Modeling with Auxiliary Decision Variables (4/12)

Learning Objectives:

- Use auxiliary decision variables to model multi-period optimization problems.
- Use auxiliary decision variables to reformualte certain non-linear optimization problems in a linear way.
- Describe what kind of non-linear objectives or constraints are allowed in Gurobi.

One powerful technique in LP/MIP modeling is to create additional decision variables to make it easier to formulate the objective or constraints. These variables are called **auxiliary decision variables** because they not necessary to encode the decision (hence "auxiliary") and their numerical value is not known in advance (hence they are decision variables, not inputs). We illustrate their use via a few examples.

Using Auxiliary Decision Variables for Multi-period problems

Exercise 4 from last week's handout

Finco Investment Corporation wishes to determine an investment strategy for the firm for the next 3 years. At present (Year 0), $100,000$ is available for investment. The goal is to maximize the cash on hand at the end of Year 3.

There are five investment options, labeled A, B, C, D, and E. The cash flow associated with investing $1 in each investment is given in the table below.

Investment option	Now (Year 0)	Year 1	Year 2	Year 3
A	-$1.00	$0.50	$1.00	0
B	$0.00	-$1.00	$0.50	$1.00
C	-$1.00	$1.20	0	0
D	-$1.00	0	0	$1.90
E	0	0	-$1.00	$1.50

For example, Option A must be invested this year, and each dollar of cash outflow now returns $0.50 in Year 1 and one dollar in Year 2. Option E mst be invested in Year 2 and each dollar returns $1.50 the next year.

To ensure that the company's portfolio is diversified, Finco required that at most $75,000 be placed in any single investment option. Returns from investments can be reinvested in the same year. For example, the positive cash flow received from Option C in Year 1 can be reinvested immediately in Option B. However, Finco cannot borrow funds, so net cash on hand must be non-negative in all years. Formulate this as an optimization problem.

Modeling Process

What is the decision (in words)?

How much to invest in each option?

What type of decision variables do you need to encode it and how many variables of each type? What does each variable mean?

5 continuous decision variables. Each variable denotes how much to invest in an option.

Define the above decision variables:

Let x_A, x_B, x_C, x_D, x_E denote how much to invest in each of the five options.

What is the objective (in words)?

Maximize cash at hand at the end of year 3.

What are the constraints (in words)? How many are there of each kind?

- Amount invested in each year cannot exceed the cash on hand from previous year plus any returns this year. (4 constraints, one for each year)
- Amount invested in each option is no more than 75000.

What auxiliary decision variables would make it easier to express the objective/constraints? Define these decision variables:
Let y_0, y_1, y_2, y_3 denote the cash at hand at the end of each year. (Continuous)
Express the objective as a linear function of decision variables.

$$\text{Maximize: } y_3$$

Express the constraints as linear functions of decision variables.

$$\begin{aligned} x_A, x_B, x_C, x_D, x_E &\leq 75000 \\ y_0, y_1, y_2, y_3 &\geq 0 \\ 10000 - x_A - x_C - x_D &= y_0 \\ y_0 + 0.5x_A + 1.2x_D - x_C &= y_1 \\ y_1 + x_A + 0.5x_B - x_E &= y_2 \\ y_2 + x_B + 1.9x_D + 1.5x_E &= y3 \end{aligned}$$

Mathematical Formulation (Concrete)

Decision variables:

- Let x_A, x_B, x_C, x_D, x_E denote how much to invest in each of the five options. (Continuous)
- Let y_0, y_1, y_2, y_3 denote the cash at hand at the end of each year. (Continuous)

