東京大学 工学部 計数工学科/物理工学科

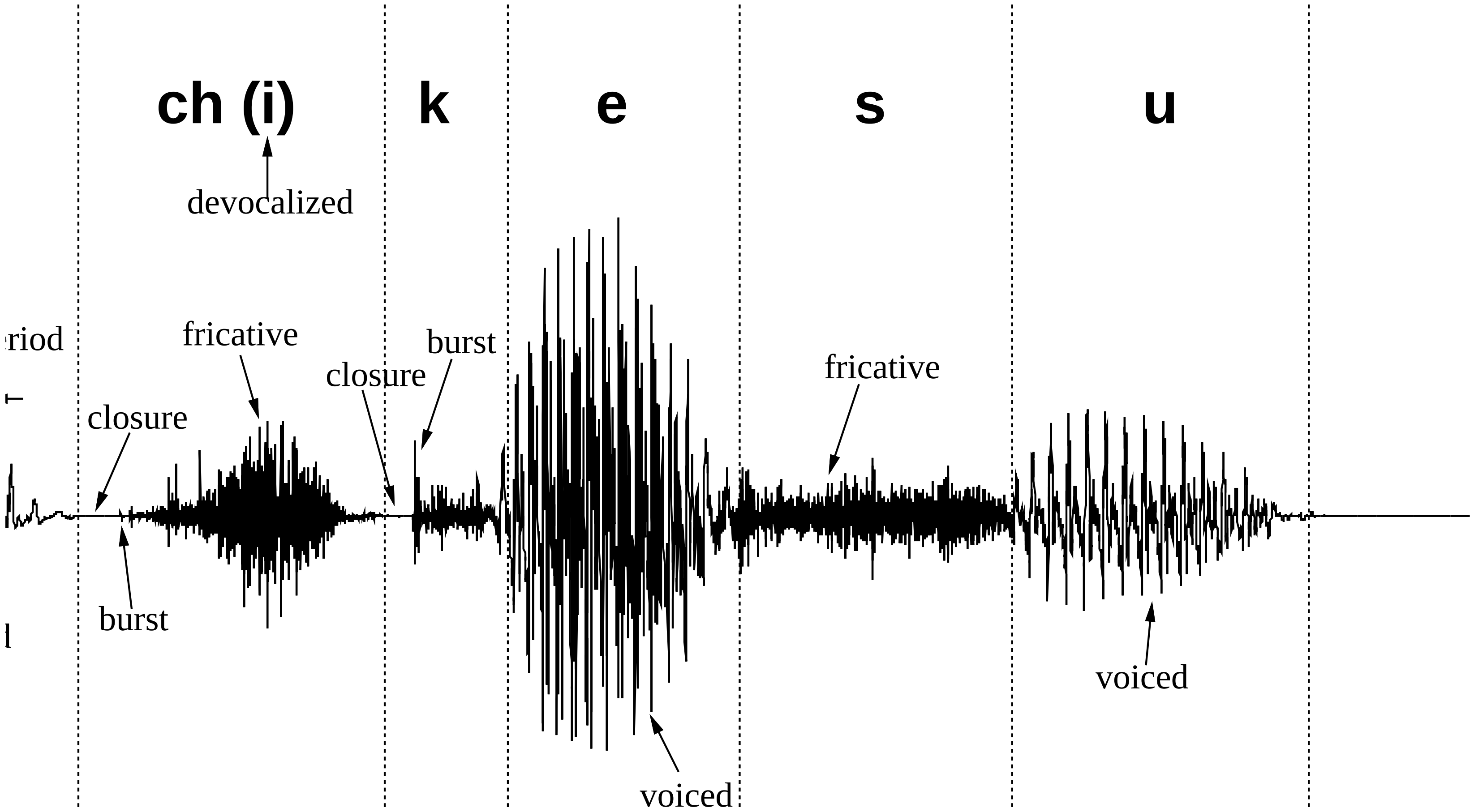
応用音響学：音声認識 – DP matching

嵯峨山 茂樹 <sagayama@hil.t.u-tokyo.ac.jp>
東京大学 工学部 計数工学科 http://hil.t.u-tokyo.ac.jp/

参考文献

- 春日正男ら「音声情報処理」 コロナ社
- 古井 貞煕「音声情報処理」森北出版
- 中川 聖一「確率モデルによる音声認識」コロナ社
- 中川 聖一「パターン情報処理」丸善
- 北 研二・中村 哲・永田 昌明「音声言語処理」森北出版
