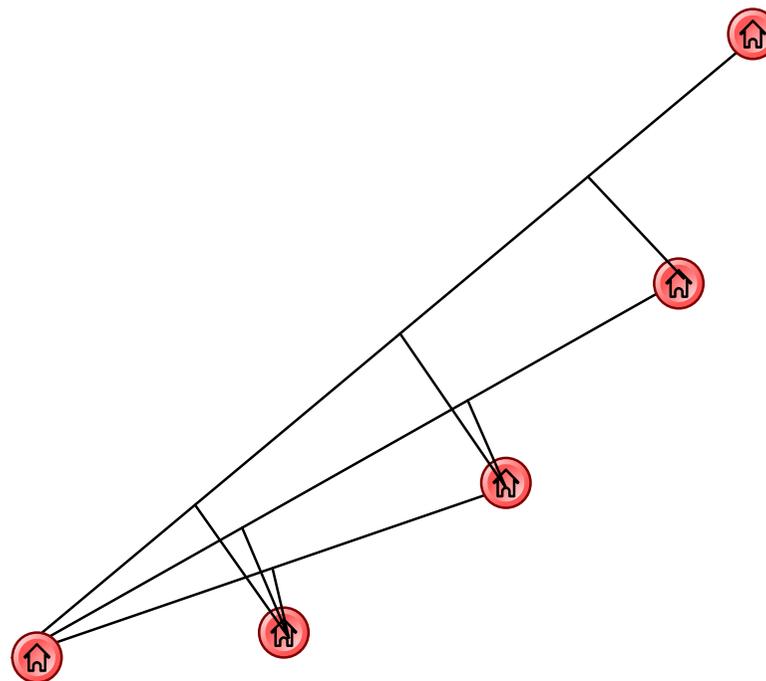
启发式算法1：试探法

基本原则：

- 尽可能前进
- 不能前进则后退一步
- 以终为始，继续前进，直到所有的点包括在内。

$$d_i = \frac{|kx_i + b - y_i|}{\sqrt{k^2 + 1}}$$

$$\max D = \max(d_i)$$



优缺点

■ 优点：

- 算法简单，易于理解。
- 编程实现方便。

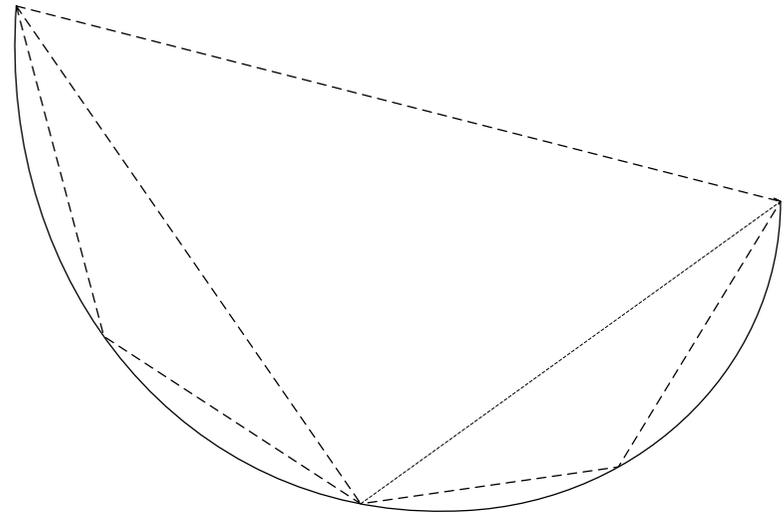
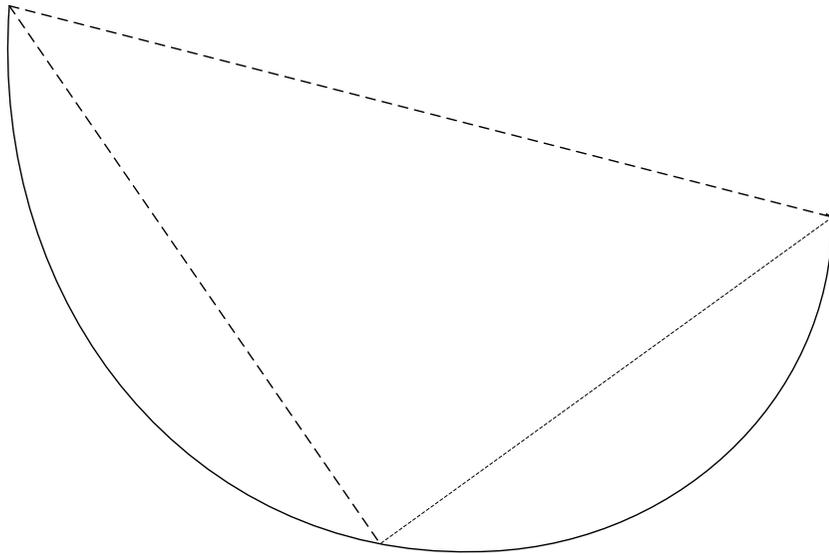
■ 缺点：