LP:

$$\begin{aligned} \text{Maximize:} &\quad y_3 \\ \text{subject to:} & \\ \text{(Limit on investment)} &\quad x_A, x_B, x_C, x_D, x_E \leq 75000 \\ \text{(Non-negative cash flow)} &\quad y_0, y_1, y_2, y_3 \geq 0 \\ \text{(Cash flow in year 0)} &\quad 10000 - x_A - x_C - x_D = y_0 \\ \text{(Cash flow in year 1)} &\quad y_0 + 0.5x_A + 1.2x_D - x_C = y_1 \\ \text{(Cash flow in year 2)} &\quad y_1 + x_A + 0.5x_B - x_E = y_2 \\ \text{(Cash flow in year 3)} &\quad y_2 + x_B + 1.9x_D + 1.5x_E = y3 \\ \text{(Non-negative investments)} &\quad x_A, x_B, x_C, x_D, x_E \geq 0 \end{aligned}$$

Mathematical Formulation (Abstract)

Input Data:

- I: set of investment options.
- n: number of years in total.
- $T = \{0, 1, \cdots, n\}$: the set of years.
- a_{it}: the cash flow multiplier for each option $i \in I$ and each year $t \in T$. (This is the same as the number in the above table in row i and column t.)
- C: the initial cash at hand.
- B: the limit to invest in each option.

Decision variable:

- For each option $i \in I$, let x_i denote how much to invest in option i.
- For each year $t \in T$, let y_t denote the cash on hand at the end of year t.

LP:

$$
\begin{array}{rrl}
\text{Maximize:} & y_n & \\
\text{subject to:} & & \\
\text{(Limit on investment)} & x_i \leq B & \text{for each option } i \in I. \\
\text{(Non-negative cash flow)} & y_t \geq 0 & \text{for each year } t \in T. \\
\text{(Cash flow in year 0)} & C + \sum_{i \in I} a_{i0} x_i = y_0 & \\
\text{(Cash flow in later years)} & y_{t-1} + \sum_{i \in I} a_{it} x_i = y_t & \text{for each year } t \geq 1. \\
\text{(Non-negative investments)} & x_i \geq 0 & \text{for each option } i \in I.
\end{array}
$$

Exercise 1 from today's handout

Paris has come to you because she needs help paying off her credit card bills. She owes the amounts on her credit cards listed in the following table.

Credit Card	Balance	Monthly Rate
Saks Fifth Avenue	\$20,000	0.5%
Bloomingdale's	\$50,000	1.0%
Macy's	\$40,000	1.5%

Paris has agreed not to shop at any of these stores anymore, and she is willing to allocate up to \$5,000 per month to pay off these credit cards. All cards must be paid off within 36 months. Paris' goal is to minimize the total of all her payments.

To solve this problem, you must understand how interests on credit card loans are calculated. To illustrate, suppose Paris pays \$5,000 on Saks during month 1. Then her Saks balance at the beginning of month 2 is 20,000 – 5,000 + 0.005*20,000. This is because Paris incurs a 0.5% interest on her balance of 20,000 during month 1. The payment made during month 1 does not affect this interest.

Help Paris solve her problem by formulating it into a linear program.

Modeling Process

What is the decision (in words)?

How much to pay for each card in each month.

What type of decision variables do you need to encode it and how many variables of each type? What does each variable mean?

3×36 continuous decision variables, one for each card and each month, denoting how much to pay for that card in that month.

Define the above decision variables:

Let x_{it} denote how much to pay card $i \in \{S, B, M\}$ in month $t \in \{1, 2, \cdots, 36\}$.

What is the objective (in words)?

Minimize total payment over all 36 months.

What are the constraints (in words)? How many are there of each kind?

- Total amount paid in each month no more than 5000 (one constraint for each month).
- Paying off all cards within 36 months.
- Equations governing the interest that needs to be paid for each card in each month. (one constraint for each card and each month).

