1. 算法原理
2. 启发式搜索

以3个整数变量为例，随机（还是按照一定方法？）生成一个现任最优解incumbnet：000，通过松弛该解的大量变量可以生成新的树，松弛该解的少量变量可以生成新的整数解，每棵新的树对应一个可能的下界Dual Bound，每个新的整数解对应一个可能的上界Primal Bound。比如通过将incumbent000中的x1，x2松弛后做LP优化，可以得到结点xx0的下界值，再计算出xx1的下界值后取最小值即得到root1的下界，若该下界比当前已知的全局下界要大，则更新全局下界为最新下界并报告New Dual Bound。同理root2也能得到一个Dual Bound，但如果这个值比全局的Dual Bound还小，则不会更新全局下界。有时松弛较少的变量可能更够直接得到整数解，比如通过将incumbent000中的x1松弛后做LP优化，可能得到一个与原来incumbent000一样的解000，这样的话incumbent不会更新；将incumbent000中的x2松弛后做LP优化，可能出现x2任不为整数，则会生成新的树或补充一些已有树的结点；将incumbent000中的x3松弛后做LP优化，可能出现出现一个新的incubment，这样的话则更新全局上界为最新上界并报告New Primal Bound。



图1

1. 松弛解诱导领域搜索

该算法其实就是让当前的最优解incumbent来知道当前结点应该怎么进行下一步的松弛，以8个整数变量为例，再当前节点得到的松弛解为0xxx101x，而当前的最优解为00001111，那么则将两则相同的部分固定下来作为常数，松弛剩余不相同的部分，形成一个新的结点。因此x1,x5和x7被固定作为常数，再对剩余5个变量的子问题进行优化，这样一次分枝就固定了3个变量，分枝速度快了很多。



图2

1. 加入割平面

割平面是一个过整数点的约束，比如x1+x2>=1就是一个割平面。添加割平面能剪去很多枝从而使搜索空间减小而加速下界的收敛，比如通过添加上述割平面后，图1中00x的分枝就被剪去，使整颗树变小，下界也可能因此上升。



图3

1. 分布式算法

分布式算法主要利用了上述的一些算法，调度多台机器从多个起始点做启发式搜索，试图快速收敛上界，另外调度一台机器大量加入割平面，试图快速收敛下界，以所有工作机的最小上界为全局上界，以最大下界为全局下界，并在全局上下界达到一定gap值后开始从产生当前全局最优解的树开始剪枝。这样启发式搜索的力度将会被提高。



图4

缺点：该方法有一个缺点，是我一直试图解决的重点。由于每台工作机的搜索空间使相互独立的，它们不会知道自己搜索的结点是否已经被其它机器搜索过，因此会出现重复搜索的问题，降低整个算法的效率。如采用4机分布式计算时虽然800%的CPU满载使用率，但真正左右用功而不是重复搜索的使用率绝对达不到这个值，但目前还无法计算这个效率，设法计算这个效率使进行改善的第一步，也是下一步的工作。

1. 实现架构

该分布式并行计算架构可以以网络中任何一台PC端作为主机，所有的服务器作为工作从机。对于服务器端，愿意提供优化计算资源的工作从机只需要调用一个后台进程，该进程在后台实时监听网络中的主机，一旦有主机发布优化任务，它立刻在后台为主机提供部分计算资源进行分布式计算。该过程对于其他服务器用户甚至服务器管理员均是透明的，不会影响服务器上的其他任何操作（除非服务器资源被耗尽）。当某台服务器不希望继续作为工作从机提供计算资源时，只需简单地输入命令便可退出监听，并且该过程对于其他用户也是透明的。

对于PC端，其作为主机可以保持设置CPLEX求解参数的功能以及随时发布优化任务，所发布的任务可被所有加入了分布式并行计算的工作机所监听到。此后主机再通过TCPIP协议将问题通过网络传输到各工作机上。PC端可以通过回调从所有服务器上获取所有参与分布式并行进程中输出的实时求解信息，与当前主机上的全局求解信息进行比较，从而了解各工作从机的求解状态，以便实现进一步的调度或终止优化。



图5

1. 项目组成

distributed\_work ╦

║ 主机文件

╠ master ╦ bin：主机可执行文件与cplex动态库

║ ║

║ ╠ data：待求解的lp文件存放目录

║ ║

║ ╠ include：编译所需头文件存放目录

║ ║

║ ╠ lib：CPLEX静态库存放目录

║ ║

║ ╠ log：求解过程中记录的log存放目录

║ ║

║ ╠ makedir：Makefile的辅助文件存放目录，

║ ║ 请在这修改求解参数

║ ╠ Makefile

║ ║

║ ╠ obj：中间目标文件存放目录

║ ║

║ ╠ sol：mip问题解文件存放目录

║ ║

║ ╚ src：主机程序源文件存放目录

║ 工作机文件

╠ client ╦ bin：cplex可执行文件与cplex动态库，

║ ║ 分布式回调函数动态库

║ ╠ include：编译库所需头文件存放目录

║ ║

║ ╠ lib：CPLEX静态库存放目录

║ ║

║ ╠ makedir：Makefile的辅助文件存放目录，

║ ║ 请现在这修改端口

║ ╠ Makefile

║ ║

║ ╠ obj：中间目标文件存放目录

║ ║

║ ╚ src：工作机动态库源文件存放目录

1. 主程序流程



图6