Applications of Integer Programming

Wide area of applications:

1. Train scheduling
2. Airline Crew scheduling
3. Production Planning
4. Electricity Generation Planning
5. Telecommunications
6. Buses for the Handicapped (or Dial-a-Ride)
7. Ground Holding of Aircraft
8. Cutting Problems
9. Roster Problems

What is an integer program?

Linear Optimization (LO) Problem:

$$(LP) \qquad \max\left\{c^T x \ : \ Ax \leq b,\ x \geq 0\right\}$$

Integer (Linear) Problem:

$$(IP) \qquad \max\left\{c^T x \ : \ Ax \leq b,\ x \geq 0,\ x \text{ integer}\right\}$$

Mixed Integer Problem:

$$(MIP) \qquad \max\left\{c^T x + d^T y \ : \ Ax + Gy \leq b,\ x \geq 0,\ y \geq 0,\ y \text{ integer}\right\}$$

Binary Integer Problem:

$$(BIP) \qquad \max\left\{c^T x \ : \ Ax \leq b,\ x \in \{0,1\}^n\right\}$$

Combinatorial (or Discrete) Optimization Problem:

$$(COP) \qquad \min_{S \subseteq N}\left\{\sum_{j \in S} c_j \ : \ S \in \mathcal{F}\right\}$$

Here $N = \{1, \ldots, n\}$, c_j is the weight of $j \in N$, and $\mathcal{F}$ denotes a family of feasible subsets of N.

Example: Rounding does not work

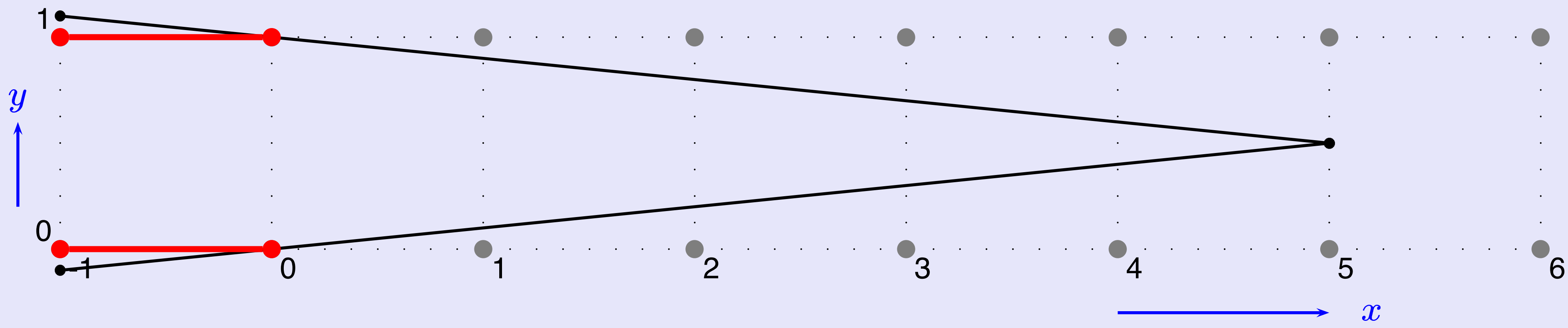
Let K be a large positive integer. Consider the problem

$$(IP) \qquad \max\{x \;:\; x - 2Ky \leq 0,\, x + 2Ky \leq 2K,\, y \text{ integer}\}.$$

The LO relaxation is

$$(LP) \qquad \max\{x \;:\; x - 2Ky \leq 0,\, x + 2Ky \leq 2K\}.$$

This problem admits the solution $x = K$, $y = \frac{1}{2}$ with objective value K. By adding the two inequalities we obtain $2x \leq 2K$, or $x \leq K$; so this solution is optimal. It is not integral, however. The IP solution has objective value 0, for the optimal solutions $x = y = 0$ and $x = 0$, $y = 1$. The figure below illustrates the example (for $K = 5$); the points that are feasible for (IP) are indicated as red.