What auxiliary decision variables would make it easier to express the objective/constraints? Define these decision variables:
Let y_{it} denote the left over balance for card i at the end of month t.
Express the objective as a linear function of decision variables.

$$\text{Minimize: } \sum_{it} x_{it}$$

Express the constraints as linear functions of decision variables.

$$\begin{aligned}
x_{St} + x_{Bt} + x_{Mt} &\leq 5000 \quad \text{for each month } 1 \leq t \leq 36. \\
y_{S(36)}, y_{B(36)}, y_{M(36)} &= 0 \\
(1 + 0.005)20000 - x_{S0} &= y_{S0} \\
(1 + 0.005)y_{S1} - x_{S1} &= y_{S1} \\
&\cdots \\
(1 + 0.01)50000 - x_{B0} &= y_{B0} \\
(1 + 0.01)y_{B1} - x_{B1} &= y_{B1} \\
&\cdots \\
(1 + 0.015)40000 - x_{M0} &= y_{M0} \\
(1 + 0.015)y_{M1} - x_{M1} &= y_{M1} \\
&\cdots
\end{aligned}$$

Mathematical Formulation (Abstract)

Input Data:

- I: set of credit cards.
- n: number of months in total.
- $T = \{1, \cdots, n\}$: the set of months.
- y_{i0}: the initial balance of card i.
- C: the total cash available to pay the credit cards each month.
- r_i: monthly interest rate for each card i.

Decision variable: For each card $i \in I$ and each month $t \in T$,

- let x_{it} (continuous) denote how much to pay this card in this month, and
- let y_{it} (continuous) denote the remaining balance in this card at the end of the month. (Note that y_{i0} is not a decision variable but is a part of the input data.)

LP:

$$\begin{aligned}
\text{Minimize:} \quad & \sum_{i \in I} \sum_{t \in T} x_{it} \\
\text{subject to:} \quad & \\
\text{(Limit on payments)} \quad & \sum_{i \in I} x_{it} \leq C \quad \text{for each month } t \in T. \\
\text{(Paying off at the end)} \quad & y_{in} = 0 \quad \text{for each card } i \in I. \\
\text{(Credit card accounting)} \quad & (1 + r_i)y_{i(t-1)} - x_{it} = y_{it} \quad \text{for each card } i \in I \text{ and each month } t \in T. \\
\text{(Non-negativivity)} \quad & x_{it}, y_{it} \geq 0 \quad \text{for each card } i \text{ and each month } t.
\end{aligned}$$

Other Tricks with Auxiliary Decision Variables

There are many modeling tricks involving auxiliary decision variables. As you are exposed to more kinds of problems, you will pick up more tricks. Knowing when to apply each trick comes from experience.

Either/or constraint

Suppose that X and Y are decision variables for how much we want to produce of each type of good, and $X + 2Y$ is the total amount iron we need, as each unit of X requires 1 unit of iron and each unit of Y 2 units. Suppose we want to either use no iron, or use at between 10 to 100 units. In other words, the supplier is unwilling to supply small amounts of iron to us, and the minimum order is 10.

Define an auxiliary binary variable Z. This variable has no meaning in the context of the problem, but is simply a part of a mathematical trick to express the either/or constraint in a linear way.

$$0Z + 10(1 - Z) \leq X + 2Y \leq 0Z + 100(1 - Z)$$

.

When $Z = 1$, then the above becomes

$$0 \leq X + 2Y \leq 0,$$

which means that $X + 2Y = 0$ (not using any iron at all). When $Z = 0$, then the above becomes

$$10 \leq X + 2Y \leq 100.$$

In general, you can use this trick to require a linear expression to be either in a certain range or in another range. (Replace $X + 2Y$ with any linear function of decision variables, and replace the ranges $[0, 0]$ and $[10, 100]$ with other numbers.)

Moving a term of the objective to a constraint

The following trick allows us to transform any non-linear objective function to a linear objective, and create a new non-linear constraint. Instead of

$$\text{Minimize } f(x)$$

we can write

$$\begin{array}{rl} \text{Minimize} & z \\ \text{subject to} & \\ \text{(New constraint)} & f(x) \leq z \end{array}$$

Here, z is a continuous auxiliary decision variable that we create. It does not necessarily have a business interpretation. It is simply a mathematical trick. This trick was used previously when minimizing the maximum of certain functions in Course notes to 17-LP modeling

