Lagrange Theory
– *Theoretical advices ...*

Thomas Stidsen

thst@man.dtu.dk

DTU-Management
Technical University of Denmark

Outline

- Preliminar definitions
- The Lagrangian dual
- The bounding ability (prop. 12.8)
- The integral property (Corollary 12.9)
- Piecewise linearity Lagrangian dual
- Subgradient optimisation

This lecture is based on Kipp Martin 12.1, 12.2, 12.3, 12.4, 12.5 (only 12.5.1)

Requirements for relaxation

The program:

$min\{f(x)|x \in \Gamma \subseteq \mathcal{R}^n\}$

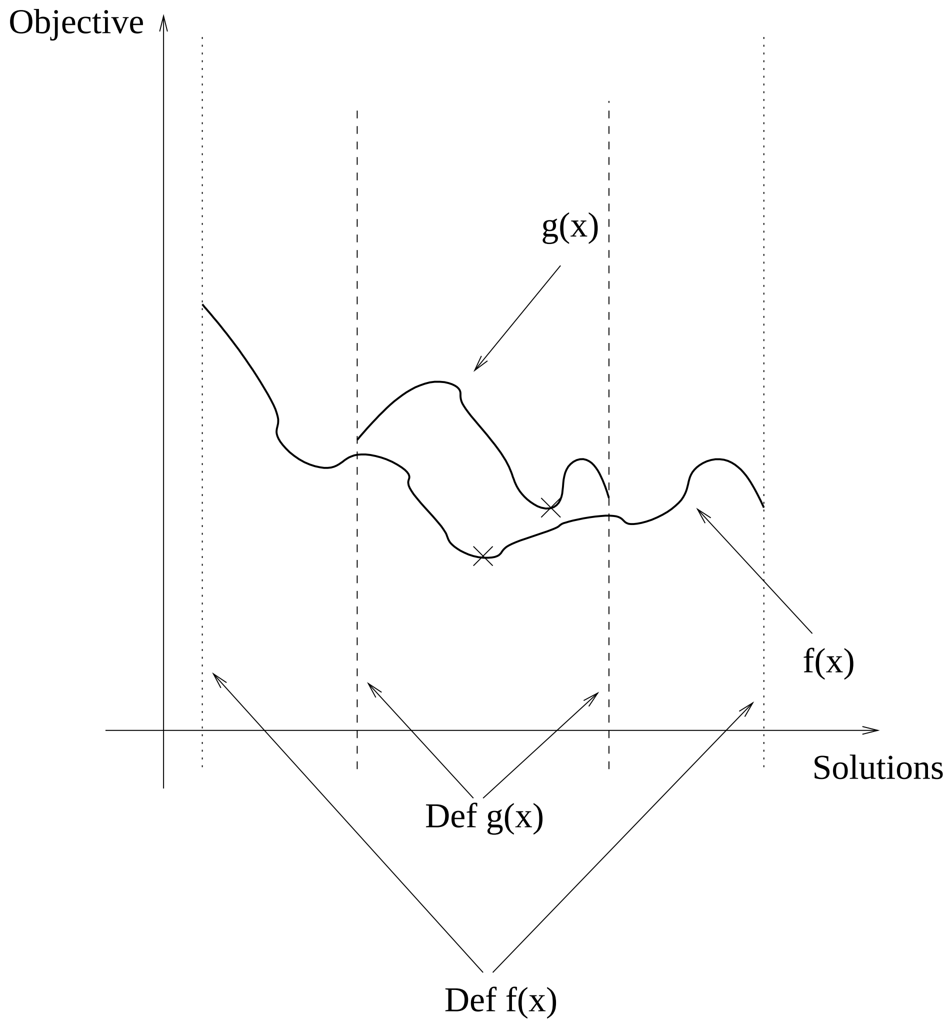
is a relaxation of:

$min\{g(x)|x \in \Gamma' \subseteq \mathcal{R}^n\}$

if:

- $\Gamma' \subseteq \Gamma$
- For $x \in \Gamma' \ : \ f(x) \leq g(x)$

Relaxation Graphically



The problem (LP)

Min:

$$c^T x$$

s.t.:

$$\begin{aligned} Ax &\geq b \\ Bx &\geq d \\ x &\geq 0 \end{aligned}$$

Lagrangian correspondence to Dantzig-Wolfe

See (p. 394-396):

- Given an LP problem (LP)
- Dantzig-Wolfe decomposition can be performed (DW)
- Create the Dual Dantzig-Wolfe decomposition (DDW)
- By a number of re-formulations the Lagrangian dual L(u) is created.

$$opt(LP) = opt(DW) = opt(DDW) = opt(L(u))$$

A few more details I

Any LP can be Dantzig-Wolfe decomposed. In RKM the Dantzig-Wolfe decomposed version (DW) includes the extreme rays. RKM talks about inverse projection. Ignore this comment. Inverse projections is used in RKM to develop the Dantzig-Wolfe decomposition, but I find in unnecessary complex.

