An Introduction to BLOM

Berkeley Library for Optimization Modeling

Jason Kong

Department of Mechanical Engineering
University of California, Berkeley

March 17, 2014

Outline

1. BLOM Introduction
   - Quick Optimization Review
   - What is BLOM?
2. Examples
3. Setting up BLOM on your computer

Outline

1. BLOM Introduction
   - Quick Optimization Review
   - What is BLOM?

2. Examples

3. Setting up BLOM on your computer

Quick Optimization Review

Some terms and definitions

$$\begin{aligned} \min_{x,u} \quad & f(x,u) \\ \text{s.t.} \quad & g(x,u) \leq 0 \\ & h(x,u) = 0 \end{aligned}$$

- Objective Function $f(x)$ - Typically what you want to minimize or maximize over.
  Examples: energy, fuel, and distance
- Inequality Constraints $g(x) \leq 0$
  Examples: $v \leq 60$ mph, $a \leq 1$ m/s
- Equality Constraints $h(x)$
  Examples: $x(t+1) = Ax(t)$ for $t = 0, 1, 2, \ldots$

Note: Currently no toolboxes built into Simulink to do this!

Outline

1. BLOM Introduction
   - Quick Optimization Review
   - What is BLOM?

2. Examples

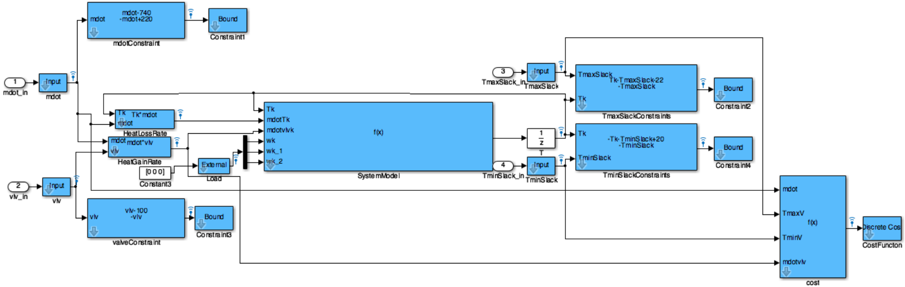
3. Setting up BLOM on your computer

Advantages of Using BLOM

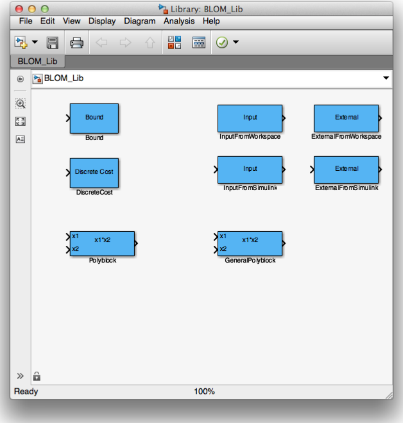
- Stands for Berkeley Library for Optimization Modeling
- Provides a graphical interface to allow users to create optimization problems using Simulink Blocks
- Exports mathematical models to optimization solvers (e.g. IPOPT)
- Great for optimization problems with “dynamics” that evolve over time

$$\begin{aligned} \min_{x,u} \quad & f(x,u) \\ \text{s.t.} \quad & g(x,u) \leq 0 \\ & h(x,u) = 0 \end{aligned}$$

General Structure and Look of a BLOM Model



BLOM Library Blocks and Their Functions



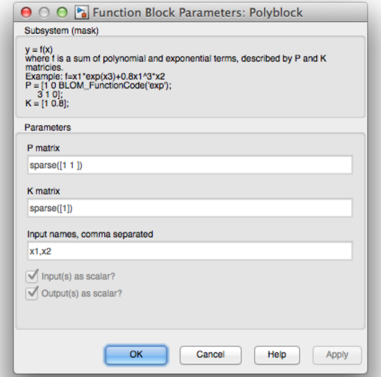
- **Externals** - Labels External Variables that can be changed via script or command line for different calls of the solver
- **Inputs** - Labels Input Variables to be optimized by the solver
- **Bounds** - Sets upper/lower bounds on a variable
- **Cost** - Cost to be minimized
- **Polyblocks** - BLOM's convenient way to create nonlinear functions