For example, instead of minimizing the non-linear expression,

$$\text{Minimize } \max(x, y)$$

We can do,

$$\begin{array}{rc} \text{Minimize} & z \\ \text{subject to} & \\ \text{(New constraint)} & \max(x,y) \leq z \end{array}$$

The non-linear constraint can be in turn transformed to a series of linear constraints:

$$\begin{array}{rl} \text{Minimize} & z \\ \text{subject to} & \\ \text{(New constraint 1)} & x \leq z \\ \text{(New constraint 2)} & y \leq z \end{array}$$

Another example of this idea is to minimize absolute values. Instead of

$$\text{Minimize } |x_1 - y_1| + |x_2 - y_2|$$

We can do,

$$\begin{array}{rr} \text{Minimize} & z_1 + z_2 \\ \text{subject to} & \\ \text{(New constraint 1)} & |x_1 - y_1| \leq z_1 \\ \text{(New constraint 2)} & |x_2 - y_2| \leq z_2 \end{array}$$

Now, because $|x| = \max(x, -x)$, we have $|x_1 - y_1| = \max(x_1 - y_1, y_1 - x_1)$, so the above non-linear constraints can be in turn be equivalently represented by the following series of linear constraints.

$$\begin{array}{rr} \text{Minimize} & z_1 + z_2 \\ \text{subject to} & \\ \text{(New constraint 1-a)} & x_1 - y_1 \leq z_1 \\ \text{(New constraint 1-b)} & y_1 - x_1 \leq z_1 \\ \text{(New constraint 2-a)} & x_2 - y_2 \leq z_2 \\ \text{(New constraint 2-b)} & y_2 - x_2 \leq z_2 \end{array}$$

Non-linear Objectives and Constraints in Gurobi

There are very few non-linear expressions allowed in Gurobi. This is because non-linear optimization is difficult in general and there are no tools out there that can solve generic large-scale non-linear optimization problems and guarantee a correct solution. (There are solvers but they may return a local optimum instead of a global optimum.) There are tractable special cases of non-linear optimization problems, but they require certain particular mathematical structure, which real world business problems seldomly satisfy. Your main strategy for tackling non-linear optimization problems should be to define extra decision variables to convert it to a linear problem as above, or to change the problem definition to solve a linear problem instead.

Gurobi does however allow you to work with certain quadratic expressions, as outlined below. **The main takeaway is that only very specific non-linear expressions are allowed.**

Minimizing a sum of squares

Let x, y and z be decision variables. **The following type of objective IS allowed** in Gurobi. This type of expressions occur natually in finance when minimizing the variance of a portfolio. **See example 8.2 in DMD chapter 8 for a concrete example.**

$$\text{Minimize: } x^2 + (x + 2y)^2 + 3(y + z)^2 - 5y$$

Notice that the objective is the sum of linear terms ($-5y$), as well as squared terms. It is okay to multiply a squared term with a positive number (like 3), but it is not okay to multiple by a negative number. In other words, **the following is NOT allowed in Gurobi** because of the $-3(y + z)^2$:

$$\text{Minimize: } x^2 + (x + 2y)^2 - 3(y + z)^2 - 5y$$

Furthermore, you can only minimize such an expression, but not maximize. So **the following is also NOT allowed in Gurobi** because it is maximizing the sum of squares:

$$\text{Maximize: } x^2 + (x + 2y)^2 + 3(y + z)^2 - 5y$$

However, it is okay if you write the expression not as a sum of squares, but in an equivalent way. For example, **the following IS allowed**, as it is simply the expansion of the first expression.

$$\text{Minimize: } 2x^2 + 7y^2 + 3z^2 + 4xy + 6yz - 5y$$

Sum of squares less than or equal to a linear expression

Instead of minimizing the sum of squares, **Gurobi also allows having the sum of squares be less than or equal to a linear expression**. For example, if x, y and z are decision variables, the following constraints are allowed.

$$\begin{aligned} x^2 + y^2 + z^2 &\leq 5 \\ (x + y)^2 &\leq 5z - 2x \\ x^2 + y^2 + 2xy &\leq 5z - 2x \end{aligned}$$