A few more details II

Then DW is dualized, getting the Dual DW (DDW). Any LP can be dualized so no limitations here.

A few more details III

This is where it gets a bit tricky. First RKM use the structure of the DDW to "conclude" the value of u_0. Then RKM performs an *INVERSE* Dantzig-Wolfe decomposition, restoring the original matrices. He then ends up with ...

The Lagrangian dual

We define the Lagrangian dual of the problem (LP):

$$L(u) := min\{(c - A^T u)^T x + b^T u | Bx \geq d, x \geq 0\}$$

In $L(u)$, the u correspond to the Lagrangian Multiplier λ from last weeks lecture.

Proposition 12.1

From the Dantzig-Wolfe correspondence to Lagrangian bounding we immediately get:

$$\text{(LP): } min\{c^T x | Ax \geq b, Bx \geq d, x \geq 0\}$$
$$=$$
$$\text{(L(u)): } max\{L(u) | u \geq 0\}$$

This comes directly because of the direct correspondence between (LP) and (L(u)), and the optimal dual value u is equal to the corresponding dual value for the constraint. Interesting, lets test that ...

Lagrangian Relaxation of Set Cover

We use the program from the last time, but only look at the dual values and the Lagrangian multipliers.

But what about integer programs ?

Min:

$$c^T x$$

s.t.:

$$\begin{aligned} Ax &\geq b \\ Bx &\geq d \\ x &\geq 0 \end{aligned}$$

$$x_j \in \mathcal{Z}, j \in I$$

The Gamma domains

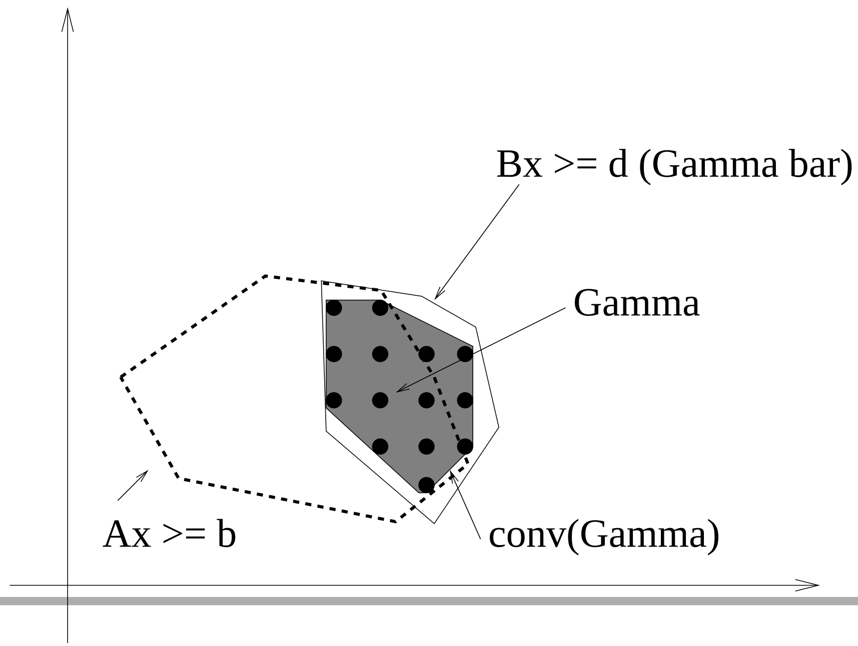
In Kipp Martin (p. 398):

$\Gamma := \{x \in \mathcal{R}^n | Bx \geq d, x_j \in \mathcal{Z}, j \in I\}$

In Kipp Martin (p. 401):

$\overline{\Gamma} := \{x \in \mathcal{R}^n | Bx \geq d, x \geq 0\}$

$conv(\Gamma)$: The convex hull of the integer points



Proposition 12.5

What happens if we take the convex hull relaxation of the MIP problem ?

$$\text{(LP): } min\{c^T x | Ax \geq b, x \in conv(\Gamma)\}$$

$$=$$

$$\text{(L(u)): } max\{L(u) | u \geq 0\}$$

This means that solving the convex relaxation, can

equivalently be achieved through optimisation of the Lagrangian program.

