IV－163　ディマンドバスのルート探索問題について

神戸大学工学部　正員　枝村俊郎
神戸大学工学部　正員○森津秀夫
南海電気鉄道　金森哲朗

1．はじめに

ディマンドバスは，バスルートやダイヤをあらかじめ決めておくのでなく，発生した需要に応じてそれらを設定するものである．対象とする需要のパターンによっていろいろなシステムが考えられている．いわゆるmany－to－many型はバスの運行の自由度が大きく，きめの細かいサービスを提供できるが，発生した需要をさばくバスルートを探索する作業は複雑である．かなり需要の少ない場合以外は人間の判断でルートを設定することは困難で，計算機を使って自動化することが必要となる．ここではmany－to－many型のディマンドバスのルート探索のアルゴリズムに関する検討を行う．

2．ディマンドバスのルート探索問題

ルートを求めるときには，利用者に最低限のサービス水準を保証したうえで，バスの利便性の向上と効率的運用を図らなければならない．そこで，明確な基準に従った最適なルートを選ばなければならない．Psaraftis[1]はルート長と乗客の待ち時間，乗車時間で目的関数を構成し，需要の発生順序と実際の乗車順序，降車順序に関する制約と，輸送定員の制約の下で，最適なルートを探索するD.P.の手法を用いたアルゴリズムを提案している．しかし，乗客にとっては乗降の順序はどうでもよく，保証してもらいたいものは，バスが来るまでの待ち時間や目的地に着くまでの時間であろう．そこで，ここでは順序制約をより合目的的な時間制約に置き換えた問題を考えることにする．なお簡単のために，バスが1台だけで，バスが出発してからは新たな需要を追加しないスタティックな場合を扱う．

対象とする問題を簡単に表わすと，次のようになる．

$$\min \ Z = w_1 t + w_2 \sum_i \{\alpha t_i^w + (2-\alpha) t_i^r\} \qquad (1)$$

$$\text{s.t.} \quad p \le C \ , \quad t_i^w \le T_i^w \ , \quad t_i^r \le T_i^r \ , \quad t = \max_i \{t_i^w + t_i^r\} \qquad (2),(3),(4),(5)$$

ここに，t_i^w，t_i^rはそれぞれ乗客iの待ち時間と乗車時間，T_i^wとT_i^rはそれに対する制約値，tはルート長，pは同時にバスに乗っている乗客の数の最大値，Cはバスの定員，そしてw_1，w_2，αは各項の重みを表わすパラメータである。式(1)～(5)では決定変数を明確にしていないが，式(1)はルート長と待ち時間，乗車時間の加重和の最小化を表わす目的関数である。式(2)は定員の制約で，式(3)，(4)は待ち時間と乗車時間に関する制約である．

3．バスルート探索のアルゴリズム

バスルートは，各乗客の乗車地点へ行くことと，それぞれの降車地点へ行くことを連ねたものである．その中で，制約条件を満たし，目的関数を最適にするものを選べばよい。このような，ある条件下で最適なバスルートを探す問題に対しては，探索木を使った解法[2]がある．これは，バスOD需要が与えられたとき，定められた起終点間を結ぶ路線を評価関数値の大きい順に列挙するものである．ここでの問題も，これと同じような考え方で解くことができる。

各乗客はバスを待っているか，乗車しているか，すでにバスから降りているかのどれかの状態にある。これを$x_i = 1, 2, 3$で表わすと，バスの移動は$x_i \le 2$であるiについて，$x_i \leftarrow x_i + 1$とすることで表現できる．そこ

で最初にバスを出発点におき，$x_i \leftarrow x_i+1$ とするステップを繰り返し．各ステップで実行可能性の検査と最適性の検査を行い，分枝とbacktrackingを行うことにする．

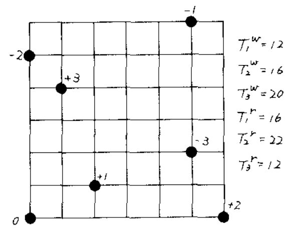
実行可能性は次のようにして調べる．いま，バスが地点kにいるとし，乗客iの乗車地点と降車地点を地点j，m，各地点間の最短所要時間を d_{jm} などとすると，以後にバスが乗客iに対する最適なルートをとったときの待ち時間と乗車時間は次のようになる．

$$t_i^{w'}=\begin{cases} t^k+d_{kj} & (x_i=1\text{のとき}) \\ t_i^w & (x_i=2,\ 3\text{のとき}) \end{cases} \qquad (6)$$

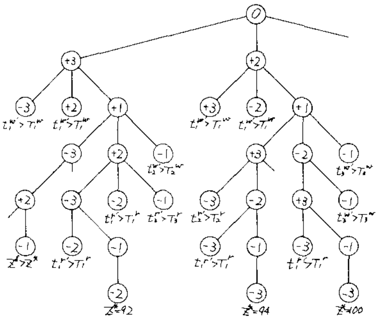
$$t_i^{r'}=\begin{cases} d_{jm} & (x_i=1\text{のとき}) \\ t^k+d_{km}-t_i^w & (x_i=2\text{のとき}) \\ t_i^r & (x_i=3\text{のとき}) \end{cases} \qquad (7)$$

ただし，t^k は地点kまでのルート長である．これから，$t_i^{w'}>T_i^w$ または $t_i^{r'}>T_i^r$ となる乗客があれば，それ以後の分枝が不要であることがわかる．また定員の制約は $x_i \leftarrow 2$ とする分枝の際に条件を満たしているか調べる．

式(6)，(7)を使えば，$t'=\max_i\{t_i^{w'}+t_i^{r'}\}$ とし，目的関数の下限値を $Z^l=w_1t'+w_2\sum_i\{\alpha t_i^{w'}+(2-\alpha)t_i^{r'}\}$ で求めることができる．暫定最適解が得られているときには，この下限値と暫定最適解の目的関数値との比較を行えばよい．



図－1　例　題



図－2　探索木

図－1の簡単な例題を解いてみる．各リンクの長さはすべて1とし，乗客iの乗車地点を$+i$，降車地点を$-i$で表わす．定員の制約を考えず，$w_1=w_2=\alpha=1$ として解いたときの探索木の一部を示すと図－2のようになる．図－2の Z^* は，その段階での暫定最適解の目的関数値である．最適解は，0→+3→+1→+2→−3→−1→−2のルートである．

4．おわりに

ここではディマンドバスのルート探索問題に対し，探索木を使った解法を示した．より現実的な問題を考えるならば，随時新しい需要を受け入れるダイナミックな問題になり，バスも複数を走らせることになる．そのためにはアルゴリズムの改良が必要であるが，とくに複数のバスを用いるとき，乗客をどのバスに割り当てるのが最適かを決めることが必要となる．また待ち時間や乗車時間の制約値をどのように与えればよいかも検討しなければならないであろう．さらに，乗客数が多くなれば，ルート探索に要する計算時間が指数関数的に増加することが予想され，妥当な近似解を短い計算時間で求めることが必要になると考えられる．そして，そのような近似解法の作成には，最適ネットワーク問題での近似解法の作成方法が参考になると思われる．

参考文献

1) Psarftis, H.N. : A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem, Transportation Science, Vol.14, No.2, pp.130～154, May, 1980.

2) 枝村俊郎・森津秀夫・松田　宏・土井元治：最適バス路線網構成システム，土木学会論文報告集，第300号，pp.95～107，1980年8月．