Approximate Dynamic Programming のノート

2013 年 5 月 15 日

第 1 章

第 2 章

Some Illustrative Models

1. Deterministic Problems
2. Stochastic Problems
   the distribution is known.
3. Information acquisition problems
   the distribution is not known.

2.1 Deterministic Problems

2.1.1 The Shortest Path Problems

I the set of nodes (intersections) in the network,
L the set of links (i, j) in the network,
$c_{ij} > 0$ the cost (typically time) to drive from i to j, $i, j \in I, (i, j) \in L$,
I_i^+ the set of nodes j for which there is a link $(i, j) \in L$,
I_j^- the set of nodes i for which there is a link $(i, j) \in L$.

node $i \in I$ にいる旅行者は, $j \in I_i^+$ のノードに移動することが出来る. スタート node q から目的 node r に最小コストで移動するための経路は何か.

v_j the minimum cost required to get from node j to r.

v_j を求める. $v_r = 0$ であることは分かる. v_j^n を $n = 1, 2, \ldots$ 回目の繰り返しによる v_j の推定値としよう. 以下のアルゴリズムで v_j を求めることができる.

0. $v_{j\neq r}^0 = M, v_r^0 = 0$, M:十分大きな値, $n = 1$ とする.
1. すべての $i \in I$ について, 適当な順番で v_i^n を以下の式で求める.

$$v_i^n = \min_{j \in I_i^+} \left(c_{ij} + v_j^{n-1} \right) \tag{2.1}$$

v_j^n が求まっていれば, 右辺の式中の v_j^{n-1} を置き換えて良い.

2. $v_i^n < v_i^{n-1}$ なる i があるならば, n を $n+1$ として 1. に戻る. ないならば, 終了する.

これには無駄が多い. もう少し標準的なアルゴリズムを考えよう. 素朴に考えて, ノード r に移動できるノードから順に更新していけば良さそうである. 以下では候補リストと書いているが, 純粋なリストと考えないように. キューを使うと良さそうである.

0. $v_{j\neq r}^0 = M, v_r^0 = 0$, M:十分大きな値, 候補リスト $C = \{r\}$, $n = 1$ とする.
1. 候補リストの左（一番上）から, node $j \in C$ を選ぶ.
2. j に移動できる node $i \in I_j^-$ のすべてに対して,
   1. $\hat{v}_i = c_{ij} + v_j$.
   2. $\hat{v}_i < v_i$ ならば let $v_i = \hat{v}_i$ として, $i \notin C$ ならば C の後ろに i を追加する.
3. j を C から取り除く. C が空でなければ, 1. に戻る. 空であれば終了する.

ほとんどすべての離散動的計画は最短経路問題と見做すことが出来る. 各ノード i がシステムの離散的な状態を表す. スタートノード q が初期状態であり, エンドノード r は任意の時刻 T での任意の状態と見做せる. $T \to \infty$ を考えることもできるが, その場合は Discount factor を含めることになるだろう.

2.1.2 The Discrete Budgeting Problem

T 個のタスクの集合 $\mathcal{T}$ に対する予算 R の配分を考える. 各タスク $t \in \mathcal{T}$ について, 割り当てる予算を a_t, 得られる報酬を $C_t(a_t)$ とする. 次の量を最大化する a の組み合わせ $\{a_t\}_{t\in\mathcal{T}}$ を求める.

$$\sum_{t\in\mathcal{T}} C_t(a_t) \tag{2.2}$$

但し, 予算に関する制約条件

$$\sum_{t\in\mathcal{T}} a_t = R \tag{2.3}$$

を満たす必要があり, さらに割当可能な予算は正

$$a_t \geq 0, \forall t \in \mathcal{T} \tag{2.4}$$

とする. これら 2 つの条件の下で報酬を最大化する問題を budgeting problem という.

$t = 1, ...T$ と順次タスクに予算を割り当てていくことを考える. 結局,

R_t t 以降のタスクに割当可能な予算

$V_t(R_t)$ 予算 R_t を t 以降のタスク $t, t+1, \ldots, T$ に割り当てた際の報酬総和の最大値 (以下では価値と呼ぶ).

