Object Oriented Solver
for Linear Complementarity Problems

Florian A. Potra and Cosmin G. Petra[*]

1   OOLCP

OOLCP (Object Oriented solver for Linear Complementarity Problems) is a software package for solving monotone linear complementarity problems. This work was supported by the National Science Foundation under Grant No. 0728878.

OOLCP is developed within the OOQP (Object Oriented Quadratic Programming solver) package. Each of the three fundamental layers of OOQP, *i.e.* interior-point algorithm, problem formulation, and linear algebra layer, has been reimplemented to work in the more general case of monotone mixed linear complementarity problems (introduced below). However OOQP's internal software organization and linear algebra classes for storing and performing operations on vector and matrices have been highly reused.

Given $A \in \mathbb{R}^{(m+n) \times n}$, $B \in \mathbb{R}^{(m+n) \times n}$, $C \in \mathbb{R}^{(m+n) \times m}$, and $b \in \mathbb{R}^{n+m}$, the mixed (horizontal) linear complementarity problem (MLCP) consists of finding the vectors $x \in \mathbb{R}^n$, $s \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ satisfying

$$
\begin{aligned}
xs &= 0 \\
Ax + Bs + Cy &= b \\
x, s &\geq 0.
\end{aligned}
\tag{1}
$$

Such complementarity problems are called monotone if

$$Au + Bv + Cw = 0 \text{ implies } u^T v \geq 0$$

for any vectors $u \in \mathbb{R}^n, v \in \mathbb{R}^n$ and $w \in \mathbb{R}^m$.

The optimization may be viewed as a special case of complementarity. Linear and quadratic programming involving inequalities constraints can be reduced to and often are solved as MLCPs. The complementarity is given by the Karush-Kuhn-Tucker conditions which, in fact, state that the optimal solution to the optimization problem is the solution of a complementarity problem, provided some regularity conditions are satisfied.

1.1   Installing and building OOLCP

Prerequisites of OOLCP include a Linux operating system, a C++ compiler for building the code, a FORTRAN 77 compiler for building dependencies, and standard *make* tools. A copy of OOLCP can be obtained from http://www.math.umbc.edu/~potra/oolcp/oolcp.tar.gz.

The archive file includes a distribution of OOQP. Please check the license terms for OOQP and obtain your own copy from http://pages.cs.wisc.edu/∼swright/ooqp/ before using OOLCP.

Once a copy of OOLCP's source code has been obtain, extract the files using

```
> tar xzvf oolcp.tar.gz
```

For compiling, type

```
> make
```

in your favorite console terminal to build the executables.

1.2 Description of the OOLCP files

In what follows, the path to the OOLCP files will be given relatively to the *src/* directory within OOLCP installation directory.

- *LcpGen/LcpGenSparseMa48.h* and *LcpGen/LcpGenSparseMa48.C* contain the definition and implementation of the *LcpGen* class that implements the OOQP abstract class *ProblemFormulation* and assembles a compatible set of objects (problem's data, variables, residuals and linear system) specific to the MLCP formulation (1). As the name says, it is intended for *sparse* problems and solves the linear systems required by interior-point methods using MA48 linear solver.

- *LcpGen/LcpGenData.h* and *LcpGen/LcpGenData.C* contain the definition and implementation of the *LcpGenData* class that derives OOQP class *Data* and provides methods for performing operations with the MLCP's data: mat-mat and mat-vec multiplications, insertion of problem sub-matrices, computing norms, input/output operations, etc.

- *LcpGen/LcpGenVars.h* and *LcpGen/LcpGenVars.C* are the definition and implementation files for *LcpGenVars* class that specializes OOQP's abstract class *Variables* for the needs of MLCP formulation.

- *LcpGen/LcpGenResiduals.h* and *LcpGen/LcpGenResiduals.C* specializes OOQP's abstract class *Residuals* for OOLCP's formulation (1).

- *LinearSolvers/Ma48Solver/Ma48Solver.h* and *LinearSolvers/Ma48Solver/Ma48Solver.C* defines a class that wraps the functionality of MA48 solver in the format required by OOQP's abstract class *LinearSolver*.

- *QpSolvers/MehrShamSolver.h* and *QpSolvers/MehrShamSolver.C* contains the implementation of a Shamansky variant of Mehrotra's algorithm.

- *LcpGen/LcpGenSparseMehrotraDriver.C* and *LcpGen/LcpGenSparseMehrShamDriver.C* are the driver files (where the *main(. . .)* function is) that solves sparse MLCPs using Mehrotra's algorithm and its Shamansky variant, respectively.

- *LcpGen/LcpGenDriver.h* contains the code that assembles the data, prepares and makes the calls to the appropriate objects, depending on which driver/method is used.

1.3 Using OOLCP

The following executables are created in the installation directory as the result of the building process:

- *lcpgen-sparse-mehrotra.exe* solves MLCPs using the Mehrotra's algorithm. The path to the file containing the problem's specification must be passed as argument. The other arguments supported, *i.e. –version, –printlevel num, –quiet, –verbose*, and *–scale* are identical to the ones supported by OOQP. Please see OOQP user guide for more information.

- *lcpgen-sparse-mehrsham.exe* solves MLCPs using the Shamansky variant of Mehrotra's algorithm. The same parameters as for *lcpgen-sparse-mehrotra.exe* can be used.

1.3.1 C/C++ interface

Interfacing OOLCP with your own C/C++ code is also possible and can be done the same way OOQP is interfaced. Please see the function *lcpgen_solve* from file *LcpGenDriver.h* for an example or consult OOQP user guide.