Example to Accompany ‘A General, Parallel Implementation of Dantzig–Wolfe Decomposition’

Joseph Rios

September 18, 2012

1 Example

This note accompanies the article entitled “A General, Parallel Implementation of Dantzig–Wolfe Decomposition” submitted to and accepted for publication in ACM Transactions on Mathematical Software [5]. Here, I provide a simple example of how the software works, complete with input and output files. The code is currently available at `http://sourceforge.net/projects/dwsolver`.

For this brief example, I will refer to the software by its command-line executable name: dwsolver. There are several user-supplied options available to dwsolver. Currently there is no callable interface. The program can print timing information, which includes the elapsed CPU time (the sum of all the seconds spent on all cores in the system) and elapsed wall clock time. To aid in debugging models and tracing solution trajectories through the algorithm, several files may be written during the course of solving. The basis may be output to a file at each iteration for the user to investigate how columns entered and exited the basis over the course of iterations. In addition, the reduced master problem may be output to a file at each iteration. After convergence upon an optimum value, the final reduced master problem can be saved to file. Solving this final reduced master problem independently at any later time would provide the optimal solution to the original linear program. As evidence of correctness and an aid in developing input files, several examples are provided with the software. Most of the examples are taken directly from popular textbooks that discuss DW. These include books from Bertsimas and Tsitsiklis [1], Lasdon [4], and Dantzig [2]. There are also two examples from instructors’ websites as well as a larger example taken from this author’s work on air traffic management.

The best way to understand the input and output for the program is to look at a single example in its entirety. For this purpose, Lasdon’s example [4] is used. The

```
Minimize
 objective: - 1 x1 - 1 x2

Subject to
 c1: x1   + 3 x2 <= 30
 c2: 2 x1 + 1 x2 <= 20

end
```

(a) Subproblem 1

```
Minimize
 objective: - 2 y1 - 1 y2

Subject to
 c1: 1 y1        <= 10
 c2:        1 y2 <= 10
 c3: 1 y1 + 1 y2 <= 15

end
```

(b) Subproblem 2

```
Minimize
 objective: - 1 x1 - 1 x2 - 2 y1 - 1 y2

Subject To
 Con1: 1 x1 + 2 x2 + 2 y1 + 1 y2 <= 40

end
```

(c) Master Problem

Figure 1: The two subproblems and master problem for the Lasdon example in LP format.

original problem as given in the textbook is as follows:

$$
\begin{array}{llrrrrll}
\text{minimize} & z = & -x_1 & -x_2 & -2y_1 & -y_2 & & \\
\text{subject to} & & x_1 & +2x_2 & +2y_1 & +y_2 & \leq 40 & (1) \\
 & & x_1 & +3x_2 & & & \leq 30 & (2) \\
 & & 2x_1 & +x_2 & & & \leq 20 & (3) \\
 & & & & y_1 & & \leq 10 & (4) \\
 & & & & & y_2 & \leq 10 & (5) \\
 & & & & y_1 & +y_2 & \leq 15 & (6)
\end{array}
$$

Inequality 1 will be the sole connecting constraint, while Inequalities 2 and 3 will form subproblem one and Inequalities 4, 5, and 6 will form subproblem two. Given this decomposition, the input files written in CPLEX's LP format [3] are provided in Figure 1. Note that in CPLEX's LP format, coefficients of unit value may be omitted and a sign on a coefficient may or may not be spaced from the coefficient. Including spacing as shown in Figure 1 is a convention often used to maintain spacing regularity in a text file. Since dwsolver is a command-line tool and there may be many more than two subproblems, a guide file is used to declare these input files to dwsolver. This guide file tells the program how many subproblem files there are, the names of each subproblem

```
2
sub1.cplex
sub2.cplex
master.cplex
0.0
```

Figure 2: The Lasdon example guide file provided to `dwsolver` with number of subproblems, subproblem file names, master file name, and objective constant value.

file, the name of the master file, and an optional, signed constant to be added to the objective value. For this example, the guide file is implemented as shown in Figure 2.

Given these files, the command line "`dwsolver -g guide_file`" produces the terminal output shown in Figure 3 along with the solution file shown in Figure 4, which provides a mapping of variable names to value assignments for the discovered optimal value. The optimal solution ($-36\frac{2}{3}$) and the accompanying variable assignments are equal to those provided by Lasdon [4].

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
DWSOLVER: Stand-alone Dantzig-Wolfe Decomposition Solver
(C) 2010 National Aeronautics and Space Administration
Covered under GPLv3 with Additional Terms
Compiled using GLPK version 4.44 (slightly modified)
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
The master currently has 3 rows and 1 columns.
No auxiliary variables introduced.  Straight to Phase II.

###  Iteration 0 of phase II ###
################################################
####  Master objective value = -3.200000e+01
################################################

###  Iteration 1 of phase II ###
No columns added, but simplex made progress.
################################################
####  Master objective value = -3.200000e+01
################################################

###  Iteration 2 of phase II ###
################################################
####  Master objective value = -3.666667e+01
################################################

###  Iteration 3 of phase II ###
I think we've converged on an optimum.
################################################
####  Master objective value = -3.666667e+01
################################################
Didn't add any columns?  Let's break.
Done with solving the relaxation...
Printing final master to done.cpxlp...   DONE!
Waiting for subthreads...                DONE!
Printing relaxed solution to file...     DONE!
Freeing subproblem data..                DONE!
Freeing globals...                       DONE!
Master made it to the end. Exiting gracefully, dignity intact.
```

Figure 3: Example terminal output from `dwsolver` for the Lasdon example.

```
x1      8.333333
x2      3.333333
y1      10.000000
y2      5.000000
```

Figure 4: Example solution file from `dwsolver` for the Lasdon example.

References

[1] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, Belmont, Massachusetts, 1997.

[2] George B. Dantzig. *Linear Programming and Extensions.* Princeton University Press, 1963.

[3] IBM. User's manual for CPLEX, 2009.

[4] Leon S. Lasdon. *Optimization Theory for Large Systems.* The MacMillan Company, 1970.

[5] Joseph Rios. A general, parallel implementation of dantzig–wolfe decomposition. *ACM Transactions on Mathematical Software (to appear)*, x(x):xx, 20xx.