Proposition 12.8

This is the most significant proposition in chapter 12:

$$min\{c^T x | Ax \geq b, x \in \overline{\Gamma}\}$$
$$\leq$$
$$min\{c^T x | Ax \geq b, x \in conv(\Gamma)\}$$
$$=$$
$$max\{L(u) | u \geq 0\}$$
$$\leq \quad \longleftarrow \text{ HERE IS THE DIFFERENCE}$$
$$min\{c^T x | Ax \geq b, x \in \Gamma\}$$

Proposition 12.8, II

This is the most significant proposition in chapter 12:

- The Lagrangian relaxation bounds the original optimization problem (naturally) (12.18 and 12.19)
- The Lagrangian relaxation bound is **possibly** better than the straight forward linear relaxation (12.16, 12.17 and 12.18)

Proposition 12.8, III

The better bounding possibility explains the importance of Lagrangian relaxation. The question is now: When is the Lagrangian relaxation bound better than the LP bound ?

Corollary 12.9 (Integrality Property)

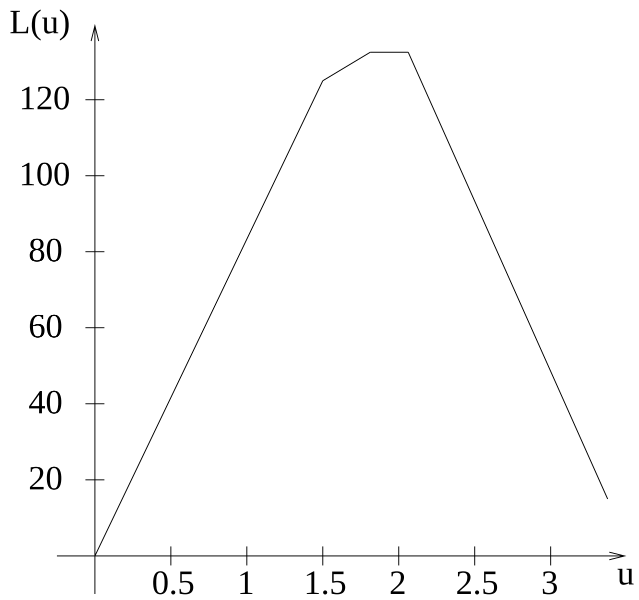
If the relaxed program has integrality property, the $\geq$ is changed to $=$:

$$min\{b^T u + (c - A^T u)^T x | x \in \Gamma\}$$
$$=$$
$$min\{b^T u + (c - A^T u)^T x | x \in \overline{\Gamma}\}$$

Hence: We should avoid integrality property of the relaxed program ! This though means we will have to find different ways to solve our Lagrangian sub-problem. The better bounding is illustrated in example 12.7.

Piecewise Linearity of Lagrangian function

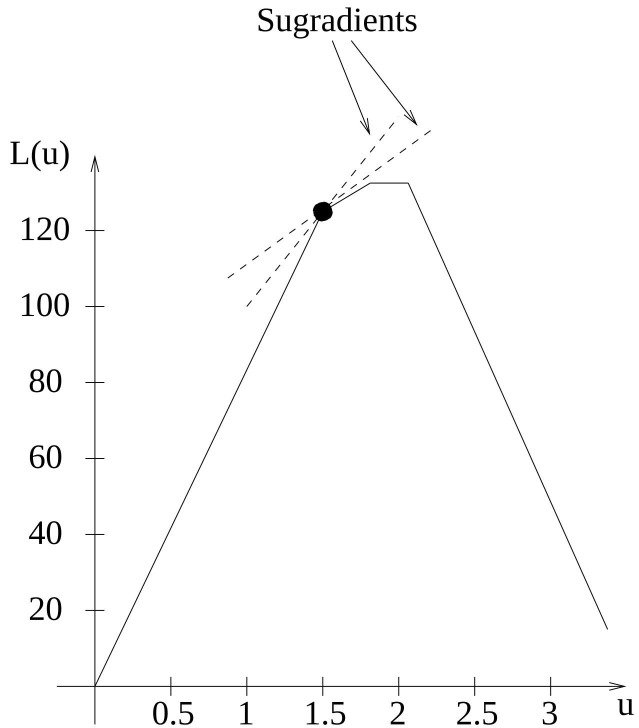
The Lagrangian function is piecewise linear (Figure 12.3 Kipp Martin):