- C.S.Myers, L.R.Rabiner: "Connected Digit Recognition Using a Level-Building DTW Algorithm," IEEE Trans. Acoust., Speech, Signal Processing, Vol. 29, No. 3, pp. 351–363, 1981.
- H.Ney: "The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition," IEEE Trans. Acoust., Speech, Signal Processing, Vol. 32, No. 2, pp. 263–271, 1984.

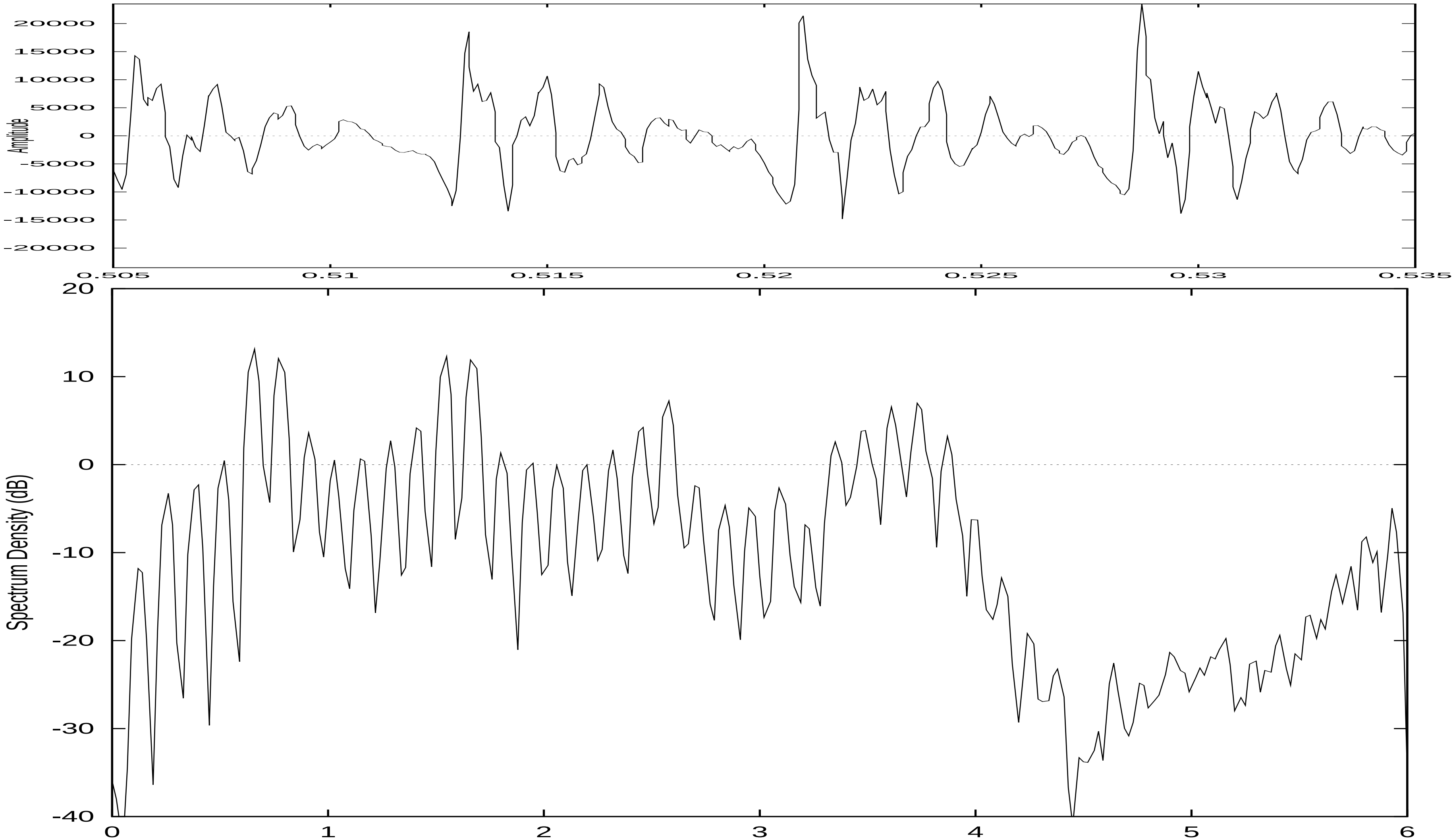
音声波形の例 (男声 単語)



