

# 运筹学 Q&A 专场

顾宗浩博士

GUROBI CTO, 联合创始人



**GUROBI**  
OPTIMIZATION

The World's Fastest Solver

## 顾宗浩 博士

- Gurobi 首席技术官 (CTO) 和联合创始人
- 上海同济大学获机械工程学士和工业管理硕士，在佐治亚理工学院获工业工程博士
- 曾担任高级软件开发人员，后来担任ILOG CPLEX研发团队的杰出科学家和架构师
- 在线性，整数和二次规划计算方面世界最领先的专家之一



# 前期问题中的热点问题

- 精确算法和启发式算法如何融合？
- Gurobi 如何识别特殊的模型结构？
- 如何建立数学模型可以帮助优化器更快求解？
- 数学优化和机器学习如何结合？
- 模型优化速度和哪些软硬件环境因素有关？
- Gurobi 非线性优化的计划？
- 算法理论和工程化为什么效果上有差异？
- 用 GPU 求解数学规划的理论难点是什么？开发难点是什么？
- 整数规划中的灵敏度分析。是否有行业统一定义和计算方法？
- 当边界移动缓慢或者无法移动时，除了设置参数之外，还有什么其他方法？
- Gurobi 优化时分支策略有哪些？分支后如何选择哪个节点优先探索？让用户选择分支策略是否具备更大的速度优势？

**请用右边控制面板中的 Question 面板来继续提问**

# 精确算法和启发式算法如何融合?

# 精确算法和启发式算法如何融合?

## 两种解决策问题的 ( 优化 ) 方法

- 启发式算法
- 优化(精确)算法

## 优化算法

- 最佳解决方案VS 某些 ( 可能质量很差 ) 解决方案
  - 达到最高效率
  - 不可行的问题 ( 启发式方法不能断定这是不可行的 )
- 易于维护
  - 不断变化的业务条件
  - 启发式方法通常是针对特定问题的 , 条件的改变很容易使启发式方法无效

## 参考

- <https://www.gurobi.com/resource/4-key-advantages-of-using-mathematical-optimization-instead-of-heuristics/>

## 用启发式方法来加强和改进优化算法

- Gurobi 内部大约有40 种启发式算法
- 客户输入
  - MIP初始解 (MIP start)
  - MIP提示(MIP varhint)
  - 分区启发式
  - 启发式回调
- 提供最优解
  - 证明最优提速约2倍

## 用优化算法改进启发式解决方案

- 问题太大，无法整体优化
  - 分区，分时，分段优化
  - 用优化向前滚动 (rolling forward)启发式
  - 用优化邻域搜索 (local search)启发式
  - 等等
- 给出启发式解，你总可以通过优化部分固定的问题来改进启发式解, 例如
  - RINS 方法

## Gurobi 如何识别特殊的模型结构?

# Gurobi 如何识别特殊的模型结构?

我们的求解器主要是黑匣子求解器

我们只识别一些常见的结构

- 约束类型: 纯0-1, 混合, 等等
- 集团(cliques or GUB)
- VUB
- 网络约束
- 等等

**如何建立数学模型可以帮助优化器更快求解？**

# 如何建立数学模型可以帮助优化器更快求解?

## 如何建立数学模型的参考书

- H.P. Williams: Model Building in Mathematical Programming
  - 所有模型都有Gurobi 对应的 Jupyter Notebook 例子
  - <https://www.gurobi.com/resource/hp-williams-modeling-examples>

## Gurobi 提供了很多材料

- 例子: <https://www.gurobi.com/resources/?category-filter=code-example>
- 培训材料

# 如何建立数学模型可以帮助优化器更快求解？

## 避免数字麻烦

- 矩阵：系数不要太大/太小
- 变量界限, 右侧项, 目标系数:不要太大
- 避免接近平行的约束
- 等等

## MIP 问题

- 同一问题通常可以有几种不同的模型
- 较小的模型是否好？
  - 也许，但不一定
- 通常较小的松弛可行区域更好，通常MIP Gap较小，通常叫tighter formulation
- 平衡强度(strong or tight)和模型尺寸

## 数学优化和机器学习如何结合？

# 数学优化(MO)和机器学习(ML)如何结合?

机器学习主要用于预测, 分类, 识别

数学优化主要用于复杂的商业决策问题

完美的AI技术团队 ( ML和MO ), 两者结合

- Forrester Opportunity Snapshot, <https://www.gurobi.com/resources/?category-filter=knowledge-center>
- 使用ML进行更准确的预测, 确定
  - 目标系数, 像产品价格
  - 右侧项, 像产品需求
  - 等等
- 使用MO 找到最佳或良好方案

