Introduction

Brandon Moretz

15th May 2020

Introduction

```
using JuMP; using Clp;

using GLPKMathProgInterface
```

Example Solver

$$min \sum_{(i,j)} c_{ij} xij$$

subject to

$$\sum_{i,j \in A} x_{ij} - \sum_{(j,i) \in A} x_{ij} = b_i \; \forall_i \in N$$

$$0 \leq x_{ij} \leq 1 \forall (i,j) \in A$$

```
model = Model(Clp.Optimizer)

@variable(model, 0 <= x <= 40)
@variable(model, y <= 0)
@variable(model, z <= 0)

@objective(model, Max, x + y + z)

@constraint(model, const1, -x + y + z <= 20)
@constraint(model, const2, x + 3y + z <= 30)

display(model)
```

```
A JuMP Model
Maximization problem with:
Variables: 3
Objective function type: GenericAffExpr{Float64,VariableRef}
`GenericAffExpr{Float64,VariableRef}`-in-`MathOptInterface.LessThan{Float64
}`: 2 constraints
```

```
`VariableRef`-in-`MathOptInterface.GreaterThan{Float64}`: 1 constraint
`VariableRef`-in-`MathOptInterface.LessThan{Float64}`: 3 constraints
Model mode: AUTOMATIC
CachingOptimizer state: EMPTY_OPTIMIZER
Solver name: Clp
Names registered in the model: const1, const2, x, y, z
```

```
optimize!(model)

results = [JuMP.value(x), JuMP.value(y), JuMP.value(z) ]

display(results)
```

```
3-element Array{Float64,1}:
 40.0
 -3.3333333333333335
  0.0
```