LPとグラフと定式化

tokoharu

2014.12.31
(改訂：2016.11.23)

1 LPと主要問題

LP(Linear Programming) とは線形計画問題のことである．

1.1 LP, LPの双対問題

まずはLPの標準的な形を記述する．A は定数行列，b, c は定数ベクトル，ベクトルに対する不等式は全ての要素に対してその不等式が成立することを表す．

LP 標準形

$$\begin{aligned} &\text{maximize} && c^T x && && (1)\\ &\text{subject to} && Ax \le b && && (2)\\ && && x_i \ge 0 && i = 1, \ldots, n && (3) \end{aligned}$$

双対 LP 標準形

$$\begin{aligned} &\text{minimize} && b^T y && && (4)\\ &\text{subject to} && A^T y \ge c && && (5)\\ && && y_j \ge 0 && j = 1, \ldots, m && (6) \end{aligned}$$

ここで, A は $m \times n$ 行列，c, x は n 次元列ベクトル，b, y は m 次元列ベクトルである．

例 :

$$A = \begin{bmatrix} 1 & 1 \\ 5 & 2 \end{bmatrix}, b = \begin{bmatrix} 1 \\ 3 \end{bmatrix}, c = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

のとき，最適解は $x = (1/3, 2/3), \gamma = 4/3$ である．図を描けばわかりやすい．これの双対LP解は $y = (1/3, 1/3), \gamma = 4/3$ である (γ は最適値を表す)．このように主問題に最適解が存在すれば双対問題にも最適解が存在し，主問題の最適値と双対問題の最適値は一致する (強双対定理).

1.1.1 LPと双対LPの変換のための表

最小化問題と最大化問題をそれぞれその双対に変換するときに変換する時には下の表を使うとよい．

最小化問題	最大化問題
変数	制約式
≤	≥
自由	=
≥	≤
制約式	変数
≤	≤
=	自由
≥	≥

不等号に対する左辺と右辺の内容が明記されていないが, 先ほどの標準形で示した内容と同じ向きである．例えば 最小化問題での変数 y には非負制約 (≥) があり, その変数に対応する制約は最大化問題における $Ax \leq b$ である．いま紹介した関係は下の表におけるの 3 行目に対応する．

1.2 最短経路問題と LP

この節では次の 3 つのトピックを紹介する．

1. 自己ループ無し最短経路問題について．
2. LP で解いた時の解と Bellman-Ford で解いた時の解の違い．
3. 2 を実際の問題で気を付ける必要があるか．

1 について．

自己ループが無ければ，最短経路問題は LP の形で書くことができる．発想としては，後述する最小費用流の特殊ケースと思えば良い．さらに，この LP の双対を考えることができ，かなりシンプルな形の式が出てくる．競技プログラミング界隈では，差分制約，俗に牛ゲーと呼ばれる．これは POJ3169(Layout) に登場する牛から来ている．

これ以後 δ_v^+ は頂点 v から出る辺の集合を表し，δ_v^- は頂点 v に入る辺の集合を表すものとする．

LP 主問題 ($|E|$ 変数， $|V|$ 制約)

$$\text{minimize} \quad \sum_{e \in E} c_e y_e \tag{7}$$

$$\text{subject to} \quad \sum_{e \in \delta_v^-} y_e - \sum_{e \in \delta_v^+} y_e \geq 0 \qquad v \neq s, v \neq t \tag{8}$$

$$\sum_{e \in \delta_v^-} y_e - \sum_{e \in \delta_v^+} y_e \geq 1 \qquad v = t \tag{9}$$

$$\sum_{e \in \delta_v^-} y_e - \sum_{e \in \delta_v^+} y_e \geq -1 \qquad v = s \tag{10}$$

$$y_e \geq 0 \qquad e \in E \tag{11}$$

LP 双対問題 ($|V|$ 変数， $|E|$ 制約)