という量を考え,$V_t(R_t)$ を求める問題になる. 動的計画法の言葉では, R_t は状態変数, a_t はアクションと呼ばれる. 今の問題では,

$$R_{t+1} = R_t - a_t =: R^M(R_t, a_t) \tag{2.5}$$

となる．R^M は状態遷移を規定する関数であり, システムモデルと呼ばれる．次の最適性方程式 (optimality equation) が成立する．

$$V_t(R_t) = \max_{0 \le a_t \le R_t} \{C_t(a_t) + V_{t+1}(R^M(R_t, a_t))\} \tag{2.6}$$

最後のタスク T の次のタスク $T+1$ は存在しないので,

$$V_T(R_T) = \max_{0 \le a_T \le R_T} C_T(a_T) \tag{2.7}$$

となる. これは便宜上 $V_{T+1}(R) = 0$ と当てはめて考えればよい. $V_T(R_T)$ が求まれば $t = T-1, T-2, \ldots, 1$ の順に $V_t(R_t)$ を求められることが分かる．解が存在する条件は, $C_t(x)$ が有限であることのみであり, ここで説明した方法は budgeting problem の一般的な解法である．

計算量は, 計算結果を適切に保持していれば,

$$\mathcal{O}\left(size(R)^2 \times T\right) :$$

となるはずである．

2.1.3 The Continuous Budgeting Problem

アクションが離散的な量の場合は a, 連続的な場合は x を用いることにする．連続量の budgeting problem を考える．特に, 次の仮定を置いた場合の解析解について説明する:

$$C_t(x) = \sqrt{x}. \tag{2.8}$$

これは読めばすぐ分かるので省略する．

2.2 Stochastic Problems

2.2.1 Decision Trees

決定木（けっていぎ）

2.2.2 A Stochastic Shortest Path Problem

経路のコストが確率的に定まる. どのタイミングで定まるかに応じて, 問題が変わる．

次のノードに移動するコストが確率的な場合:

$$v_i^n = \min_{j \in I_i^+} \mathrm{E}\{c_{ij}(W) + v_j^{n-1}\}, W : \mathrm{RandomVariable}. \tag{2.9}$$

次のノードに移動するコストは確定的であるが, それ以降は確率的である場合:

$$v_i^n = \mathrm{E}\left[\min_{j \in I_i^+}\{c_{ij}(W) + v_j^{n-1}\}\right], W : \mathrm{RandomVariable}. \tag{2.10}$$

確率過程と条件付き確率を用いて表現できそうだ．

2.2.3 The Gambling Problem

ギャンブラーが, 全 N 回の賭けを行う. 各回にどれだけの資金を賭けるべきか.

p, q 各回の勝ち負けの確率. $q = 1 - p, q < p$.

S_n $n = 0, 1, \ldots, N$ 回目の賭け後の資産, この問題では状態と呼ぶ.

a_n n 回目の賭け金額. 離散的な値とし, $a_n \leq S_{n-1}$ という制約を課す.

W_n n 回目の賭けにギャンブラーが勝つ場合,1 負ける場合 0 となる確率変数.

$V_n(S_n)$ n 回目の賭け後に S_n の資産を保持していることの価値. この価値は, 将来的に得られる価値の期待値とするもので, 最終時点 N での価値については $\ln(S_N)$ と定義する.

$V_0(S_0)$ を最大化するような $\{a_n\}_{n\geq 0}$ を求めたい. これは金融工学で出てくる, デリバティブ価格の評価の話と同一である.

S_n, V_n に関する漸化式は次のようになる:

$$S_{n+1} = S_n + a_{n+1}W_{n+1} - (1 - W_{n+1})a_{n+1} =: R^M(S_n, a_{n+1}), \tag{2.11}$$

$$V_n(S_n) = \max_{0\leq a_{n+1}\leq S_n} \mathrm{E}\left[V_{n+1}(R^M(S_n, a_{n+1})) \mid S_n\right] \tag{2.12}$$

$$= \max_{0\leq a_{n+1}\leq S_n} \mathrm{E}\left[V_{n+1}\left(S_n + a_{n+1}W_{n+1} - (1 - W_{n+1})a_{n+1}\right) \mid S_n\right] \tag{2.13}$$

価値 $V_n(S_n)$ は,n 回目の賭けが終了した時点での情報に基づいて, $n + 1$ 回目以降の賭けを確率的に生じるものと見做した期待値を最大化するものとして表現している.

