

線形計画のソフトウェアと LP ファイルによる記述

この資料では，CPLEX や SCIP などのソフトウェアを用いて線形計画問題や整数計画問題を実際に解くための初歩的な方法について解説する。

-
- この資料は，主に「制約付き最適化のプログラムを実際に動かしてみたい」という人向けの，補助資料です．講義の履修や単位の取得に，プログラミングは必要ではありません。
-

1. ソフトウェア

線形計画や整数計画には，商用・非商用ともに多くのソルバーがある．そのうちのいくつかを，以下にあげる．

- SCIP
 - 非商用（アカデミックユースは無償），現在は ver. 3.1.0
 - ダウンロードページ：<http://scip.zib.de/>
- GLPK
 - 非商用（GNU GPL ライセンス），現在は ver. 4.55
 - ダウンロードページ：<https://www.gnu.org/software/glpk/>
- IBM ILOG CPLEX
 - 商用，ver. 12.6
 - ダウンロードページ：
<http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/>
- Gurobi Optimizer
 - 商用，現在は ver. 5.6
 - ダウンロードページ：<http://www.gurobi.com/>

この他に，Matlab のツールボックス linprog や Scilab のツールボックス linpro などでも，手軽に線形計画問題を解くことができる．また，いろいろな最適化ソルバーの性能の比較として Hans Mittelmann 氏の web ページ <http://plato.asu.edu/bench.html> がよく知られており，ベンチマーク実験の結果が継続的に公開されている．

2. LP ファイル形式

最適化問題をソルバーに入力するには，いろいろな方法がある．ここでは，多くのソルバーが対応している LP ファイル形式の文法の基礎を解説する．

2.1. 簡単な例

まず，線形計画問題

$$\left. \begin{array}{ll} \max & 70x_1 + 30x_2 \\ \text{s. t.} & 5x_1 + 2x_2 \leq 80, \\ & 2x_1 + 3x_2 \leq 40, \\ & 3x_1 + 7x_2 \leq 70, \\ & x_1 \geq 0, \quad x_2 \geq 0 \end{array} \right\} \quad (1)$$

を例に，LP ファイル形式の概要をみてみよう．LP ファイルは，テキストファイルであり，問題 (1) を次のように記述する：

ファイル名：‘product1.lp’

```
1  Maximize                \ 最大化問題であることを示す
2  obj:   70 x_1 + 30 x_2    \ 目的関数を書く
3
4  Subject to              \ ここから制約のセクションが始まる
5  r_1:   5 x_1 + 2 x_2 <= 80 \ 制約式に名前を付けることができる
6  r_2:   2 x_1 + 3 x_2 <= 40
7  r_3:   3 x_1
8          + 7 x_2 <= 70    \ 一つの制約式が複数行にまたがってもよい
9
10 Bounds                  \ 変数の上下限値のセクションが始まる
11  0 <= x_1
12  0 <= x_2
13
14 End                    \ ファイルの終わりを表す
```

2.2. 実行方法

SCIP を実行するには，SCIP のコマンドウィンドウで次のようにタイプする：

```
SCIP> read product1.lp
SCIP> optimize
SCIP> write solution product1.sol
```

その結果，ファイル ‘product1.sol’ に最適解が次のように保存される：

‘product1.sol’

```
1  solution status: optimal solution found
2  objective value:                1127.27272727273
3  x_1                            14.5454545454545   (obj:70)
4  x_2                            3.63636363636364   (obj:30)
```

GLPK を用いる場合は，コマンドプロンプトで

```
> glpsol --cpxlp product1.lp -o solution.txt
```

のようにタイプする。

2.3. LP ファイル形式の解説

- バックスラッシュ (\) の右側はコメント扱いになる.
- 目的関数のセクション
 - 最小化問題では ‘Minimize’, 最大化問題では ‘Maximize’ で書き始める (大文字・小文字の別は任意).
 - 変数名は, 16 文字を超えない範囲で, アルファベットと数字を組み合わせる自由につけることができる.
 - 演算子と変数の間や, 係数と変数の間には, 半角スペースを入れる. 目的関数は複数行にまたがってもよい. ただし, 一つの変数名の途中で改行を入れてはいけない.
- 制約のセクション
 - ‘Subject to’ に続いて, 制約式を書き並べる.
 - それぞれの制約式には, 名前をつけることができる. 制約式の左に「名前 + ‘:’」とタイプする.
 - 一つの制約式は, 新しい行から書き始める.
 - 制約式は, 等号 ‘=’ か不等号 ‘<=’ ‘>=’ を用いて記述する. 変数を含む項は, 等号や不等号の左辺にまとめる. 変数を含まない項 (つまり, 定数項) は, 等号や不等号の右に書く.
- 変数の上下限值セクション
 - ‘Bounds’ に続いて, それぞれの変数の上下限值制約を書き並べる.
 - ここに登場しない変数は, 非負変数であるとみなされる (従って, 2.1 節の例では, 実は, ‘0 <= x_1’ と ‘0 <= x_2’ は書かなくてもよい).
 - 上限値と下限値の両方を指定するときは, たとえば ‘2 <= x_1 <= 3.5’ などと書く.
 - 自由変数 (上下限值がない変数) は, ‘-inf <= x_2 <= inf’ と書くか, ‘x_2 free’ と書く (このように書いておかないと, 自動的に ‘0 <= x_2’ という制約が課されてしまう).
- 変数型のセクション
 - 問題に整数変数が含まれるときには, 変数型を指定する (2.1 節の例は線形計画問題なので, このセクションはない).
 - ‘Binary’ に続いて, 0-1 変数の変数名を並べる.
 - ‘General’ に続いて, 整数変数の変数名を並べる. なお, 負の整数値も許す変数の場合には, 「変数の上下限值セクション」でそのように指定しておく必要がある (その指定をしないと, 非負の整数変数とみなされる).
- ファイルの最後には ‘End’ と書く.
- LP ファイル形式の詳細は, たとえば CPLEX のマニュアルの web ページ
<http://pic.dhe.ibm.com/infocenter/cosinfoc/v12r6/index.jsp>
などに情報がある.

2.4. 混合整数計画の例

実数変数 x_1, x_2, x_3 と整数変数 x_4 をもつ混合整数計画問題

$$\left. \begin{array}{ll} \max & x_1 + 2x_2 + 3x_3 + 4x_4 \\ \text{s. t.} & -x_1 + x_2 + x_3 + 10x_4 \leq 20, \\ & x_1 - 3x_2 + x_3 \leq 30, \\ & x_2 - 3.5x_4 = 0, \\ & 0 \leq x_1 \leq 40, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad 2 \leq x_4 \leq 3, \\ & x_4 \in \mathbb{Z} \end{array} \right\} \quad (2)$$

の場合には、LP ファイルは次のように書ける：

```
1  Maximize
2    obj: x1 + 2 x2 + 3 x3 + x4
3  Subject to
4    c1: - x1 + x2 + x3 + 10 x4 <= 20
5    c2: x1 - 3 x2 + x3 <= 30
6    c3: x2 - 3.5 x4 = 0
7  Bounds
8    0 <= x1 <= 40
9    2 <= x4 <= 3
10 General
11   x4
12 End
```

なお、最適解は

$$(x_1, x_2, x_3, x_4) = (40, 10.5, 19.5, 3)$$

である。