$$\text{maximize} \quad x_t - x_s \tag{12}$$

$$\text{subject to} \quad x_j - x_i \leq c_e \qquad e = (i, j) \in E \tag{13}$$

$$x_v \geq 0 \qquad v \in V \tag{14}$$

2. について．

Bellman-Ford を走らせた場合と LP で解いた場合の差異を考える．この時に重要なのは Bellman-Ford の実装であ

るが，初期値として s のみ距離 0，それ以外の全ての頂点の距離を INF としておき，更新が起きれば順次更新する，という事にする．この Bellman-Ford を走らせた時に負閉路が検出されれば双対問題は解が存在せず，主問題では負の方向へ発散となる．同じく Bellman-Ford を走らせた時に x_t が INF のままなら，双対問題では正の方向へ発散，主問題では解なしとなる．

ここで,LP 双対問題を解いたときの差違に気をつける．気を付けなければいけないのは，s から t の経路に含まれ得ないような負閉路が存在するときである．この時には LP は主問題で負の方向へ発散，双対問題では解なしになる一方，Bellman-Ford では x_t の値が正しく出てしまう．Bellman-Ford を走らせるときに任意の頂点の近傍を調べて更新するなら問題なかっただろう．

したがって，解きたい LP が差分制約の形をしていた時，それを最短経路用のグラフに直して最短経路問題を解くことになるが，この時にはそのグラフでの正確な $s-t$ 最短経路を求めてもおかしくなる場合がある．(著者もこのあたりは怪しいので間違っている可能性があります．間違っている場合には教えてください)

3. について．

（問題が無いケース）牛ゲーという名前のルーツである POJ3169 を見てみる．これをグラフに直すと，頂点 N から任意の頂点は到達可能であることがわかる．したがって，負閉路がどこかに存在すれば，$[1,\ldots,N,\ldots,$ (負閉路を何周もする)$,\ldots,1,\ldots,N]$ という，負閉路を含む経路を構成できる．したがって 2 で危惧されたような s-t 経路に含まれ得ないような負閉路は存在しないことになる．つまり，2 のようなことは考えずに最短経路を解いてしまえば良い．

（問題があるケース）パソコン甲子園 2014 予選問 10(=AOJ0304) を見てみる．これは先程の場合とは異なり，t の指定がない．先頭が 1 であることから，始点は頂点 1 で，頂点 1 は任意の頂点から到達可能であることがわかる．この場合には，ある頂点 v を終点と固定した時に s-v 経路以外にループが出来うる．具体的には，頂点 1 から到達不可能なところに負閉路を置けば良い．したがって，v を終点と固定した時には，s-v 最短経路と最適解が異なってしまう．この問題の場合には，そこに気をつけて最初に負閉路判定をし，負閉路が存在すれば不可能．負閉路がなければ頂点 1 から到達不可能な頂点が存在するかどうかを判定し，存在すれば inf, のようにしなければならない．(... が, どうもジャッジケースにはそのような微妙な入力は入っていないようであるのでどちらでも AC が返ってくるようだ)

1.3 最大流問題と LP

最大流最小カット定理は有名な定理だが，ここではその定式化と，実際に最大流最小カット定理がどのように双対になっているかを見る．また，定式化は [2] を参考にした．

与えられる定数は c_{ij} であり，これは辺 (i,j) に流せる最大流量を意味する．

LP 主問題

新たに辺 (t,s) を追加して x_{ts} を最大化する最大循環流問題として考える．

したがって辺 (t,s) を辺集合 E に付け加えて E' としておく．

この状況下で $|E|+1$ 変数 $|E|+|V|$ 制約の次の式で書ける．

$$\text{maximize} \quad x_{ts} \tag{15}$$

$$\text{subject to} \quad x_e \le c_e \qquad e \in E \tag{16}$$

$$\sum_{e \in \delta_v^- \subset E'} x_e - \sum_{e \in \delta_v^+ \subset E'} x_e \le 0 \qquad v \in V \tag{17}$$

$$x_e \ge 0 \qquad e \in E' \tag{18}$$

E, E' がそれぞれ登場していることに注意.