Ingredients of an IP or BIP

1. Data of the problem instance

2. Decision variables

3. Objective function

In COP's we look for a suitable subset S of the set $N = \{1, \dots, n\}$. By representing the set S by its binary incidence vector $x \in \{0, 1\}^n$:

$$x_i = \begin{cases} 1, & \text{if } i \in S \\ 0, & \text{if } i \notin S, \end{cases}$$

every COP can be formulated as a BIP.

The Assignment Problem

n people are available to carry out n jobs. Each person is assigned to carry out exactly 1 job. The cost for assigning job j to person i is c_{ij}. The aim is to find a minimal cost assignment.

Decision variables: $x_{ij} = 1$ if person i does job j, otherwise $x_{ij} = 0$.

Constraints:

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1, \ldots, n$$
$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1, \ldots, n$$

Objective: $\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$.

The 0 - 1 Knapsack Problem

Budget b is available for investment in n possible projects the coming year; a_j is the outlet for project j and c_j is its expected return. The aim is to maximize the total expected return.

Decision variables: $x_j = 1$ if project j is selected, otherwise $x_j = 0$.

Constraints: $\sum_{j=1}^{n} a_j x_j \leq b$.

Objective: $\max \sum_{j=1}^{n} c_j x_j$.

The Set Covering Problem

$M = \{1, \ldots, m\}$ denotes a set of regions (or locations), and $N = \{1, \ldots, n\}$ a set of potential centers from which these regions can be serviced. $S_j \subseteq M$ are the regions that can be serviced by center $j \in N$, and c_j the installation cost of service center j. The problem of finding centers such that all regions are serviced at minimal cost is a COP:

$$\min_{T \subseteq N} \left\{ \sum_{j \in T} c_j \ : \ \cup_{j \in T} S_j = M \right\}.$$

For a BIP formulation we use a incidence matrix A whose columns are the incidence vectors of the sets S_j.

Decision variables: $x_j = 1$ if center j is selected, otherwise $x_j = 0$.

Constraints: $\displaystyle\sum_{j=1}^{n} a_{ij} x_j \geq 1, \quad i = 1, \ldots, m.$

Objective: $\displaystyle\min \sum_{j=1}^{n} c_j x_j.$

The Traveling Salesman Problem (TSP)

A salesman must visit each of n cities exactly once and then return to his starting point. The time to travel from city i to city j is c_{ij}. The aim is to find an order of the cities that minimizes the length of the tour.

We give a BIP formulation.

Decision variables: $x_{ij} = 1$ if the salesman goes directly from city i to city j, otherwise $x_{ij} = 0$; $x_{ii} = 0$ for every i.

Assingment constraints:

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1, \ldots, n$$
$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1, \ldots, n$$

Subtour elimination constraints:

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq 1, \quad S \subseteq N, S \neq \emptyset.$$

or, alternatively,

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \leq |S| - 1, \quad S \subseteq N, 2 \leq |S| \leq n - 1.$$

Objective: $\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$.

Alternative Subtour Elimination Constraints

Below we give a MIP formulation of the TSP.

Decision variables: $x_{ij} = 1$ if the salesman goes directly from city i to city j, otherwise $x_j = 0$; $x_{ii} = 0$ for every i.

Assingment constraints:

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1, \ldots, n$$
$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1, \ldots, n$$

Subtour elimination constraints: For each node i, we introduce a variable u_i. We want to assign values to these variables in such a way that for $i \neq 1$:

$$x_{ij} = 1 \quad \Rightarrow \quad u_j > u_i.$$

Obviously, this logical constraint can be satisfied if and only if the assigment does not contain a subtour (if there are subtours, there is a subtour that does not contain city 1; look at the values of u_i in the nodes of this subtour). The above logical constraint can be modelled as a linear constraint as follows:

$$u_i - u_j + n x_{ij} \leq n - 1, \quad 2 \leq i \leq n, \quad 1 \leq j \leq n$$

Objective: $\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$.

Complete enumeration $\rightarrow$ Combinatorial Explosion

Problem	Number of possibilities
Assignment Problem	$n!$
Travelling Salesman Problem	$(n-1)!$
Knapsack and Set Covering Problem	$\approx 2^{n-1}$

These functions grow exponentially fast. For example, a TSP with 101 cities has approximately 9.33×10^{157} tours. If a computer were able to evaluate 10^{10} (10.000 million) tours per second, it would take about 3×10^{140} years to evaluate all the tours.

