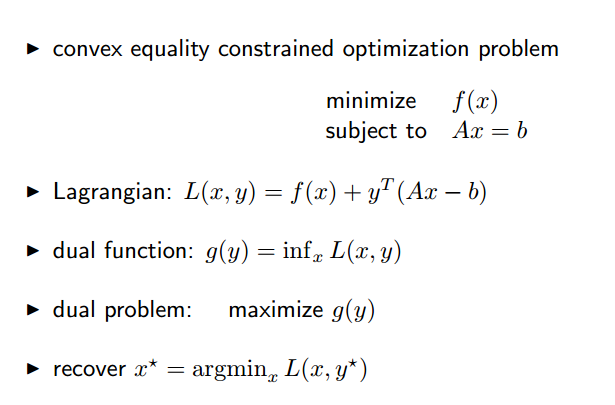
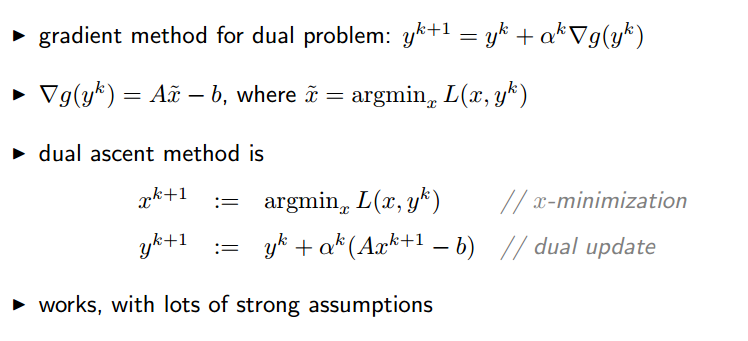
Alternating Direction Method of Multipliers

Prof S. Boyd

# 1. 常规拉格朗日松弛



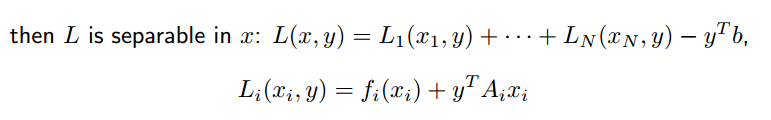


（可以理解为什么对偶问题还是min：因为对偶问题应该是max（lamda）min（x），因为lamda在这里是先确定了，所以只剩下min（x），所以看起来就是原问题直接将约束条件对偶上去，而min这个符号不改变）

（1）将约束条件放至目标函数上，引入乘子y，通过梯度法迭代乘子y，通过每步迭代中给定y从而迭代更新变量x，最终k步后求得最优解。

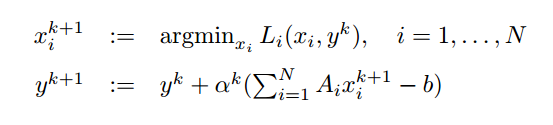
（2）常规拉格朗日松弛可分解性：

对于常规拉格朗日松弛的f，可拆解时，将LR问题分解成多个子问题，



可以实现并行计算。

（3）子问题（各个子问题独立）中，迭代方式：



（4）常规拉格朗日松弛优点：

1、可平行计算子问题

2、梯度法更新乘子，保证协调性

（5）常规拉格朗日松弛缺点：计算通常很慢并且有很多的假设

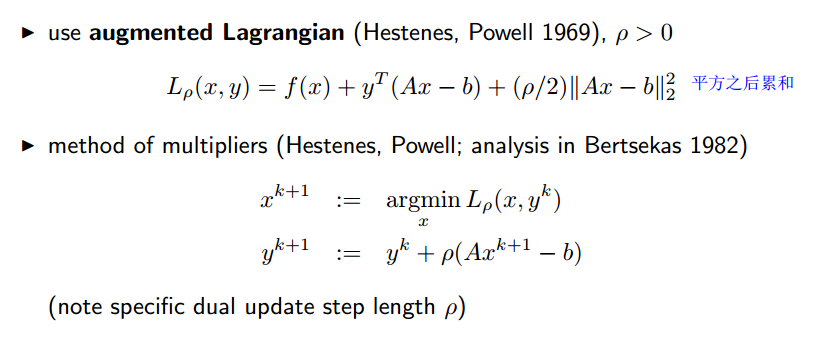
# 2.增广拉格朗日方法

增广拉格朗日方法在常规拉格朗日方法的基础上添加了二次惩罚项，从而使得转换后的问题能够更快求解，也使得收敛的条件不那么苛刻。缺点是因为加了二次项，就不能像常规拉格朗日松弛一样分解成子问题并行计算了（quadratic penalty destroys splitting of the x-update, so can’t do decomposition）。