LP 双対問題

主問題における v に関する不等式制約に対応する双対問題の変数が p であり，主問題における e に関する不等式制約 $(x_e \leq c_e)$ に対応する双対問題の変数が y であるとする．

この時次のような $|E| + |V|$ 変数 $|E| + 1$ 制約の次の式が書ける．

$$\text{minimize} \quad \sum_{e \in E} c_e y_e \tag{19}$$

$$\text{subject to} \quad y_e + p_j - p_i \geq 0 \qquad e = (i, j) \in E \tag{20}$$

$$p_s - p_t \geq 1 \tag{21}$$

$$p_v \geq 0, y_e \geq 0 \qquad v \in V, e \in E \tag{22}$$

これが最小カットになることはなんとなくはわかるが，真面目な証明をしようとすると厳密性に自信がない．ひとまず説明をしようすると次のようになる．

y_e をあまり大きくしたくないことを考えると p で生じる差はできるだけ小さくしたくなり，$p_s - p_t = 1$ の場合が最適ということはわかる．さらに p_s, p_t 以外の数が p に登場しても嬉しくない．なので平行移動させて $p_s = 1, p_t = 0$ としてよい．これで頂点に対応する p_i のうちどれを 1 にしてどれを 0 にするかという問題になり y_e はギリギリのところまで抑えるのがよいのでこれも $0, 1$ になり，結局最小カット問題になる．

1.4 最小費用流と LP

最小費用流も有名であるので LP として書いてみる．最近は最小費用流の双対を用いる問題も複数問確認されているのでそれの扱い方も後で紹介する．

まず定式化のための準備をするが, 今回は通常の最小費用流問題ではなく, 次のような「最小流量制約付き最小費用循環流問題」を考える．この問題では各辺に対して 3 種類の定数が与えられる． これらは c_{ij}, l_{ij}, u_{ij} と表記され，それぞれ (i, j) 間に 1 流すコスト， 辺 (i, j) 間に流す最小流量・最大流量を表す． [1]

通常見られる最小費用流の形では, 特別な頂点 s, t を用意して流量 F を流すときの最小コストを求める形になるため上記の形で扱えないように見えるかもしれない．しかし, これは「$t - s$ 間に辺を張り,$l_{ts} = u_{ts} = F, c_{ts} = 0$ と設定する」ことで上記枠組みに入る同値な問題に変換が可能である．また, 逆に最小流量制約付き最小費用循環流問題も通常の $s - t$ 最小費用流の形でかけることも有名である．詳しくは蟻本を見るのが良いだろう．

まずは最小化問題を LP の形で書いてみる．

$$\text{minimize} \quad \sum_{e \in E} c_e x_e \tag{23}$$

$$\text{subject to} \quad x_e \geq l_e \qquad e \in E \qquad \text{... 双対変数 } a_e \tag{24}$$

$$-x_e \geq -u_e \qquad e \in E \qquad \text{... 双対変数 } b_e \tag{25}$$

$$\sum_{e \in \delta_v^- \subset E'} x_e - \sum_{e \in \delta_v^+ \subset E'} x_e = 0 \qquad v \in V \qquad \text{... 双対変数 } p_v \tag{26}$$

双対変数は上記のような対応になるように a, b, p とするとき, この問題に対する双対問題は次のようになる．

[1] 今回わざわざ最小費用循環流で書いた理由は, 特別な変数が無いため双対をとった後に意味づけをしやすいからである．

$$\text{maixmize} \qquad \sum_{e\in E} l_e a_e - u_e b_e \tag{27}$$

$$\text{subject to} \qquad a_e - b_e + p_u - p_v = c_e \qquad e = (v,u) \in E \qquad \text{... 主変数 } x \tag{28}$$

$$a_e \geq 0, b_e \geq 0 \tag{29}$$

この双対問題を観察すると, 各頂点に対するポテンシャル p の値を固定してみると他の変数 a, b は一意に定まる性質があることに気づく．次に目的関数をみると, 総和の形の要素をひとつずつ（つまり各辺 e ごとに）見るとこれは $p_u - p_v$ に関する折れ線関数で $p_u - p_v = c_e$ を境に変化する形になる．

