Solving Linear Equations with CVX

CVX is a Matlab toolbox for the modeling and solution of convex optimization problems. That may sound intimidating, but it turns out convex optimization applies to many material and energy balances used in process flowsheeting. In particular, the CVX toolbox provides an elegant way to represent and solve the linear material and energy balances encountered in this course.

Contents

Downloading and Installing CVX

Step 1 If you haven't already done so, download and install the current version of Matlab. Follow the instructions at https://oit.nd.edu/software-downloads/

Step 2 Download CVX from http://cvxr.com/cvx/download. Download `cvx-rd.zip` which is the redistributable version for Mac and PC systems. Double-click to unpack the downloaded package into a folder. Once complete you should have a folder name `cvx`. Move that folder into your Matlab working directory.

Step 3 Start Matlab, then in Matlab navigate to the `cvx` directory you created in the previous step. Run the command `cvx_setup` in the Matlab command window. Depending on your setup, you'll see a report describing the installation and testing of CVX.

Example: Solve a simple system of linear equations

This first example demonstrates solution of a simple set of linear equations.

CVX consists of specialized statements that are contained between `cvx_begin` and `cvx_end`. The `variables` statement identifies the mathematical unknowns you wish to compute with CVX. Then the equations are written with a double 'equals' sign == to distinguish them from standard Matlab assignments. Following the `cvx_end` statement, CVX attempts to find a solution for the unknowns that satisfy the given equations.

```
cvx_begin
    variables x y
    3*x + 4*y == 26;
    2*x - 3*y == -11;
cvx_end

display(['x = ',num2str(x)]);
display(['y = ',num2str(y)]);
```

```
Homogeneous problem detected; solution determined analytically.
Status: Solved
Optimal value (cvx_optval): +0

x = 2
y = 5
```

Example: Murphy Example 2.8 -- Ammonia Synthesis

The following model is a solution to Example 2.8 from the course textbook. Variable `X` denotes the extent of reaction.

```
cvx_begin quiet

    variables H1 N1 H2 N2 A2  X

    % Specifications
    N1 == 150;
    H1 == 4*N1;
    N2 == 0.3*N1;

    % Material Balances
    0 == N1 - N2 - X;
    0 == H1 - H2 - 3*X;
    0 ==    - A2 + 2*X;

cvx_end

display(['Extent of Reaction = ', num2str(X)]);
```

```
Extent of Reaction = 105
```

Published with MATLAB® R2014a