Polyblock

The polyblock allows us to easily create nonlinear equations based on the inputs given

$$f(x) = \sum_{k=1}^{r} K_k \left(\prod_{j=1}^{n} v(x_j, P_{kj}) \right)$$

$$v(x,p) = \begin{cases} x^p & \text{if } p \text{ is not an exception code} \\ \exp(x) & \text{if } p \text{ is the code for } \exp \\ \sin(x) & \text{if } p \text{ is the code for } \sin \\ \tanh(x) & \text{if } p \text{ is the code for } \tanh \\ \text{etc.} & \end{cases}$$



```
y = f(x)
where f is a sum of polynomial and exponential terms, described by P and K
matricies.
Example: f=x1*exp(x3)+0.8x1^3*x2
P = [1 0 BLOM_FunctionCode('exp');
     3 1 0];
K = [1 0.8];
```

P matrix: sparse([1 1])

K matrix: sparse([1])

Input names, comma separated: x1,x2

☑ Input(s) as scalar?

☑ Output(s) as scalar?

Note: it is entirely optional to use the Polyblock.

P & K Polyblock Example

$$y_1 = 2x_1^2 x_2, \quad y_2 = 3x_1 + x_2^4$$

$$P = \begin{bmatrix} 2 & 1 \\ 1 & 0 \\ 0 & 4 \end{bmatrix} \qquad K = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 1 \end{bmatrix}$$

$$y_1 = 2x_1 + 3\sin(x_2), y_2 = 3x_1^2 e^{x_3} + 0.2\tan(x_2)x_4^3$$

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \text{BLOM\_FunctionCode('sin')} & 0 & 0 \\ 2 & 0 & \text{BLOM\_FunctionCode('exp')} & 0 \\ 0 & \text{BLOM\_FunctionCode('tan')} & 0 & 3 \end{bmatrix}$$

$$K = \begin{bmatrix} 2 & 1 \\ 3 & 0.2 \end{bmatrix}$$

Calling BLOM

Always remember to call BLOM_addpath before starting. This adds all the BLOM files to your path so that you can call them.

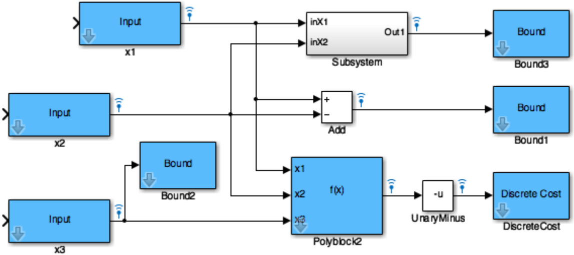
```
BLOM_SetDataLogging('ModelName')
ModelSpec = BLOM_ExtractModel('ModelName', timesteps)
[RunResults ResultsVec] = BLOM_RunModel(ModelSpec);
[OptGuess ExtVars InitialStates ] = ...
     BLOM_SplitResults(ModelSpec,RunResults);
SolverStruct = BLOM_ExportToSolver(ModelSpec,'IPOPT');
SolverStructData = ...
     BLOM_SetProblemData(SolverStruct,ModelSpec,OptGuess, ExtVars, InitialStates);
SolverResult  =  BLOM_RunSolver(SolverStructData,ModelSpec);
```

Note: ModelName is the name of your model.

timesteps is the number of time steps you want the model to run for.

A Simple Optimization Problem

$$\begin{aligned} \max \quad f(x) = 3x_1 + x_2 - x_3^2 + 2x_3 \\ x_1^2 + x_2^2 \leq 5 \\ x_1 - x_2 \leq 1 \\ x_3 \geq 0 \end{aligned}$$



Calling BLOM

Always remember to call BLOM_addpath before starting. This adds all the BLOM files to your path so that you can call them.

```
BLOM_SetDataLogging('ModelName')
ModelSpec = BLOM_ExtractModel('ModelName', timesteps)
[RunResults ResultsVec] = BLOM_RunModel(ModelSpec);
[OptGuess ExtVars InitialStates ] = ...
     BLOM_SplitResults(ModelSpec,RunResults);
SolverStruct = BLOM_ExportToSolver(ModelSpec,'IPOPT');
SolverStructData = ...
     BLOM_SetProblemData(SolverStruct,ModelSpec,OptGuess, ExtVars, InitialStates);
SolverResult  =  BLOM_RunSolver(SolverStructData,ModelSpec);
```

Note: ModelName is the name of your model.

timesteps is the number of time steps you want the model to run for.