したがって, そういうタイプの問題を見れば最小費用流で解けるかも, と思うのが良いだろう．具体的な問題例は 3.1, 3.2 で見る．

2 ネットワークフローでの定式化

2.1 Maximum Closure Problem / Project Selection Problem

Maximum Closure Problem (もしくは Project Selection Problem) とは次のような問題である．

N 個の要素がある．最初どの頂点も集合 B に属しているが，これを集合 A に移すことで利益を最大化したい．
要素 i が A に属する時には利得 p_i を得るという情報が与えられる．
さらに 3 つ組の列 (x_j, y_j, z_j) が与えられ，これは x_j が A に属し，かつ y_j が B に属していた時に $z_j(\geq 0)$ だけ損失をすることを意味する．
得られる利得の最大値を答えよ．

この問題に対する解は

$$\sum_{v\in V} \max(0, p_v) - \text{maxflow}(s,t)$$

で求めることができる．
ただし， maxflow(s, t) とは,
1. 要素に対応する点の他に，新しく s, t の頂点を導入した $N+2$ 頂点のグラフの上に,
2. $p_v > 0$ であれば s から v に容量 p_v の辺を張り,
3. $p_v < 0$ であれば v から t に容量 $-p_v$ の辺を張り,
4. x_j から y_j に容量 z_j の辺を張ったときの,
5. s-t 間の最大流の値である．

最小カットを用いて解く問題は，この問題を通して考えると見通しが良くなる場合がある．

Project Selection が適用できる問題例としては下記 3.4 節の内容がひとつ．プログラミングコンテストチャレンジブック（蟻本）内にある問題では,wifi-towers, the year of code jam に適用が可能である．ただし, 前者はストレートに適用するのに対して後者は少し状況がややこしいことに注意する．というのも二部グラフの性質を利用して問題を言い換えているからである．

2.2 有理数フローについて注意

下記 Problems でも紹介するとおり，有理数の最大流問題の解を求めなければならない場合がある．単純に Dinic のアルゴリズムにおいて int を double に変更すれば良いかというと，実際には調整が難しい (過去に泥沼にハマった経験がある)．整数フローに直して解く方が正確性も勝るので良い．

特に，二分探索をするのであれば，分母と分子でそれぞれ整数の変数を持つと実装がしやすい．

3 Problems

3.1 AOJ2230; How to Create a Good Game

問題概要 : DAG が与えられる．DAG の最大長を維持しながら辺に長さを足す．足せる長さの合計の最大値を答えよ．

3.1.1 以前の解法

とりあえず LP として記述してみると，次のような形になる．$|E| + |V|$ 変数， $|E| + 1$ 制約．

$$\text{maximize} \quad \sum_{e \in E} a_e \tag{30}$$

$$\text{subject to} \quad a_e - x_j + x_i \leq -c_e \qquad e = (i, j) \in E \tag{31}$$

$$x_t \leq D \tag{32}$$

$$x_v \geq 0, a_e \geq 0 \qquad v \in V, e \in E \tag{33}$$

この LP に対して双対を取ってみる．

$$\text{minimize} \quad Dp_t - \sum_{e \in E} c_e y_e \tag{34}$$

$$\text{subject to} \quad -\sum_{e \in \delta_v^-} y_e + \sum_{e \in \delta_v^+} y_e \geq 0 \qquad v \in V, v \neq t \tag{35}$$