This is proved (stated) in Proposition 12.10

What is a subgradient ? (formally)

A vector $\gamma \in \mathcal{R}^m$ is a subgradient if:
$L(u) \leq L(\overline{u}) + (u - \overline{u})^T \gamma \ , \ u \in \mathcal{R}^m$



This is proved (stated) in Proposition 12.10 (figure from example 12.13 and p. 409).

How can we calculate a subgradient ?

In Proposition 12.16, it is proven that the vector $\overline{\gamma} = (b - A\overline{x})$ is a subgradient. Hence we can simply look at the “slackness” (including negative slackness) of the relaxed constraints to calculate a subgradient. Notice that a subgradient gives a direction with maximal improvement (in more than two dimensions).

The subgradient optimisation algorithm

The algorithm is given in (12.18):

Step 1: (Initialization) Let $j \leftarrow 0, u^j \in (R^m)^+$ and $\epsilon > 0$
Step 2: $\gamma^j \leftarrow g(x^j)$ where $x^j \in \Gamma(u^j)$.
Step 3: Let $u^{j+1} \leftarrow max\{0, u^j + t_j \gamma^j\}$ where t_j is a positive scalar called the step size.
Step 4: If $||u^{j+1} - u^j|| < \epsilon$ stop, else let $j \leftarrow j + 1$ and go to **Step 2**.

How should the step size t_j be calculated ?

In Kipp Martin the same step size calculation is (12.25 and equally in Beasly):

$$t_j = \frac{\theta_j(L^* - L(u^j))}{||\gamma^j||}$$

This seems to be the standard approach, but there are several other possibilities. If the steps are chosen sufficiently well, convergence is guaranteed, but it may be slow ...

When is the Lagrangian Solution optimal ?

Proposition 12.14: What are the conditions for optimality (not just bounding) the problem ? Given an optimal solution $x(u)$, require that:

- $Bx(u) \leq d$, i.e. a feasible solution to the original problem.
- if $u_i > 0 \Rightarrow Bx(u)_i = d_i$, i.e. a kind of complementary slackness condition.

General L-relaxation Considerations

When relaxing we need to consider three things:

1. The **strength** of the Lagrangian bound
2. The **ease** of solving the Lagrangian subproblem (given multipliers)
3. The **ease** of performing multiplier optimisation

Exercise

Please look at the exercise (Exercise 8 from last week). I will go through the exercise later.

The Budgeting Problem (RKM 12.19)

$$
\begin{aligned}
min \quad & \sum_{ij} C_{ij} \cdot x_{ij} \\
s.t.: \quad \sum_{j} x_{ij} &= 1 \qquad \forall i \\
\sum_{i} x_{ij} &= 1 \qquad \forall j \\
\sum_{ij} T_{ij} \cdot x_{ij} &\leq 18 \\
x_{ij} &\in \{0, 1\}
\end{aligned}
$$

Which constraints to dualize ?

We have three possibilities:

1. Both types of constraints, i.e. completely un-constrained
2. Dualize the assignment constraints, i.e. a knapsack problem
3. Dualize the limit constraint, i.e. assignment problem

The L-relaxed Budgeting Problem (1)

Min:

$$\sum_{ij} C_{ij} \cdot x_{ij} + \sum_{i} \gamma_i (1 - \sum_{j} x_{ij}) + \sum_{j} \gamma_j (1 - \sum_{i} x_{ij}) +$$

$$\gamma(18 - \sum_{ij} T_{ij} \cdot x_{ij})$$

s.t.:

$$x_{ij} \in \{0, 1\} \qquad \gamma \geq 0$$

The L-relaxed Budgeting Problem (1), II

This is easy to optimize, just pick the x_{ij} for which there is a negative coefficient, but if relaxed, i.e. $x_{ij} \in [0, 1]$ the corresponding Linear Program possess the integrality property.