N.B. 1 year has only $31.536.000 \approx 3.2 \times 10^{7}$ seconds.

Modelling fixed costs

In case of setup costs f for a facility we have to deal with a cost function of the form

$$h(x) = \begin{cases} f + p\,x, & \text{if } 0 < x \leq C, \\ 0, & \text{if } x = 0. \end{cases}$$

This can be modelled by using a binary variable y:

$$y = 1 \text{ if } x > 0, \text{ otherwise } y = 0,$$

and writing

$$h(x) = fy + p\,x$$

and using the constraint

$$x \leq Cy, \quad y \in \{0, 1\}.$$

Note that this is not completely satisfactory: it allows the solution $x = 0$, $y = 1$, with costs f. But, if we are minimizing cost, this solution will never be part of an optimal solution (unless $f = 0$).

Uncapacitated Facility Location (UFL)

$M = \{1, \ldots, m\}$ denotes a set of clients, and $N = \{1, \ldots, n\}$ a set of potential depots from which these clients can be serviced. The setup costs for depot j are f_j. If the demand of client i is completely delivered by depot j the transportation costs are c_{ij}. The problem is to decide which (potential) depots should be used so as to minimize the total costs.

Decision variables:

$y_j = 1$ if depot j is selected, otherwise $y_j = 0$

x_{ij} is the fraction of client's i demand delivered by depot j ($0 \leq x_{ij} \leq 1$).

Constraints:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$$

$$\sum_{i \in M} x_{ij} \leq m y_j \text{ for } j \in N, \quad y_j \in \{0, 1\} \text{ for } j \in N.$$

Objective: $\min \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{j \in N} f_j y_j.$

Uncapacitated Lot-Sizing (ULS)

The aim is to find an optimal production plan for an n-period horizon for a single product. The data are

f_t	fixed cost of producing in period t
p_t	unit production cost in period t
h_t	unit storage cost in period t
d_t	demand at the end of period t

Decision variables:
- x_t: amount produced in period t
- s_t: stock at the end of period t
- y_t: $y_t = 1$ if production occurs in period t, otherwise $y_t = 0$

Constraints:

$$\begin{aligned} s_{t-1} + x_t &= d_t + s_t, && t = 1, \ldots, n \\ x_t &\leq My_t, && t = 1, \ldots, n \quad (\text{ e.g. } M = \sum_{t=1}^{n} d_t) \\ s_0 = 0,\, s_t,\, x_t &\geq 0, && t = 1, \ldots, n \end{aligned}$$

Objective: $\min \sum_{t=1}^{n} p_t x_t + \sum_{t=1}^{n} h_t s_t + \sum_{t=1}^{n} f_t y_t.$

Discrete Alternatives (or Disjunctions)

Suppose that two jobs must be processed on the same machine and cannot be processed simultaneously. If p_i are the processing times, and the variables t_i the starting times $(i = 1, 2)$ for the two jobs, then either job 1 precedes job 2, and so $t_2 \geq t_1 + p_1$ or job 2 comes first and $t_1 \geq t_2 + p_2$.

More generally, consider the situation where $x \in \mathbf{R}^n$ satisfies $0 \leq x \leq u$ and

$$\text{either } x^T a^1 \leq b^1 \quad \text{or } x^T a^2 \leq b^2.$$

N.B. Usually we require $x^T a^1 \leq b^1$ **and** $x^T a^2 \leq b^2$, which defines a convex region. In the **or** case the feasible region is not convex!!

By introducing a large number $M \geq \max\left\{x^T a^i - b^i \;:\; 0 \leq x \leq u,\, i \in \{1, 2\}\right\}$ and binary variables y_i, this can be modelled by linear and binary constraint as follows:

$$\begin{aligned}
x^T a^i - b^i &\leq M(1 - y_i), \quad i = 1, 2 \\
y_1 + y_2 &= 1, \\
y_i &\in \{0, 1\}, \qquad\quad i = 1, 2 \\
0 \leq x &\leq u.
\end{aligned}$$

Definition 1 *A subset of $\mathbf{R}^n$ described by a finite set of linear constraints*

$$P = \{x \in \mathbf{R}^n \ : \ Ax \leq b\}$$

is called a polyhedron.