How to get parameters out

- `SolverResult` struct contains all variables
- variables named based on block (e.g. `Gain1`)
  - If it's inside a subsystem, separated by underscore (e.g. `Subsystem_Add1`)
  - Blocks with vector outputs or multiple outputs will have a matrix within `SolverResult` that has a column for each variable

Trajectory Control Example

$$\min \sum ||v_k||_{R_1} + ||\theta_k||_{R_2} + ||y_k - y_{ref,k}||_Q$$

$$0 \leq v_k \leq v_{max} \quad 0 \leq \theta_k \leq \theta_{max}$$

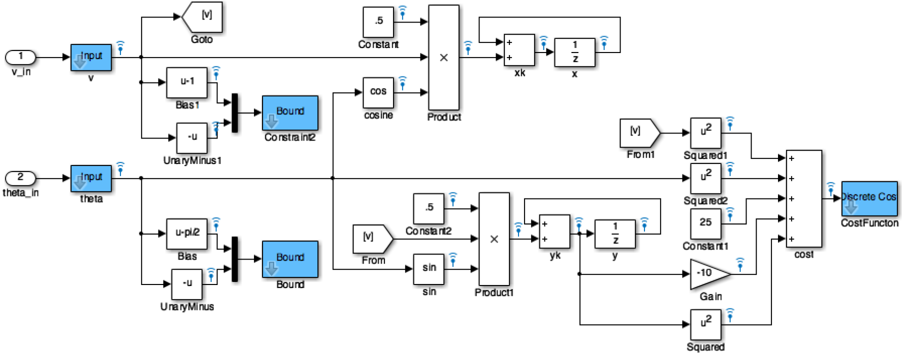
$$x_{k+1} = x_k + v_k \cos\theta_k \Delta t$$

$$y_{k+1} = y_k + v_k \cos\theta_k \Delta t$$

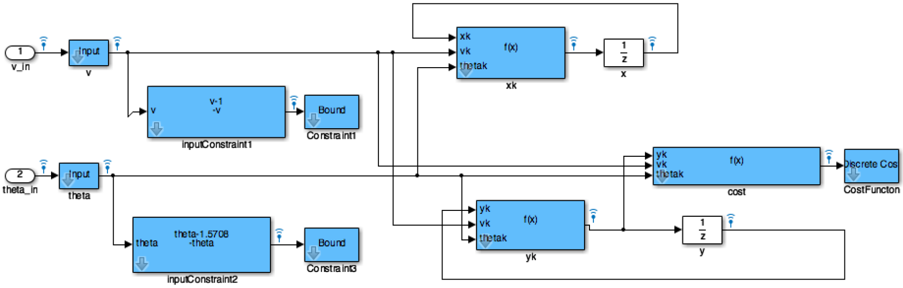
$$H_p = 120, \delta t = 0.5, R_1 = R_2 = 1$$

$$Q = 1, v_{max} = 1, \theta_{max} = \frac{\pi}{2}$$

Two different ways to do this in BLOM



Two different ways to do this in BLOM



How to get BLOM

- http://mpclab.net/Trac/wiki/SVNsetup Here are instructions on how to get SVN and how to get BLOM running
- `svn checkout http://www.mpclab.net/BLOM/ desired_directory`
- Each time you open up BLOM, make sure to get the latest version by typing svn update within that folder (or update through TortoiseSVN)
- On Mac or Linux machines, you may need to compile IPOPT and then run `BLOM_Setup` (http://mpclab.net/Trac/wiki/CompilingIpopt)
- Software: Windows - TortoiseSVN, Mac - command line OR TortoiseHM