Reducing Dimensionality by Relaxation

Lagrangian vs. Surrogate Relaxation

Dennis L. Bricker
Dept. of Industrial Engineering
The University of Iowa

TWO DIMENSIONAL KNAPSACK

Consider a knapsack with a weight capacity of **15** and a volume capacity of **12**.

Item #	Value	Weight	Volume
1	2	6	2
2	3	10	3
3	5	6	8
4	10	2	6
5	2	10	2
6	10	5	6
7	10	2	4
8	13	6	4
9	4	3	4
10	8	5	2
11	6	5	5
12	4	10	7

The state space for the two-dimensional DP model has 16×13 = **208** elements.

Suppose that we relax the volume restriction, using ***Lagrangian relaxation*** with multiplier ("shadow price") λ:

$$Max \ \sum v_j x_j + \lambda\left(b_2 - \sum_{j=1}^{n} a_{2j} x_j\right) = \sum_{j=1}^{n}\left(v_j - a_{2j}\right) x_j + \lambda b_2$$

The *shadow price* is interpreted as the value of one unit of volume, so that the profit contribution vj of an item must be adjusted by subtracting the value of the volume which it occupies, λa_{2j}.

The result is a **one-dimensional knapsack** problem which is more easily solved, having a much smaller state space (only 16 elements, rather than 208).

The difficulty lies in selecting the best values of the shadow price, λ:

generally the search for the best λ requires solution of a *sequence* of one-dimensional knapsack problems.

Furthermore, it may happen that the method fails to yield the optimal solution!

Lagrangian Relaxation

If initially, λ = 0, the result is

item	Value	Weight	Volume
4	10	2	6
6	10	5	6
7	10	2	4
8	13	6	4
Total:		15	20
Capacity:		15	12

Volume of the contents: 20
(which exceeds capacity 12)

Value of contents: 43
Lagrangian objective function

$$\sum_j \left(v_j - \lambda a_{2j}\right) x_j + \lambda b_2$$

is also 43 (since λ = 0), and therefore **43** is an upper bound on the optimum.

Since the volume restriction is violated, we ***increase*** the Lagrangian multiplier. Arbitrarily, let us set it equal to 1.0. This results in the solution:

item	Value	Weight	Volume
4	10	2	6
7	10	2	4
8	13	6	4
10	8	5	2
Total:		15	**16**
Capacity:		15	12

Volume of contents: 16
(exceeds volume restriction)

Value of contents: 25
Value of Lagrangian relaxation:

$$\sum_j \left(v_j - \lambda a_{2j}\right) x_j + \lambda b_2 = 25 + \lambda \times 12$$

= **37**,
which is an *improved* upper bound on the optimum.

Since the volume capacity is still exceeded, we increase the Lagrangian multiplier again, to λ = 2.00, resulting in the solution:

item	Value	Weight	Volume
7	10	2	4
8	13	6	4
10	8	5	2
Total:		13	**10**
Capacity:		15	12

Value of contents: 11
Value of Lagrangian relaxation:

$$\sum_j \left(v_j - \lambda a_{2j}\right) x_j + \lambda b_2 = 11 + \lambda \times 12 = \mathbf{35}$$

Another improvement on the upper bound!

The volume restriction is now slack, indicating that we should now ***decrease*** the multiplier. Linear interpolation would suggest λ = 1.3333, which results in the solution:

Lambda = 4/3

item	Value	Weight	Volume
4	10	2	6
7	10	2	4
8	13	6	4
10	8	5	2
Total:		15	**16**
Capacity:		15	12

Volume of contents: 16 *(infeasible!)*
Value of contents: 20
⇒ Value of Lagrangian relaxation:

$$\sum_j \left(v_j - \lambda a_{2j}\right) x_j + \lambda b_2$$

= 20+λ×12 = **36**
(upper bound is not improved!)

Since the volume restriction is again violated, we increase the Lagrangian multiplier, this time to 1.6667:

```
Lambda = 5/3
*** Optimal value is  14 ***
*** There are 2 optimal solutions ***
```

Optimal Solution No. 1

item	Value	Weight	Volume
7	10	2	4
8	13	6	4
10	8	5	2
Total:		13	**10**
Capacity:		15	12

Volume of contents: **10**

Optimal Solution No. 2

item	Value	Weight	Volume
4	10	2	6
7	10	2	4
8	13	6	4
10	8	5	2
Total:		15	16
Capacity:		15	12

Volume of contents: **16**

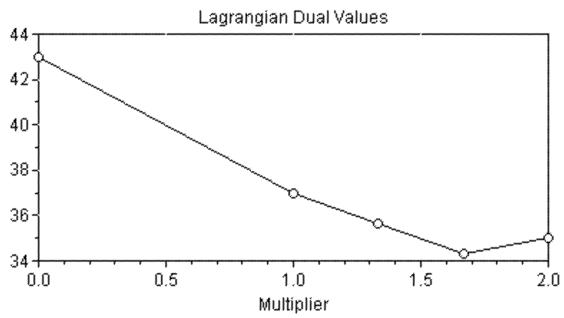
Value of contents is 14 ⇒ value of Lagrangian relaxation is

$$\sum_j \left(v_j - \lambda a_{2j}\right)x_j + \lambda b_2 = 14 + \lambda \times 12 = \mathbf{34}$$

(an improvement upon the upper bound).

Since volume capacity (12) lies ***within*** the interval [10,16],

the Lagrangian relaxation method has failed to find the solution.