Definition 2 *A polyhedron $P \subseteq \mathbf{R}^{n+p}$ is called a (linear) formulation for a set $X \subseteq \mathbf{Z}^n \times \mathbf{R}^p$ if and only if $X = P \cap (\mathbf{Z}^n \times \mathbf{R}^p)$.*

N.B. A set $X \subseteq \mathbf{Z}^n \times \mathbf{R}^p$ may have more than one formulation. For example:

$$\emptyset = \left\{x \in \mathbf{R} \ : \ x^2 < 0\right\} = \left\{x \in \mathbf{R}^n \ : \ x^T x < 0\right\} = \{x \in \mathbf{R} \ : \ x \leq 4, \, x \geq 7\}\,.$$

Also, the polyhedra

$$\begin{aligned}
P_1 &= \left\{x \in \mathbf{R}^4 \ : \ 0 \leq x_i \leq 1 \ (i = 1, \ldots, 4), \, 83x_1 + 61x_2 + 49x_3 + 20x_4 \leq 100\right\}, \\
P_2 &= \left\{x \in \mathbf{R}^4 \ : \ 0 \leq x_i \leq 1 \ (i = 1, \ldots, 4), \, 4x_1 + 3x_2 + 2x_3 + x_4 \leq 4\right\}, \\
P_3 &= \left\{ x \in \mathbf{R}^4 \ : \ \begin{array}{ccccccccc}
4x_1 & + & 3x_2 & + & 2x_3 & + & 1x_4 & \leq & 4 \\
1x_1 & + & 1x_2 & + & 1x_3 & + & & \leq & 1 \\
1x_1 & & & & & + & 1x_4 & \leq & 1 \\
\multicolumn{9}{c}{0 \leq x_i \leq 1 \ (i = 1, \ldots, 4)}
\end{array} \right\}
\end{aligned}$$

are formulations for the same set X of points

$$X = \{(0,0,0,0), (1,0,0,0), (0,1,0,0), (0,0,1,0), (0,0,0,1), (0,1,0,1), (0,0,1,1)\}\,.$$

Equivalent Formulation for UFL

Decision variables:

$y_j = 1$ if depot j is selected, otherwise $y_j = 0$

x_{ij} is the fraction of client's i demand delivered by depot j ($0 \leq x_{ij} \leq 1$).

Constraints:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$$

$$\sum_{i \in M} x_{ij} \leq m y_j \text{ for } j \in N$$

$$y_j \in \{0, 1\} \text{ for } j \in N.$$

Objective: $\min \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{j \in N} f_j y_j.$

The second constraint ($\sum_{i \in M} x_{ij} \leq m y_j$ for $j \in N$) just models the requirement that if $x_{ij} > 0$ for some i, then the depot j should be open, i.e. $y_j = 1$. Since $0 \leq x_{ij} \leq 1$, this can be achieved also by using a different set of constraints:

$$0 \leq x_{ij} \leq y_j \text{ for } i \in M, j \in N.$$

Extended Formulation for ULS

Data

f_t	fixed cost of producing in period t
p_t	unit production cost in period t
h_t	unit storage cost in period t
d_t	demand at the end of period t

Decision variables: w_{it}: amount produced in period i to satisfy demand in period t

y_t: $y_t = 1$ if production occurs in period t, otherwise $y_t = 0$

Constraints:

$$\begin{aligned}
\sum_{i=1}^{n} w_{it} &\geq d_t, && t = 1, \ldots, n \\
w_{it} &\leq d_t y_i, && i = 1, \ldots, n,\ t = i, \ldots, n \\
w_{it} &\geq 0, && i, t = 1, \ldots, n
\end{aligned}$$

Objective: $\min \sum_{i=1}^{n} \sum_{t=i}^{n} c_{it} w_{it} + \sum_{t=1}^{n} f_t y_t.$

Here $c_{it} = p_i + h_{i+1} + \ldots + h_t$.