Speech Waveform /uchikesu/

図1. 音声波形の例「打ち消す」

音声の短時間スペクトル(periodogram)の例



短時間音声波形とスペクトルの例 (男声 /a/)

音声時間パターンの変動

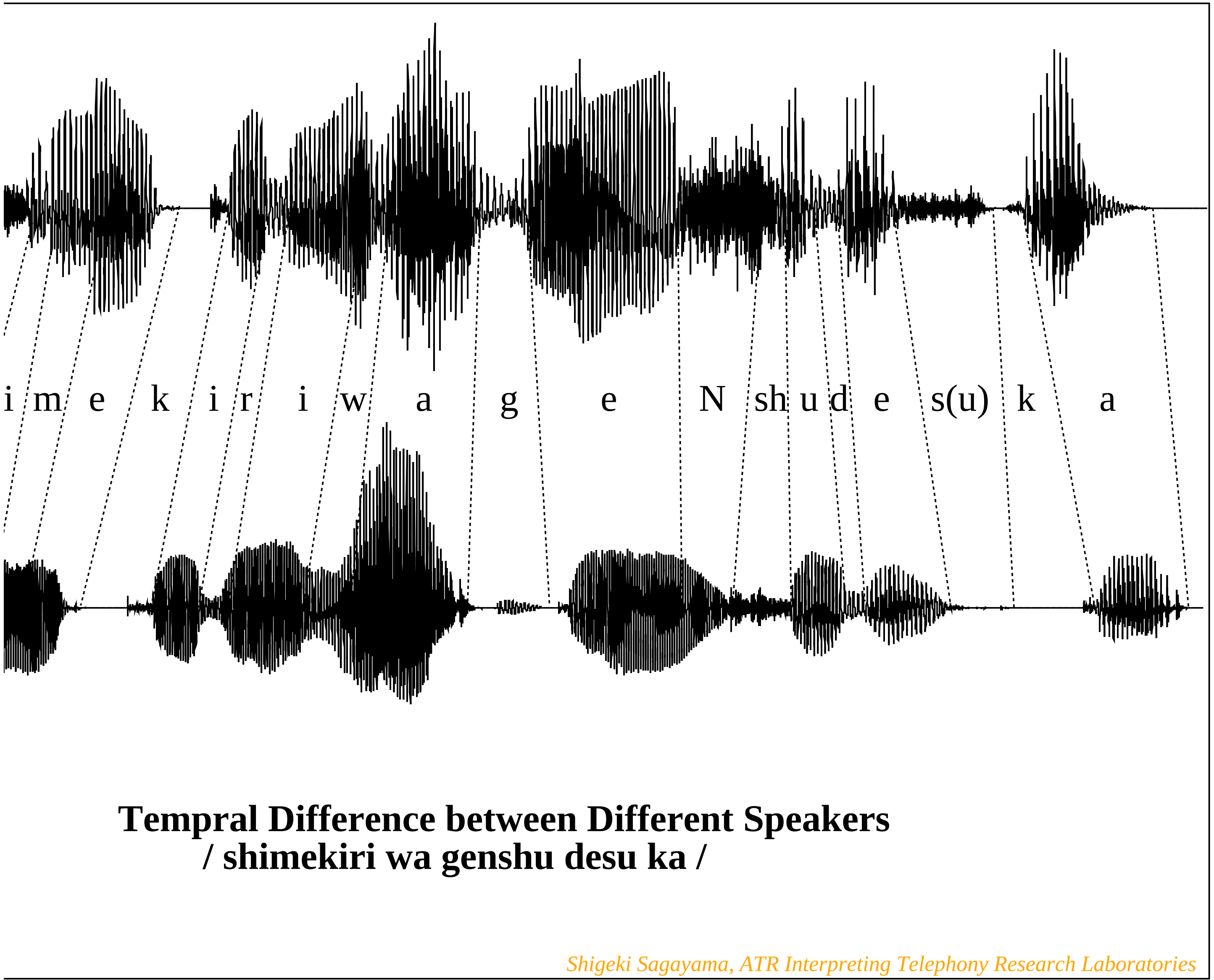


図2. 日本語「締切は厳守ですか」の男女2名の音声波形

DP マッチング (動的時間軸伸縮 DTW)

