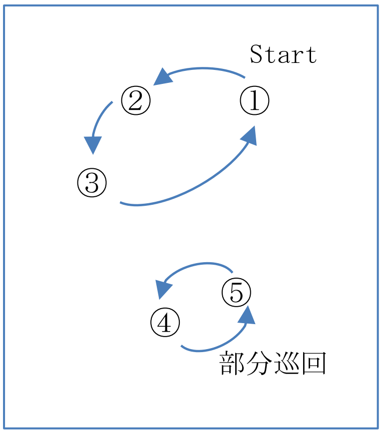
たのしいサイエンス通信
2023年度
3

Miller-Tucker-Zemlin の式



前回「日本一周都道府県庁の旅」の最後に、スタート地点からつながらない部分(部分巡回)ができてしまうものができるといいました。当然前回の

- 各県庁からは他の一つの県庁しか行かない
  $\sum_v x(i,v) = 1$
- 各県庁には他の 1 つの県庁からしか入らない
  $\sum_v x(v,j) = 1$

の 2 条件だけではこういった例のほうが短く出てきます。これを解消するには何通りかあって、出てきた部分巡回を使用禁止とする式を追加する、そしてまた計算をさせて別の部分巡回が出てきたらそれも禁止する式を追加する…を繰り返し、最終的にひとつながりになるまで頑張るといった方法。これはかなりの労力を要します。何せこのような組み合わせが何本あるか、考えただけでも想像を絶します。

まあそのあたりもコンピューターにさせるため、そういったことを数式にして、条件式として設定すればいいのです。当初調べていて最初に見つけた理屈はこちらです。

$$\sum_{i,j \in W} x(i,j) \leq |W| - 1 \ , |W| \geq 2 \, , W \neq V$$

何のことやらとお思いでしょうが、わかりやすく（？）言うと、部分巡回になっている部分、例えば右上の図では④と⑤の２つの県庁で部分巡回になっているときこの中では 2 本の道があるということです。一般には n 箇所での県庁での部分巡回には n 本の道ができることになります。したがって日本全体以外でのとある n 箇所の県庁を含む部分内での道を n 本未満にしようという式となります。

しかしこの方法でもコンピューターといえど相当な計算が必要です。もう少し「楽に」できる式はないだろうか。

それを実現させるための式のひとつとして**Miller-Tucker-Zemlinの式**（以下MTZ）があります。nを県庁数、実変数$1 \leq u(i) \leq n$を用いて

$$u(i) + x(i,j) \leq u(j) + (n-1)(1-x(i,j)) \quad (2 \leq j \leq n)$$

という式です。

わかりにくいかもしれませんが、ゆっくり検証してみます。県庁に番号①から降られているとします。実は$u(i)$は番号ⓘの順番を表すことになります。例えば仮に①をスタートとします。$u(1) \geq 1$となります。今①から②につながる$(x(1,2)=1)$とします。MTZに代入すると$u(1)+1 \leq u(2)$となります。同様に②から③につながるなら$u(2)+1 \leq u(3)\cdots$となっていきます。こうしてすべての県庁が（仮に①からⓝまで順に）つながると、

$$u(1) + (n-1) \leq u(2) + (n-2) \leq \cdots \leq u(n)$$

となり、$1 \leq u(i) \leq n$の条件から$u(1)=1, u(2)=2, \cdots, u(n)=n$ という数が当てはまります。なお i から j につながらないなら$u(i) \leq u(j)+n-1$となり、これはどの数でも成り立ちます。これが上の図のように④と⑤で部分巡回ができてしまうと、$u(4)+1 \leq u(5), u(5)+1 \leq u(4)$という矛盾する式が生まれます。やっていることは数列の不等式だけなのですが、よくこんな式を思いついたなと感心します。高校数学の式だけでここまでできるんだ。高校数学だけでも使いようによってここまで役立つんだと感動を覚えたものです。そりゃこの式に見つけた人の名前が付くわけだ。ひょっとするとあなたも数学で歴史に名を刻むことができるかもしれませんよ？

で、この式を設定して日本一周県庁所在地の旅の計算をした（させた）のですが…それが前回の結果です。そもそもデータ数が多すぎたわけで、実際のところこの巡回セールスマン問題は（データが多い時には）そもそも効率的な解が求まらないといったところで有名な問題です。例えば九州一周（7県）では一瞬で計算してくれたのですが、45都府県と一気に増やしすぎると計算量が爆発的に大きくなります。無料で利用させていただいているNEOS Serverの限界（計算時間10時間の制約あり）ではさすがに無理があるようです。　（逸）

過去の記事は
こちらから