# 数学优化和机器学习如何结合？

## 在MIP算法中使用ML

- 有一些研究报告，例如关于分支变量的选择，很少成功的例子

## 在ML算法中使用MIP

- 回归是ML的重要方法，它是一个优化问题
- 有更多研究报告和更多成功例子

## 使用ML 直接解商业决策问题

- 用作启发式算法
  - ML解TSP问题
  - 只能对很小的TSP问题找到相对不错的可行解

## Gurobi 客户调查

- 大多数项目使用机器学习提高数据精度，再输入优化模型
- 几乎一半的Gurobi用户看到公司目前用ML解的问题可以用MO更好地解决

**模型优化速度和哪些软硬件环境因素有关？**

# 模型优化速度和哪些软硬件环境因素有关？

操作系统，例如Windows，Linux或Mac，影响不大

计算机语言或API

- Gurobi OO API非常轻便，为了提高效率，将引擎（算法）部分用C语言编写
- 与建模部分相比，大多数情况在引擎（优化）部分花费的时间要多得多

电脑芯片(CPU GHz )

- 速度越高越好
- MIP: 在32核以下的内核越多越好
- 缓存通道: 越多越好, 越快越好
- 新一代芯片通常会更快

分布式MIP求解器

- 适用于MIP模型需要大量的B & B节点来解

## Gurobi 非线性优化的计划?

# Gurobi 非线性优化的计划?

改进非凸MIQCP求解器

添加非凸QP或通用NLP连续求解器

- 对于找到非凸MIQCP或MINLP的良好可行解很重要

通用MINLP求解器

## 算法理论和工程化为什么效果上有差异？

# 算法理论和工程化为什么效果上有差异？

理论算法考虑在最坏的情况下计算复杂度

- LP 例子：比较单纯形法和椭球法
  - 椭球法: 多项式，但是实践非常慢，总是接近最坏的理论情况
  - 单纯形法：最坏的情况不是多项式，但实际上迭代次数与列和行的总数之和基本上是线性的

取决于实施(Implementation)

- Gurobi LP单纯形算法优求解器比LP Solver 求解器快得多

数值处理的好坏

用 GPU 求解数学规划的理论难点是什么？ 开发难点是什么？

# 用 GPU 求解数学规划的理论难点是什么？开发难点是什么？



## LP单纯形法

- 瓶颈是内存访问(memory access), 难以并行化

## MIP分支定界算法

- 解许多节点松弛问题
- 但每个节点松弛问题对于GPU并行性而言都足够不同

有一些研究, 但目前为止还没有成功报告

整数规划中的灵敏度分析。是否有行业统一定义和计算方法?

# 整数规划中的灵敏度分析。是否有行业统一定义和计算方法？



## 定义

- 线性规划灵敏度分析: 运筹学教科书通常有定义
  - 系数( $c$ , rhs, lb, ub)单位改变造成目标值的改变
  - 假定最优基不变, 是近似的, 主要考虑计算速度
- 整数线性规划灵敏度分析
  - 系数( $c$ , rhs, lb, ub)单位改变造成目标值的改变

## Naive(天真的)算法

- 对每个系数, 求解一个MIP

## 我们的方法 (Gurobi 9.0)

- 求解一个大模型以获取所有灵敏度数值

当边界移动缓慢或者无法移动时，除了设置参数之外，还有什么其他方法？

当边界移动缓慢或者无法移动时，除了设置参数之外，还有什么其他方法？



用调整工具(tuning tool), 找更好的参数

如果目标下限移动太慢，考虑割平面回调

如果当前最佳可行解改善太慢，考虑

- MIP初始解 (MIP start)
- MIP提示(MIP varhint)
- 分区启发式
- 启发式回调

考虑重新建模

Gurobi 优化时分支策略有哪些？  
分支后如何选择哪个节点优先探索？  
让用户选择分支策略是否具备更大的速度优势？

# Gurobi 优化时分支策略有哪些？分支后如何选择哪个节点优先探索？ 让用户选择分支策略是否具备更大的速度优势？



## 分支策略

- 分数值最大的整数变量(Integer variable with most fractional value)
  - 通常非常糟糕
- Strong branch
  - 很花时
- Pseudo cost
  - 接近默认值
- Pseudo reduced cost

## 节点选择

- BranchDir参数

## 用户选择分支策略

- 分支优先(Branch priority)

谢谢!  
MORE Q&A



**GUROBI**  
OPTIMIZATION

The World's Fastest Solver

## 下一步

- 立即试用 Gurobi 9.0 !
  - 获得 Gurobi 30天商业试用许可，请联系 [help@gurobi.cn](mailto:help@gurobi.cn)
  - 学术和研究许可是免费的
- 有关Gurobi 销售问题，请联系： [help@gurobi.cn](mailto:help@gurobi.cn)
- 要了解我们的下一个网络研讨会会议，请访问：[www.gurobi.com/events](http://www.gurobi.com/events)