$$p_t - \sum_{e \in E} y_e \geq 0 \qquad v = t \tag{36}$$

$$y_e \geq 1 \qquad e \in E \tag{37}$$

$$y_e \geq 0, p_t \geq 0 \qquad e \in E \tag{38}$$

これは，p_t を元々のグラフの頂点 t から t' への辺の容量と解釈することで，最低流量 1 の s-t' 最小費用流であると解釈できる．

3.1.2 改訂版

1.4 節の双対の意味づけを思い出せば, この問題はポテンシャル差による関数でかける．

まず,DAG の最大長 $(= D)$ を維持する制約について考える．DAG の始点終点を作り s, t としておけば問題は s, t のポテンシャル差を一定に保つ制約を入れることになる．これは t から s に辺を結び, 双対でいうところのポテンシャル差が $-D$ より小さいときは利得を $-\infty$ として,$-D$ 以上の利得は 0 と設定すればよいため, (t, s) に最大流量 0, 最小流量 $-\infty$, コスト $-D$ の辺を張ることになる．つまり (s, t) に $u_e = \infty, c_e = D$ の辺を張ることと同じ．

次に辺に長さを追加できる制約について考えると, これは双対の意味でポテンシャル差が $-d_e$(2 点間の距離) より大きくできず、それより小さくすると 1 ずつ利得が増えることになるので,$u_e = \infty, l_e = 1, c_e = -d_e$ として辺を張ればよい．

ここまでできれば最小費用循環流でのコスト最小化になり, めでたく最小費用流で解ける．

3.2 Bangkok Regional 2016 J

問題概要 (一次資料は hadrori さんの速報 : https://gist.github.com/hadrori/1178953f90ec9b102ba05178f26d7af4) : 出版社は $N(\leq 200)$ 冊の本を X 日で出版しなければならない．i 番目の本は印刷を開始してから出版までに $A_i(\leq$ 1,000,000) 日かかり，$C_i(\leq$ 1,000,000) だけコストがかかる．しかし，追加で $D_i(\leq 100)$ だけ払うと 1 日短くでき, これを複数回適用可能だが最短で $B_i(\leq A_i)$ 日までしか短縮できない．いくつかの組 (u, v) が与えられて，v の印刷作業は u の出版より後に開始される必要がある．(組の数は最大で $N*(N-1)/2$) N 冊全ての本が X 日以内に出版されるのに必要な費用が最小となる出版の方法について，各本の出版開始日と短縮させる日数を出力せよ．そのような操作が存在しないときは”Impossible”と出力せよ．

解法 : Impossible なケースは割愛。解が存在すればすべての本の出版開始日を定めればそれらの差分で目的関数が定まるので 1.4 節のように考える．

まず,X 日以内という制約について考える．これは新しい開始日頂点と最終日頂点を作って s, t とおいておけば, 最大化問題に対する目的関数に,$p_t - p_s > X$ であれば利得は $-\infty$ となる．これは t, s の間に $c_{st} = X, u_{st} = \infty$ の辺を張ることで表現できる．

次に与えられた組 u, v から生じる目的関数値について考える．もしポテンシャル差が $p_u - p_v < -A_u$ であれば, 通常の印刷をすればよいので何も起こらないので利得は 0 になる. ポテンシャル差が $-A_u \leq p_u - p_v \leq -B_u$ を満たせば, ポテンシャル差が 1 増えると $-D_u$ ずつ利得が増える．そしてそれ以外のときにはできないので $-\infty$ だけ利得がある．

これは折れ線な関数だが, 折れている箇所が二つあり,1.4 の枠組みでは (u, v) 間に 1 本の辺を張るだけでは表現できないことが分かる．しかし,(u, v) 間に 2 本の辺張るものと考えると表現が出来ることがわかる．具体的には,$-A_u$ より小さいと 0, 大きくなると $-D_u$ ずつ利得が増える関数と,$-B_u$ より小さいと 0, 大きくなると $-INF$ ずつ利得が増える関数があると思うと良い．これは v から u へ,1 本目を $c = -A_i, u = d_u$, 2 本目を $c = -B_i, u = \infty$ という辺を張れば良い.（最後の式の $u = d_u$ について, 左辺の u は容量, 右辺の u は頂点を表していることに注意.