The L-relaxed Budgeting Problem (3)

Min:

$$\sum_{ij} C_{ij} \cdot x_{ij} + \gamma(18 - \sum_{ij} T_{ij} \cdot x_{ij})$$

s.t.:

$$\begin{aligned} \sum_{j} x_{ij} &= 1 \qquad \forall i \\ \sum_{i} x_{ij} &= 1 \qquad \forall j \end{aligned}$$

$$x_{ij} \in \{0,1\} \qquad \gamma \geq 0$$

The L-relaxed Budgeting Problem (3), II

This is the wellknown assignment problem for which there are efficient solution algorithms, but it also possess the integrality property if relaxed.

The Budgeting Problem (2)

Min:

$$\sum_{ij} C_{ij} \cdot x_{ij} + \sum_{i} \gamma_i (1 - \sum_{j} x_{ij}) + \sum_{j} \gamma_j (1 - \sum_{i} x_{ij})$$

s.t.:

$$\begin{aligned} \sum_{ij} T_{ij} \cdot x_{ij} &\leq 18 \\ x_{ij} &\in \{0, 1\} \end{aligned}$$

The Budgeting Problem (2), II

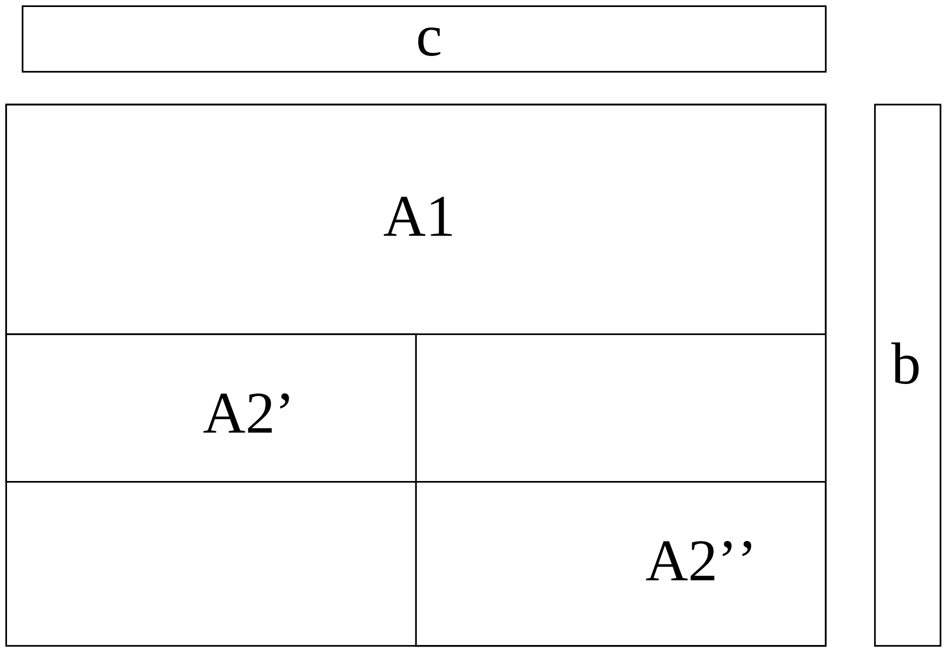
This is the wellknown knapsack problem, which is theoretically NP-hard. Since there are polynomial LP solvers, it cannot possess the integrality property. Furthermore there are reasonably efficient specialised optimisation methods and it is hence an interesting candidate for relaxation.

Lagrangian optimization vs. Dantzig-Wolfe

- In Dantzig-Wolfe, the master problem is given a number of columns from the subproblem to choose between, but unfortunately it may possibly choose several fractional columns. Improving constrained solution, i.e. adding variables.
- In Lagrangian relaxation, the relaxed problem only allows one solution, but this may not be feasible, because it may violate the relaxed constraints or it may be non-optimal if it is feasible. Improving bound.

The Block-Angular Structure

Suppose the optimisation problem has the following structure Given two convex sets, in two dimensions:



The Block-Angular Structure

We can use the structure in the following way:

- The variables are linked together by the A_1 matrix.
- If we seperate the matrixes and are able to solve the problems seperatly, we can solve the problem by solving the subproblem for the matrix'es A_1' and A_2''
- By performing Lagrangian relaxation of the A_1 matrix, the subproblems may be solved seperately, like in Dantzig-Wolfe optimisation.