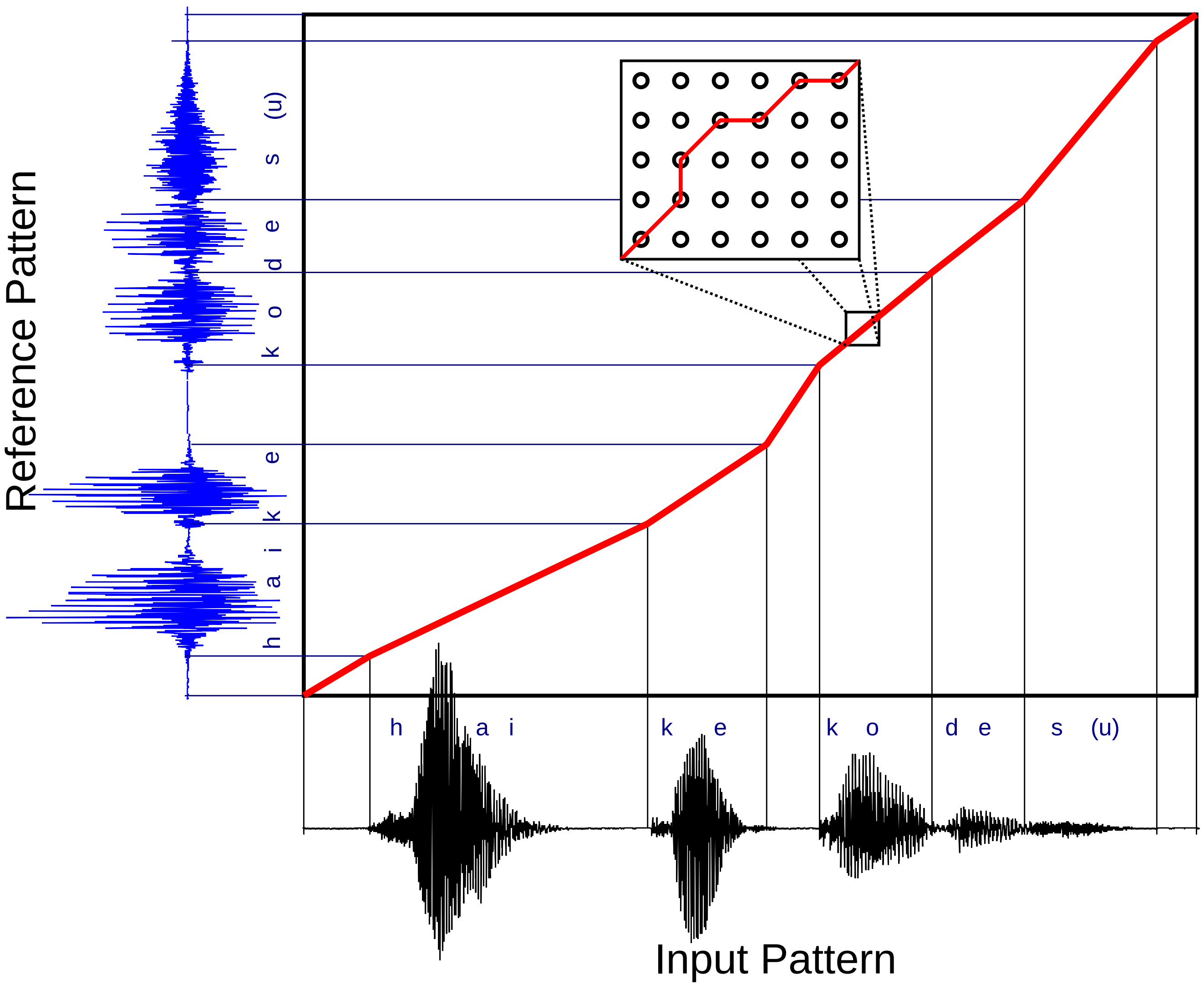


図3. 入力パターンと標準パターンの間の DP matching (Dynamic Time Warping)