n の大きい値から 0 に向けて考える. N のとき $V_N(S_N) = \ln S_N$ である. $N - 1$ のとき,

$$V_{N-1}(S_{N-1}) = \max_{0\leq a_N\leq S_{N-1}} \mathrm{E}\left[\ln\left(S_{N-1} + a_N W_N - (1 - W_N)a_N\right) \mid S_{N-1}\right] \tag{2.14}$$

$$= \max_{0\leq a_N\leq S_{N-1}} \left[p\ln\left(S_{N-1} + a_N\right) + (1 - p)\ln\left(S_{N-1} - a_N\right)\right] \tag{2.15}$$

max 内の関数は a_N について凸なので, 極値を調べることで max を与える a_N と解析解を求めることができる. 結果, a_n と V_n が各 n で同様の関数型となることと最適解が $p > 0.5$ を前提とすることが分かる.

2.2.4 Asset Valuation

2.2.5 The Asset Acquisition Problem I

時点 t にインターバル $t+1$ に使用される分の products を x 購入する. x は一般には連続量であり, 多次元ベクトルとなりうる.

R_t assets on hand at time t before we make a new order decision, and before we have satisfied any demands arising in time interval t,

x_t amount of products purchased at time t to be used during time interval $t + 1$,

$\hat{D}_t$ random demands that arise between $t-1$ and t,
p^p, p^s 固定の purchase 価格, sell 価格,
$C_t(x_t)$ the amount we earn between $t-1$ and t, including the decision we make at time t.

$$R_{t+1} = R_t - \min(R_t, \hat{D}_t) + x_t \tag{2.16}$$
$$C_t(x_t) = p^s \min(R_t, \hat{D}_t) - p^p x_t \tag{2.17}$$

period t で要求される $\hat{D}_t$ に応える（売り出す）前の量として R_t を定義していることに注意する. 後のセクションのためにこのように定義しておくとのこと. period と interval という複数のおそらく同一の意味の言葉が出てきていたり, その定義が明確でないので分かりにくい.

区間 $(t-1, t]$ をインターバル t と呼び, R_t は $t-1$ 直後の時点での在庫量, $\hat{D}_t$ はインターバル t 内, 厳密に考えると $(t-1, t)$ で発生する需要の総和で, x_t は $\hat{D}_t$ を満たした直後の時点 t に購入する量ということになるだろう. 在庫量の変化の流れを R_t を用いて表すと, 次のように考えていることになる.

$$R_t \to (R_t - \hat{D}_t) \to (R_t - \hat{D}_t + x_t) = R_{t+1}$$

$C_t(x_t)$ の定義も腑に落ちる.

Contribution の総和を最大化するという問題については, ベルマン方程式によって解くことが出来る. R_t が状態変数であり, $V_t(R_t)$ を価値関数と定義して, これを最大化する.

$$V_t(R_t) = \max_{x_t} \left(C_t(x_t) + \gamma \mathrm{E}\{V_{t+1}(R_{t+1})\}\right) \tag{2.18}$$

これは, t 時点までの情報を $\mathcal{F}_t$ として条件付き確率を用いつつ, $\hat{D}(t)$ も明示して書くと,

$$V_t(R_t, \hat{D}_t \mid \mathcal{F}_t) = \max_{x_t} \Big(C_t(x_t, \hat{D}_t) + \gamma \mathrm{E}\{V_{t+1}(R_{t+1}, \hat{D}_{t+1} \mid \mathcal{F}_t)\}\Big) \tag{2.19}$$

となり, $\hat{D}_{t+1}, \ldots$ を通して確率的な量の期待値を考えることになることが分かる.

2.2.6 The Asset Acquisition Problem II

多くの asset 取得問題においては需要の他にも不確実性の原因となるものがある. まずは価格である. さらに, 保持している量に対する外生的な増減がありうる.

需要と供給に関連する記号を定義する.

x_t^p asset purchased(acquired) at time t to be used during time interval $t+1$,
x_t^s amount of assets sold to satisfy demands during time interval t,
x_t (x_t^p, x_t^s),
R_t resource level at time t before any decisions are made,
D_t demands waiting to be served at time t.

在庫が無かったり需要が無ければ売れない, ということで $x_t^s \le \min(R_t, D_t)$ としておく.

D_t の定義をみると, t まで serve するのを待たされている需要, とある. 前の節でみたように, $(t-1,t)$ に発生する需要の総和であり, それらは t 丁度に満たされると解釈しようということかと思ったが, 需要が累積することを言いたいのかもしれない.