Ideal Formulation

Definition 3 *Given a set $X \subseteq \mathbf{R}^n$, the convex hull of X, denoted as $\mathrm{conv}\,(X)$, is defined as*

$$\mathrm{conv}\,(X) = \left\{ x \ : \ x = \sum_{i=1}^{k} \lambda_i x^i,\ \sum_{i=1}^{k} \lambda_i = 1,\ \lambda \geq 0,\ \left\{x^1, \ldots, x^k\right\} \subseteq X \right\}$$

Proposition 1 $\mathrm{conv}\,(X)$ *is a polyhedron (i.e. $\exists A, b \ni \mathrm{conv}\,(X) = \{x \ : \ Ax \leq b\}$)*

Proposition 2 *The extreme points of $\mathrm{conv}\,(X)$ all belong to X.*

Due to the last two results, one has

$$\max\left\{c^T x \ : \ x \in X\right\} = \max\left\{c^T x \ : \ x \in \mathrm{conv}\,(X)\right\} = \max\left\{c^T x \ : \ Ax \leq b\right\}.$$

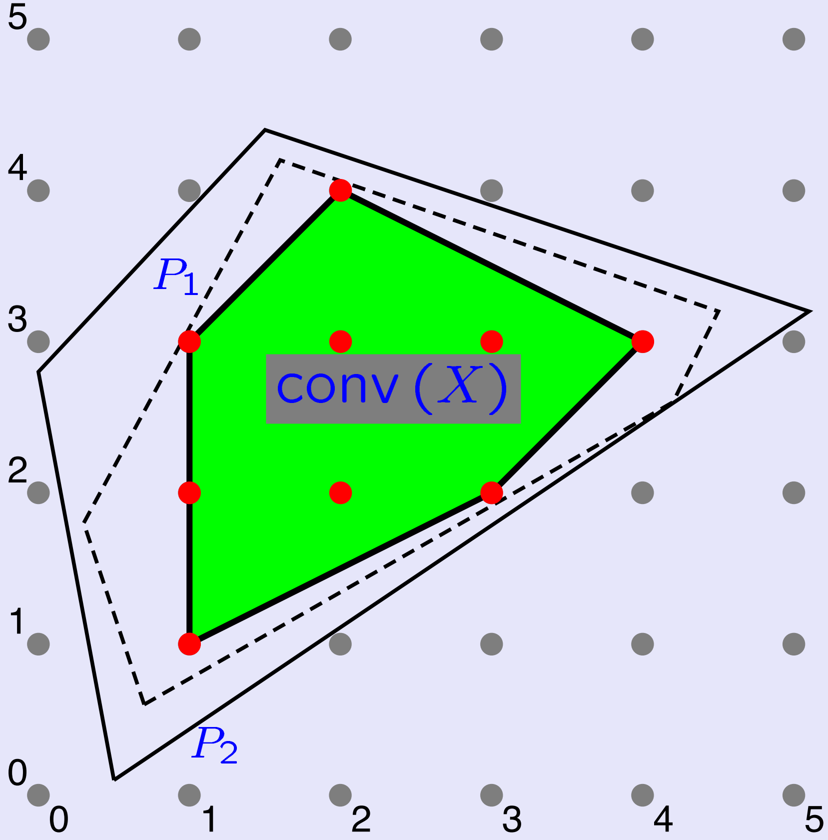
Thus $\mathrm{conv}\,(X)$ yields an ideal formulation of X: the solution of the last formulation immediately gives the solution of the first problem.

However, in general, it is hard to find A and b, and, moreover, the number of inequalities in $Ax \leq b$ to describe $\mathrm{conv}\,(X)$ may be enormous (exponential).

For any formulation P we have $X \subseteq \mathrm{conv}\,(X) \subseteq P$.

Definition 4 *Given a set $X \subseteq \mathbf{R}^n$, and two formulations P_1 and P_2 for X, we call P_1 a better formulation than P_2 if $P_1 \subseteq P_2$.*

The figure below illustrates the above definition.