DPによる孤立発声単語音声認識の構成

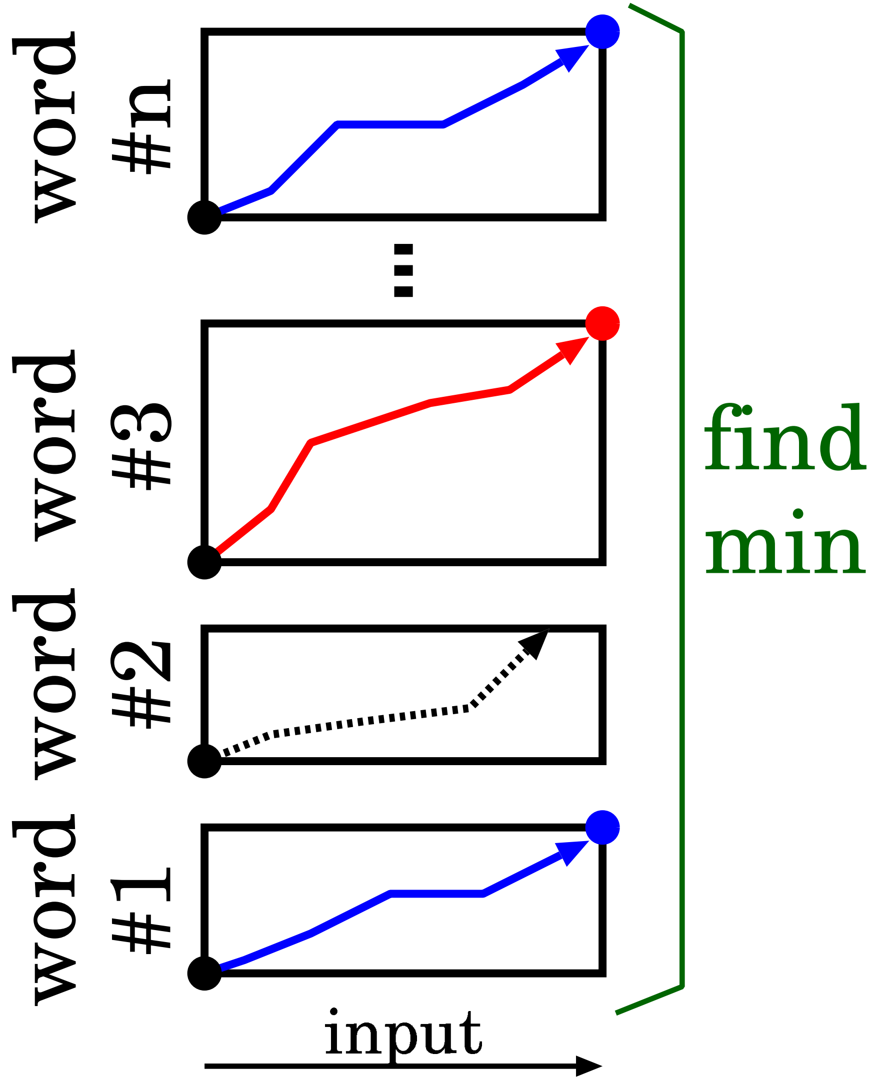


図4. n 個のテンプレートとDPマッチングを行い、累積距離が最小の単語を認識結果とする。

スペクトルマッチング距離

- 瞬時スペクトル間の距離尺度

$$\text{距離} = d(\text{スペクトル}1,\ \text{スペクトル}2)$$

- 距離尺度の例
  - 板倉・齋藤距離
  - COSH 尺度
  - WLR 尺度(重みつき尤度比)
  - PWLR 尺度(パワー重みつき尤度比)
  - LPC ケプストラム間ユークリッド距離
  - LPC 重みつきケプストラム間ユークリッド距離
  - Mahalanobis距離 (→ HMMへ発展)

動的時間整合(DTW)