3.3 ジョブ割当問題

この問題は [1] から引用した．

問題概要 : n 個のジョブと m 人の作業員がいる．S_i はジョブ i を実行できる作業員の集合である．以下の様な定式化のもとで x_{ij} としてありうる最適解をひとつ求めよ．

$$\text{minimize} \qquad \max_{j=1,\ldots,m} \sum_{i:j\in S_i} x_{ij} \tag{39}$$

$$\text{subject to} \qquad \sum_{j\in S_i} x_{ij} = t_i \qquad\qquad i = 1, \ldots, n \tag{40}$$

$$\tag{41}$$

まず，max が出現した時の常套手段は，それを変数にし，max の構成要素を制約式に持ってくることなので，次

のように式変形をする．

$$\text{minimize} \quad T \tag{42}$$

$$\text{subject to} \quad \sum_{j \in S_i} x_{ij} = t_i \qquad i = 1, \ldots, n \tag{43}$$

$$\sum_{i : j \in S_i} x_{ij} \leq T \qquad j = 1, \ldots, m \tag{44}$$

この式で，ある値 T を決め打ちした時に，目的値が T 以下の解が存在するかどうかは判定可能である． 具体的には，頂点 s と頂点 i の間に容量 t_i の辺を張り，i と $j(\in S_i)$ の間に容量無限大の辺を張り，j と t の間に容量 T の辺を張って，$\sum_{i=1}^{n} t_i$ だけ最大流が流れるかどうかをチェックすればよい．

あとは T を二分探索すれば最適値を見つけることができる．

3.4 POJ3155; Hard Life

問題概要 : 無向グラフ G が与えられる．この部分グラフの中で (辺数) / (頂点数) を最大化されるような部分グラフを求めよ (頂点数 ≤ 100， 辺数 ≤ 1000)

二分探索で解を求めることを考える．すなわち，(辺数)/(頂点数) $> x$ なるような部分グラフが存在するかどうかを判定する．

式変形をすると

$$(\text{辺数}) - x(\text{頂点数}) > 0 \tag{45}$$

なる部分グラフを持つかを判定する問題になる．

(45) 式の最大値が 0 より大きいことを判定することは Project Selection Problem を適用すれば判別できる． [2]

具体的には辺の集合と頂点の集合を合わせた集合から利得最大になるように要素を選ぶ問題と考えて，辺を選ぶと利得が +1 され，頂点を選ぶと利得が $-x$ されると解釈すれば良い．さらに，ある辺を使用すればその両端点は使用しなければならないが，この制約も Project Selection Problem で表現可能である．

したがって (パラメータ x によって作られた Project Selection Problem の解) > 0 を判定する二分探索を実装すれば良い．

謝辞

この文章を書く際，Mi_Sawa さんと uwi さんに文章の添削や助言をして頂きました．この場を借りて感謝いたします．

[2] Project Selection Problem において最適値は 0 以上の値になる．(45) 式の最大値は，部分グラフとして空のグラフを選べると考えれば 0 以上になる．しかし空グラフは元の問題の定義では定義ができない．したがって，最適値が 0 になるような x では元の問題の実行可能でない解である可能性がある．そのため，(辺数) $- x$(頂点数) ≤ 0 という式で考えようとしたり，他の問題でぴったり 0 になるときのグラフを復元しようと思った時には苦労する可能性がある．

編集履歴

2014/10/17	アジア地区予選用のライブラリの一部として書く
	LP，最短経路，最大流，最小費用流, Maximum Closure Problem,
	How to create a good game，ジョブ割当問題を書く
2014/12/14(?)	有理数フロー, Hard Life を追記する。
– 2014/12/31	最短経路の記事を大幅修正. LP と最短経路の橋渡しで怪しいところを明確に記述.
	辺の集合を表す記法としてδを用いることにした
2016/11/23	*ProjectSelection* の例を追加
2016/11/24 – 12/1	最小費用流まわりの改訂および問題の追加
2016/12/1	*AOJ*0304 について調査

参考文献

[1] 組合せ最適化, B. コルテ

[2] 近似アルゴリズム, V.V. ヴァジラーニ

[3] プログラミングコンテストチャレンジブック, 秋葉拓哉 他