By using a **2-dimensional state** variable, we can find the two optimal solutions, with value 31:

Optimal Solution No. 1

stage	state		decision
12	15	12	Omit
11	15	12	Omit
10	15	12	Include
9	10	10	Omit
8	10	10	Include
7	4	6	Omit
6	4	6	Omit
5	4	6	Omit
4	4	6	Include
3	2	0	Omit
2	2	0	Omit
1	2	0	Omit
0	2	0	

Total weight: 13
Total volume: 12

Optimal Solution No. 2

stage	state		decision
12	15	12	Omit
11	15	12	Omit
10	15	12	Include
9	10	10	Omit
8	10	10	Include
7	4	6	Include
6	2	2	Omit
5	2	2	Omit
4	2	2	Omit
3	2	2	Omit
2	2	2	Omit
1	2	2	Omit
0	2	2	

Total weight: 13
Total volume: 10

The Lagrangian ***duality gap*** is therefore 34 – 31 = 3, almost 10%.

Surrogate relaxation

In **Lagrangian relaxation**, the dimension of the state space is reduced by enforcing only one of the two resource restrictions, and assigning a "shadow price" to the other.

In **surrogate relaxation**, the dimension of the state space is reduced by replacing the original two resource restrictions with a nonnegative linear combination, i.e., multiplying each resource restriction by a nonnegative number and summing them.

Surrogate Relaxation

$$\textit{Maximize} \sum_{j=1}^{n} v_j x_j$$

$$\textit{subject to} \sum_{j=1}^{n}\left(\mu_1 a_{1j} + \mu_2 a_{2j}\right)x_j \le \mu_1 b_1 + \mu_2 b_2$$

$$x_j \in \{0,1\} \text{ for all } j = 1,2,\ldots n$$

As in the case of Lagrangian relaxation, we are left with a one-dimensional knapsack problem to be solved for every choice of multiplier vector. Assuming that the original resource requirements are integer, in order that the coefficients in this knapsack constraint be integer, we require that the multipliers μ_1 and μ_2 be integer. This then means that the state space must be ***expanded*** to include {0, 1, 2, … $\mu_1 b_1 + \mu_2 b_2$ }.

In each type of relaxation, the set of feasible solutions is increased by this type of relaxation, so that the optimal solution of the one-dimensional knapsack problem may not be feasible in the original two-dimensional knapsack problem

Example: Begin arbitrarily with multipliers $(\mu_1, \mu_2) = (1,1)$.
Note: the state space is now {0, 1, 2, … [15+12] }, i.e., of cardinality **28**.

```
*** Optimal value is   33 ***
*** There are 2 optimal solutions ***
```

Optimal Solution No. 1

item	Value	Weight	Volume
4	10	2	6
7	10	2	4
8	13	6	4
Total:		10	14
Capacity:		15	12

Optimal Solution No. 2

item	Value	Weight	Volume
6	10	5	6
7	10	2	4
8	13	6	4
Total:		13	14
Capacity:		15	12

Since the volume constraint is violated by both solutions, increase that resource's surrogate multiplier relative to that of the weight constraint:

$$(\mu_1, \mu_2) = (1,2)$$

so that the state space is now {0, 1, … [15+2×12]}, with cardinality **40**.

The solution of the 1-dimensional knapsack problem obtained with surrogate multipliers $(\mu_1, \mu_2) = (1,2)$ is

*** Optimal value is 33 ***

Optimal Solution

item	Value	Weight	Volume
4	10	2	6
7	10	2	4
8	13	6	4
Total:		10	14
Capacity:		15	12

Since the volume constraint is again violated, we further increase that resource's surrogate multiplier relative to that of the weight constraint:

$$(\mu_1, \mu_2) = (1,3)$$

so that the state space is now {0, 1, 2, … [15+3×12]} with cardinality **52**.

The optimal solution of the one-dimensional knapsack problem obtained with surrogate multipliers $(\mu_1, \mu_2) = (1,3)$ is

*** Optimal value is 31 ***
*** There are 2 optimal solutions ***

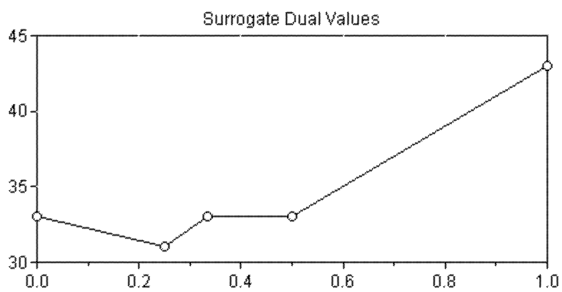
Optimal solution # 1

item	Value	Weight	Volume
4	10	2	6
8	13	6	4
10	8	5	2
Total:		13	12
Capacity:		15	12

Optimal solution # 2

item	Value	Weight	Volume
7	10	2	4
8	13	6	4
10	8	5	2
Total:		13	10
Capacity:		15	12

Both of these solutions are feasible in the original 2-dimensional problem! *Hence they are optimal in the original problem-- the surrogate duality gap is* ***zero****, while the Lagrangian duality gap was positive!*



Note that, since the surrogate relaxation isn't effected by scaling the multipliers, they can be normalized so as to sum to 1.0, and the search for the best surrogate multipliers is a one-dimensional search in the interval [0, 1]!

Note: Theory tells us that, in general,

surrogate duality gap $\le$ *Lagrangian duality gap!*

The "downside":

if we are using DP to solve the surrogate relaxation, we must restrict the multipliers to be integer (or equivalently, rational). The result is that the size of the state space must be increased, so our purpose of reducing the state space of the 2-dimensional DP is defeated!

See: Greenberg, H. J. and W. P. Pierskalla (1970). “Surrogate Mathematical Programming.” *Operations Research* **18**: 924-939.