Oil blending

Oil blending

How can an oil company maximize its profits?

An oil company manufactures different types of gasoline and diesel. Each type of gasoline is produced by blending different types of crude oils that must be purchased. The company must decide how much crude oil to buy in order to maximize its profit while respecting processing capacities and quality levels and satisfying customer demand.

The model (`Oil-blending.lp`)

`Crude1`, `Crude2`, and `Crude3` denote the different types of crude oil that can be purchased. `Super` and `Regular` represent the different types of gasoline produced, and `Diesel` represents the diesel produced.

Modeling the variables

`Blend` variables are used to represent the amount of each type of crude oil used to make each type of gasoline or diesel. Inventory variables (`Inventory`) denote the amount spent for storing each product.

Modeling the data

As this is an `.lp` file, the data is embedded in the model. For each type of crude oil, there are capacities of what can be bought, the buying price, the octane level, and the lead level. For each type of gasoline or diesel, there is customer demand, selling prices, and octane and lead levels. There is a maximum level of production imposed by the factory's limit as well as a fixed production cost. There are inventory costs for each type of final product and blending proportions. All of these have actual values in the model.

Modeling the objective function

The objective or goal is to maximize profit, which is made from sales of the final products minus total costs. The costs consist of the purchase cost of the crude oils, production costs, and inventory costs.

```
Maximize
 obj: 21 Blend("Crude1")("Super") + 31 Blend("Crude2")("Super")
      + 41 Blend("Crude3")("Super") + 11 Blend("Crude1")("Regular")
      + 21 Blend("Crude2")("Regular") + 31 Blend("Crude3")("Regular")
      + Blend("Crude1")("Diesel") + 11 Blend("Crude2")("Diesel")
      + 21 Blend("Crude3")("Diesel") - 0.1 Inventory("Super")
      - 0.1 Inventory("Regular") - 0.1 Inventory("Diesel")
```

Modeling the constraints

Demand constraints ensure that customer demand is met for each final product:

```
 ct_demand({"Super"}):       Blend({"Crude1"})({"Super"})
                             + Blend({"Crude2"})({"Super"})
                             + Blend({"Crude3"})({"Super"})
                             - Inventory({"Super"})  = 3000
 ct_demand({"Regular"}):     Blend({"Crude1"})({"Regular"})
                             + Blend({"Crude2"})({"Regular"})
```

```
                           + Blend({"Crude3"})({"Regular"})
                           - Inventory({"Regular"})  = 2000
ct_demand({"Diesel"}):     Blend({"Crude1"})({"Diesel"})
                           + Blend({"Crude2"})({"Diesel"})
                           + Blend({"Crude3"})({"Diesel"})
                           - Inventory({"Diesel"})  = 1000
```

and Capacity constraints limit the blending quantities to the factory capacities:

```
ct_capacity({"Crude1"}):   Blend({"Crude1"})({"Super"})
                           + Blend({"Crude1"})({"Regular"})
                           + Blend({"Crude1"})({"Diesel"}) <= 5000
ct_capacity({"Crude2"}):   Blend({"Crude2"})({"Super"})
                           + Blend({"Crude2"})({"Regular"})
                           + Blend({"Crude2"})({"Diesel"}) <= 5000
ct_capacity({"Crude3"}):   Blend({"Crude3"})({"Super"})
                           + Blend({"Crude3"})({"Regular"})
                           + Blend({"Crude3"})({"Diesel"}) <= 5000
ct_total_max_prod:         Blend({"Crude1"})({"Super"})
                           + Blend({"Crude2"})({"Super"})
                           + Blend({"Crude3"})({"Super"})
                           + Blend({"Crude1"})({"Regular"})
                           + Blend({"Crude2"})({"Regular"})
                           + Blend({"Crude3"})({"Regular"})
                           + Blend({"Crude1"})({"Diesel"})
                           + Blend({"Crude2"})({"Diesel"})
                           + Blend({"Crude3"})({"Diesel"}) <= 14000
```

Octane and Lead constraints enforce the quality criteria for the gasoline and diesel:

```
ct_octane_min({"Super"}):   2 Blend({"Crude1"})({"Super"})
                            - 4 Blend({"Crude2"})({"Super"})
                            - 2 Blend({"Crude3"})({"Super"}) >= 0
ct_octane_min({"Regular"}): 4 Blend({"Crude1"})({"Regular"})
                            - 2 Blend({"Crude2"})({"Regular"}) >= 0
ct_octane_min({"Diesel"}):  6 Blend({"Crude1"})({"Diesel"})
                            + 2 Blend({"Crude3"})({"Diesel"}) >= 0
ct_lead_max({"Super"}):     - 0.5 Blend({"Crude1"})({"Super"})
                            + Blend({"Crude2"})({"Super"})
                            + 2 Blend({"Crude3"})({"Super"}) <= 0
ct_lead_max({"Regular"}):   - 1.5 Blend({"Crude1"})({"Regular"})
                            + Blend({"Crude3"})({"Regular"}) <= 0
ct_lead_max({"Diesel"}):    - 0.5 Blend({"Crude1"})({"Diesel"})
                            + Blend({"Crude2"})({"Diesel"})
                            + 2 Blend({"Crude3"})({"Diesel"}) <= 0
```

The data

The data is embedded in the model here as this is an `.lp` file.

The solution

The solution output gives the optimal solution with `objectiveValue 287750` which is the total profit that can be made from final product sales minus all the costs. The solution also shows how to achieve this by setting the decision variables to the best values measured by the goal. These values determine how much of each crude oil must be blended to make each type of gasoline or diesel and how much money must be spent on inventory to gain the most profit.

These variable values are provided in an `.xml` file format. For example:

```
<variable status="BS" index="0" name="Blend({"Crude1"})({"Super"})"
                                          reducedCost="-0" value="2222.22222222222"/>
```

The variable status indicates whether the variable is basic (BS), at its lower limit (LL) or upper limit (UL). The index is an internal number used to reference each variable according to the order it was declared in the model. The optimal value for this variable indicates that 2222.22 units of `Crude1` must be blended to make `Super` gasoline.

Another example of an optimal variable value given in the solution is as follows:

```
<variable status="LL" index="9" name="Inventory({"Super"})"
                                                  reducedCost="-20.9" value="0"/>
```

Reduced costs are provided like this for non-basic variables indicating by how much the objective coefficients of these variables would need to be changed in order for them to be used in the solution. Negative reduced costs indicate that it is not profitable for these variables to be used in the solution. The optimal value for this variable is zero which indicates that no `Super` gasoline must be stored.

The constraints are also provided in a similar format with dual costs given for constraints with zero slack, indicating the effect on the objective values if the right hand side value of the constraints were increased. Negative dual costs indicate that it is not profitable for these values to be increased.