AMPL

A Quick-Start Guide

What is AMPL ?

- Algebraic modeling language for optimization
- Supports LP, MILP, MINLP
- Interfaces with solvers (CPLEX, Gurobi, etc.)
- Separates model, data, and execution

AMPL Files

Each problem instance uses :

- **.mod** : model (formulation)
- **.dat** : data (parameters)
- **.run** : execution script

Model Components

- Parameters
- Sets
- Variables
- Objective
- Constraints

Parameters

```
param n > 0;
param w{1..n} > 0;
param a{1..n,1..m};
```

Sets

```
set N := 1..n;
param w{j in N} > 0;
```

Variables

```
var x{j in N} >= 0, <= 1, binary;
```

Objective

```
maximize total_profit:
   sum{j in N} p[j]*x[j];
```

Constraints

```
subject to capacity:
  sum{j in N} w[j]*x[j] <= c;
```

Knapsack Model

```
param n > 0;
set N := 1..n;
param p{j in N} > 0;
param w{j in N} > 0;
param c > 0;

var x{j in N} binary;

maximize profit:
 sum{j in N} p[j]*x[j];

subject to capacity:
 sum{j in N} w[j]*x[j] <= c;
```

Data Example

```
param n := 3;
param c := 50;

param p :=
1 10
2 20
3 30;
```

Run Script

```
model model.mod;
data data.dat;

option solver gurobi;
solve;
```

Running AMPL

- Using IDE : `include file.run`
- Command line : `ampl file.run`

Useful Commands

```
display x;
expand capacity;

for {j in N} {
   display x[j];
}
```

Why to use it

- Clear modeling structure
- Solver-independent
- Scalable and expressive

Links

We will use the NEOS server services :
https ://neos-server.org/neos/solvers/index.html
Some exemples are available at :
`https://dev.ampl.com/ampl/books/ampl/examples`

Let's start by the diet model we saw the first lecture
`https://dev.ampl.com/ampl/books/ampl/examples/diet.mod.html`