Comparison of Formulations for a Knapsack Problem

Earlier we found three formulations for a Knapsack set, namely

$$
\begin{aligned}
P_1 &= \left\{x \in \mathbf{R}^4 \,:\, 0 \le x_i \le 1 \ (i = 1, \ldots, 4),\ 83x_1 + 61x_2 + 49x_3 + 20x_4 \le 100\right\}, \\
P_2 &= \left\{x \in \mathbf{R}^4 \,:\, 0 \le x_i \le 1 \ (i = 1, \ldots, 4),\ 4x_1 + 3x_2 + 2x_3 + x_4 \le 4\right\}, \\
P_3 &= \left\{ x \in \mathbf{R}^4 \,:\, \begin{array}{ccccccccc}
4x_1 & + & 3x_2 & + & 2x_3 & + & 1x_4 & \le & 4 \\
1x_1 & + & 1x_2 & + & 1x_3 & + & & \le & 1 \\
1x_1 & & & & & + & 1x_4 & \le & 1 \\
\multicolumn{9}{c}{0 \le x_i \le 1 \ (i = 1, \ldots, 4)}
\end{array} \right\}
\end{aligned}
$$

where X is the set of points

$$X = \{(0,0,0,0), (1,0,0,0), (0,1,0,0), (0,0,1,0), (0,0,0,1), (0,1,0,1), (0,0,1,1)\}.$$

One may easily verify that $P_3 \subseteq P_2 \subseteq P_1$. So P_3 is the best formulation of the three. In fact, in this case one has $P_3 = \text{conv}\,(X)$, so P_3 is an ideal formulation.

Comparison of the Formulations for UFL

We already have two formulations for the set X underlying UFL:

P_1	P_2
$(i)\quad \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$	$(i)\quad \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \ldots, m$
$(ii)\quad \sum_{i \in M} x_{ij} \leq m y_j$ for $j \in N$	$(ii)\quad 0 \leq x_{ij} \leq y_j$ for $i \in M$, $j \in N$
$(iii)\quad y_j \in [0,1]$ for $j \in N$.	$(iii)\quad y_j \in [0,1]$ for $j \in N$.

The formulations differ only in the second constraint. With j fixed, taking the sum over $i \in M = \{1, \ldots, m\}$ in $P_2(ii)$ yields $P_1(ii)$, since $|M| = m$. Hence $P_2 \subseteq P_1$. In fact, $P_2 \subset P_1$, as we show now. For simplicity assume that $m = kn$ for some integer $k \geq 2$. If we partition the clients in k groups, each of which is serviced by exactly one of the depots, and define x_{ij} accordingly, then $\sum_{i \in M} x_{ij} = k \leq m$ for each $j \in N$. Taking $y_j = k/m$ for each depot, we obtain a solution that satisfies P_1, but which does not satisfy P_2.

Comparison of the Formulations for ULS

We already have two formulations for the set X underlying ULS:

P_1					P_2				
(i)	$s_{t-1} + x_t$	$=$	$d_t + s_t,$	$t \in N$	(i)	$\sum_{t=1}^{n} w_{it}$	$\geq$	$d_t,$	$t \in N$
(ii)	x_t	$\leq$	$My_t,$	$t \in N$	(ii)	w_{it}	$\leq$	$d_t y_i,$	$t \in N, t \geq i$
(iii)	s_t, x_t	$\geq$	$0,$	$t \in N$	(iii)	w_{it}	$\geq$	$0,$	$i, t \in N$
(iv)	y_t	$\in$	$[0,1],$	$t \in N$	(iv)	y_t	$\in$	$[0,1]$	$t \in N$
(v)	s_0	$=$	$0\,.$		(v)	x_i	$=$	$\sum_{t=i}^{n} w_{it},$	$i \in N.$

The situation is less clear than in the UFL case, since the variables in both formulations are not the same. However, given a solution of P_2, we can compute each x_i form $P_2(v)$. Then we can compute the corresponding variables s_t from $P_1(i)$ and $P_1(v)$. We omit these computations, but in this way it can be shown that $P_2 \subseteq P_1$. The inclusion is strict, because $x_t = d_t$, $y_t = d_t/M$ (for each t) is a solution of P_1 (take $s_t = 0$ for each t) that can not be satisfied by P_2.

N.B. It can be shown that formulation P_2 is ideal.