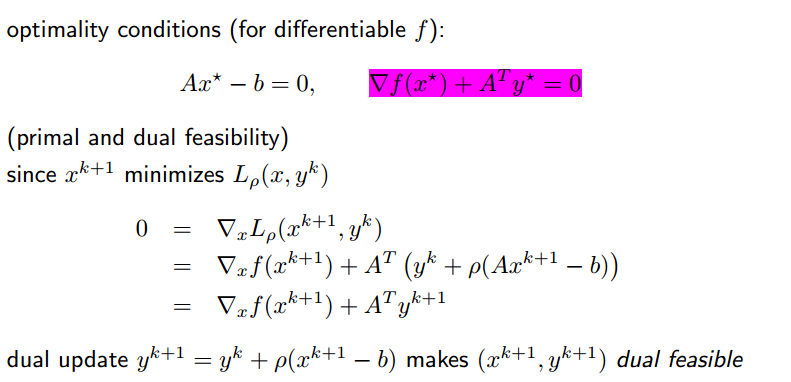
加二次项的作用：

To yield convergence without assumptions like strict convexity or finiteness of f.

使得没有严格凸或者有限假设的f也可以适用，这种方法具有更加优越的收敛性质。

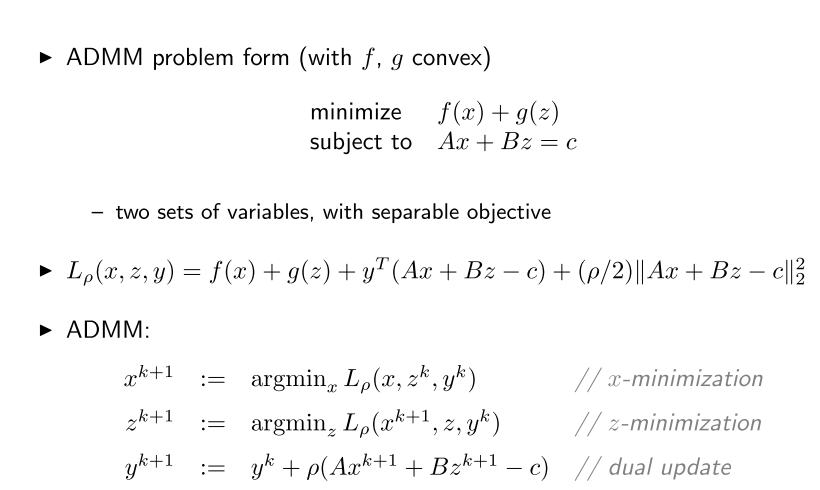


乘子的迭代方式同时也保证了对偶问题的可行性，推导如下：



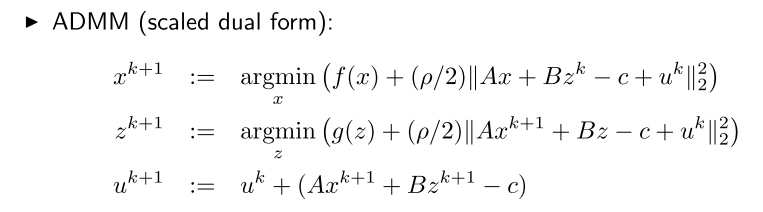
# 3.ADMM

又可分解，又可保证对偶可行性



在求解时，先求x（z，y固定），再求z（x,y固定），然后求y（x，z固定）。即交替

改写后：u=y/p



这样就把x和z就拆分开了。（自己没能成功将u加入二次项中）

补充知识：拉格朗日松弛原问题、对偶问题。

<https://www.cnblogs.com/90zeng/p/Lagrange_duality.html>

初始问题是求min问题，对偶问题是对拉格朗日函数先取最小化，再取最大化；而原问题则是对拉格朗日函数先取最大化，再取最小化。

则原问题函数是min（x）max（a,b）f(x,a,b)，所以原问题是：max（a,b）f(x,a,b)

则对偶问题函数是max（a,b）min（x）f(x,a,b)，所以对偶问题是：min（x）f(x,a,b)