次は価格である.

p_t^p　market price for purchasing assets at time t,
p_t^s　market price for selling assets at time t,
p_t　(p_t^p, p_t^s).

そして外生的な変動を表す量.

$\hat{R}_t$　exogenous changes to the asset levels on hand that occur during time interval t,
$\hat{D}_t$　demand for the resources during time interval t,
$\hat{p}_t^p$　change in the purchase price that occurs between $t-1$ and t,
$\hat{p}_t^s$　change in the selling price that occurs between $t-1$ and t,
$\hat{p}_t$　$(\hat{p}_t^p, \hat{p}_t^s)$.

価格そのものを直接的に確率変数とするのではなく, 変化分を確率的なものとして定式化している. 需要についてもどうも増加分を確率的なものとして定義していると解釈するのが適切にみえる.

これらはまとめて扱うと見通しが良い. 外生的な変動を W_t, 状態を S_t とする;

$$W_t = (\hat{R}_t, \hat{D}_t, \hat{p}_t) \tag{2.20}$$
$$S_t = (R_t, D_t, p_t) \tag{2.21}$$

状態変数の transition を generically に表す. システムモデルと呼んだり, 遷移関数 (transition function) と呼んだりするものである.

$$S_{t+1} = S^M(S_t, x_t, W_{t+1}) \tag{2.22}$$

具体的な遷移関数の形を書いてみよう.

$$R_{t+1} = R_t - x_t^s + x_t^p + \hat{R}_t, \tag{2.23}$$
$$D_{t+1} = D_t - x_t^s + \hat{D}_{t+1}, \tag{2.24}$$
$$p_{t+1}^p = p_t^p + \hat{p}_t^p, \tag{2.25}$$
$$p_{t+1}^s = p_t^s + \hat{p}_t^s. \tag{2.26}$$

1 期間の貢献関数

$$C_t(S_t, x_t) = p_t^s x_t^s - p_t^p x_t^p. \tag{2.27}$$

最適意思決定はベルマン方程式を解けば求まる:

$$V_t(S_t) = \max_{x_t} \left(C_t(S_t, x_t) + \gamma \mathrm{E}\{V_{t+1}(S^M(S_t, x_t, W_{t+1}) \mid \mathcal{F}_t)\} \right) \tag{2.28}$$

2.2.7 The Lagged Asset Acquisition Problem

将来の各時点での需要に対して前もって asset を買うことを考えよう. 具体例としては, 旅行代理店がホテルを先に予約購入したり, 航空券を買うような場合がある.

$x_{t,t'}$ assets purchased at time t to be used to satisfy demands that become known during time interval between $t'-1$ and t',

x_t $(x_{t,t+1}, x_{t,t+2}, \dots,) =: (x_{t,t'})_{t' \geq t}$,

$\hat{D}_t$ demands for the resources that become known during time interval t,

$R_{t,t'}$ total assets acquired on or before time t that may be used to satisfy demands that become known between $t'-1$ and t',

R_t $(R_{t,t'})_{t' \geq t}$.

$\hat{D}_t$ の需要に対して $x_{t,t}$ を反映させることはできないものとする. 在庫は前もって必要ということである. 出荷に時間がかかると考えてもいいだろう. 制約 $x_{t,t} = 0$ を課すということである.

遷移関数は

$$R_{t+1,t'} = \begin{cases} \left(R_{t,t} - \min(R_{t,t}, \hat{D}_t)\right) + x_{t,t+1} + R_{t,t+1}, & t' = t+1 \\ R_{t,t'} + x_{t,t'}, & t' \geq t \end{cases} \tag{2.29}$$

貢献関数は, 価格を p^p, p^s も定義されているとして,

$$C_t(R_t, \hat{D}_t) = p^s \min(R_{t,t}, \hat{D}_t) - \sum_{t'>t} p^p x_{t,t'} \tag{2.30}$$

最適行動 (価値を最大化する $x_{t,t'}$ の決定) については, やはりベルマン方程式を解けばよい. が, 本節の x_t や R_t は多次元ベクトルであり, すべての状態と行動を数値的に列挙するのは事実上不可能であるところが, 前節までと異なる.

2.2.8 The Batch Replenishment Problem

2.2.9 The Transformer Replacement Problem

2.2.10 The Dynamic Assignment Problem

2.3 Information Acquisition Problems