- 局部优化，没有考虑到数据的整体结构。
- 如果数据是锯齿形，会造成分段数太多。

启发式算法2：二分法

基本思路：

- 利用二叉树结构，不断分裂，
- 直到每一部分都不需要分裂



起点

优缺点

■ 优点：

- 与二分法类似，可借用二叉树的结构
- 考虑到数据的整体特性，避免锯齿形问题。

■ 缺点

- 分裂后，分裂点不再改变，显然不是最优解
- 需要较大的空间来存储二叉树的结构



算法的性能指标

分段数 m ：分段直线的总数，越少越好。

- 数据约简率 $reductionRate(\%)$ ：关键点数和所有数据点数 n 之比，反映了数据约简的效率，越小越好。实际上，该指标与指标1密切相关，因为关键点数等于分段数加一。

$$reductionRate = \frac{m+1}{n} * 100\%$$

- 纵向误差 $longthError(\%)$ ：1减去分段直线的长度之和和所有点首尾相连的长度之和的比，反映了直线拟合曲线在长度上的损失，越小越好。

$$LongthError = \left(1 - \frac{\sum_{i=1}^m k_i}{\sum_{j=1}^{n-1} l_j}\right) * 100\%$$

算法的性能指标



- 横向误差的平均值：所有点到相应直线段投影距离的平均值，越小表明算法的鲁棒性越好。
- 横向误差的最大值：所有点到相应直线段投影距离的最大值，检验算法是否满足横向误差要求，同时也间接表明算法的鲁棒性，越小越好。
- 运行时间：反映算法的时间上的效率，越小越好。

算法运行效果



- 程序演示1
- 程序演示2

算法运行效果对比 (区间1)

区间1(9935组数据)	算法1		算法2	
	1	2	1	2
横向误差要求 (ft.)	1	2	1	2
分段数	228	153	315	212
约简率(%)	2.30	1.55	3.18	2.14
纵向误差(%)	0.0047	0.0086	0.0025	0.0057
横向误差的最大值(ft)	0.9995	1.998	0.9940	1.979f
横向误差的平均值(ft)	0.55	1.134	0.2553	0.5927
运行时间(s)	8.292	8.542	33.388	29.873

算法运行效果对比 (区间2)

区间2 (8452组数据)	算法1		算法2	
	1	2	1	2
横向误差要求 (ft.)	1	2	1	2
分段数	233	167	327	230
约简率(%)	2.77	1.9877	3.8807	2.7331
纵向误差(%)	0.0048	0.0097776	0.0028	0.0055
横向误差的最大值(ft)	0.999758	1.99988	0.9994	1.9880
横向误差的平均值(ft)	0.542445	1.08519	0.2847	0.5464
运行时间(s)	5.738	5.038	28.497	26.117

算法效果比较分析 (1)

■ 算法1在分段数、约简率方面要优于算法2

■ 算法2在纵向误差、横向误差的最大值、横向误差的平均值等指标方面要优于算法1

■ 随着误差要求的增大，

- 算法1和算法2的分段数都会减少，约简率都会增加，
- 纵向误差、横向误差的最大值、横向误差的平均值等三个性能指标会增大。

算法效果比较分析（2）



- 因为区间2的数据较为弯曲，在相同的误差要求和算法的情况下，区间2的数据的分段数和约简率要大于区间1的数据的分段数和约简率，其他三项指标则没有明显的区别。
- 两种算法的运行时间都较短，算法效率都比较高。相对而言，算法1速度较快。不同误差要求对算法的速度影响较小，不超过10%。

正在进行的工作



- 三维数据
- 电子地图格式
- 列车定位的快速匹配算法
- 列车行驶里程的快速计算

讨论

- 能否利用组合优化，求得模型1情况下的最优解？
- 能否利用非线性优化，求得模型2情况下的最优解？
- 最优的算法也许需要太长的时间？
- 能否比启发式算法更好？