- DTW : Dynamic Time Warping (最初は DP(Dynamic Programing: 動的計画法)マッチング)
  - 日本で育った手法 (独立にソ連から発表あり)
  - HMMの出現まで音声認識手法の主流(→ HMMはDPの一般化)
  - 日本電気中央研究所の迫江(現在、九州大学)および千葉。
- DPの原理: 最適化問題が部分問題に分解できるので、効率の高い最適経路探索問題が解ける
  - 距離尺度により、標準パターンと入力パターンの間を非線形対応づけ
  - 探索に DP(動的計画法)アルゴリズムを用いる。
  - 傾斜制限:「音声入力各フレームは、標準パターンのどれかのフレームに必ず対応付ける。同一フレームには2フレームまでは対応付けて良い。」
  - 窓制限:「マッチング経路は、対角線から一定幅以内になくてはならない。」

単純な動的計画法の原理

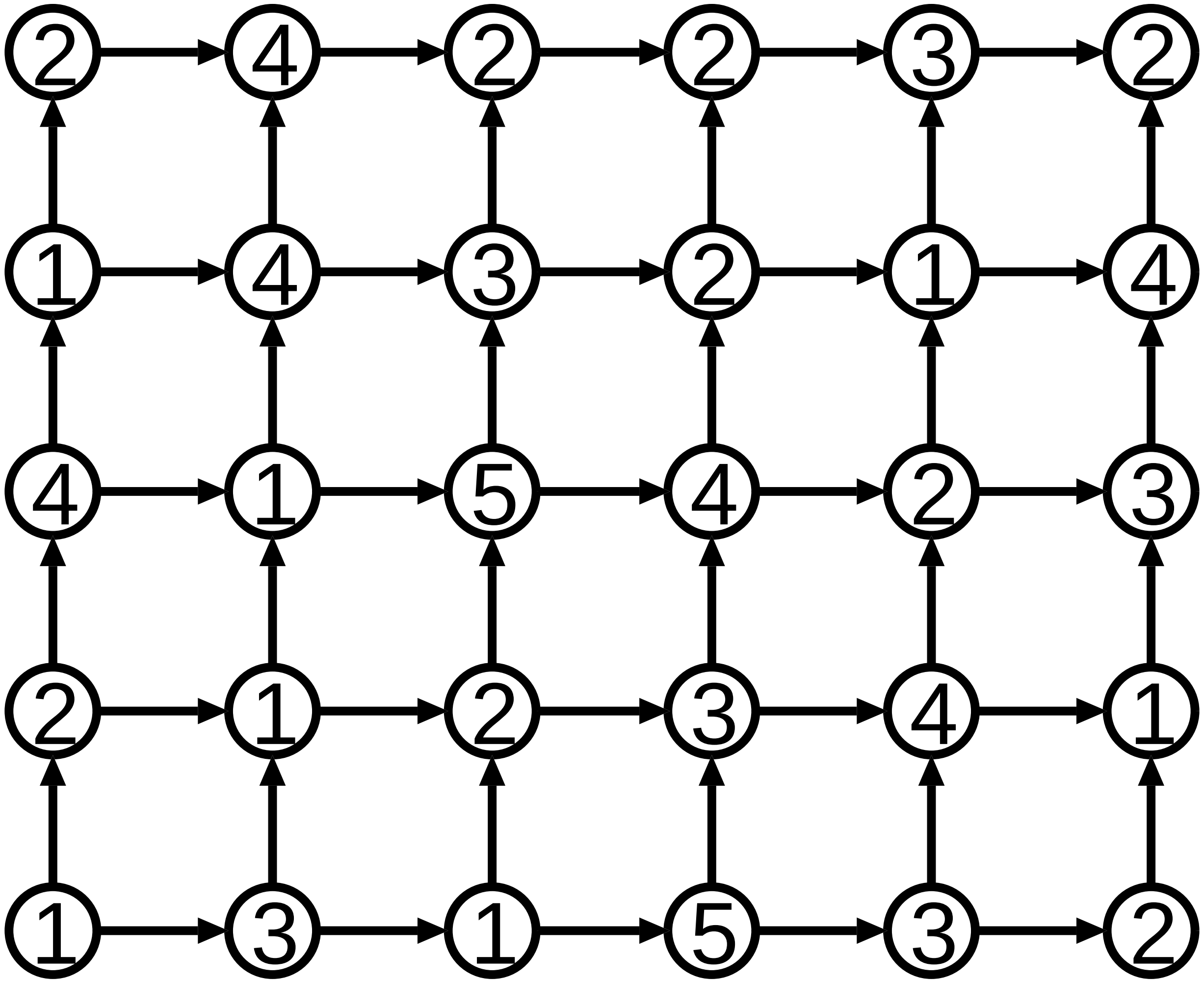


図5. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 1 列目

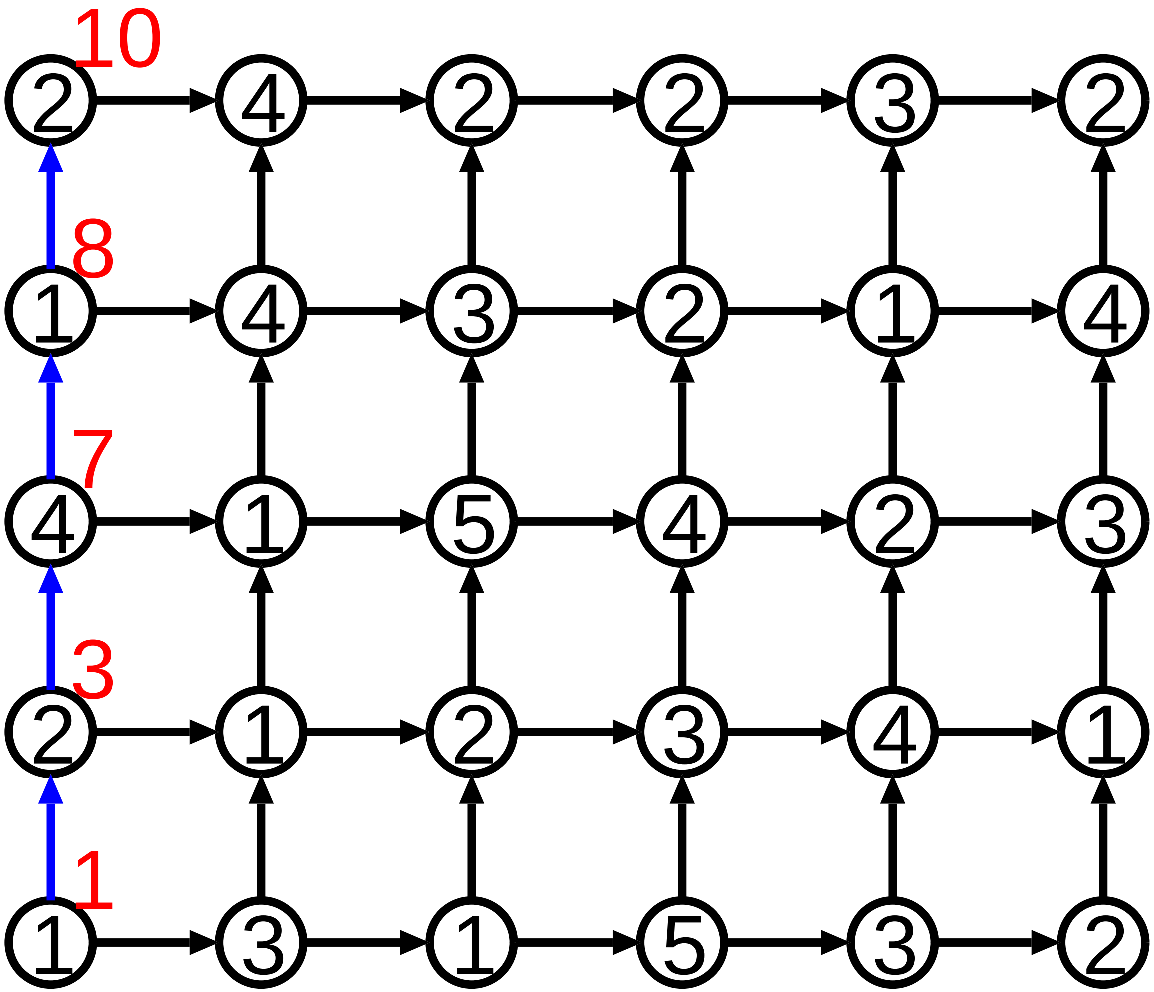


図6. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 2 列目

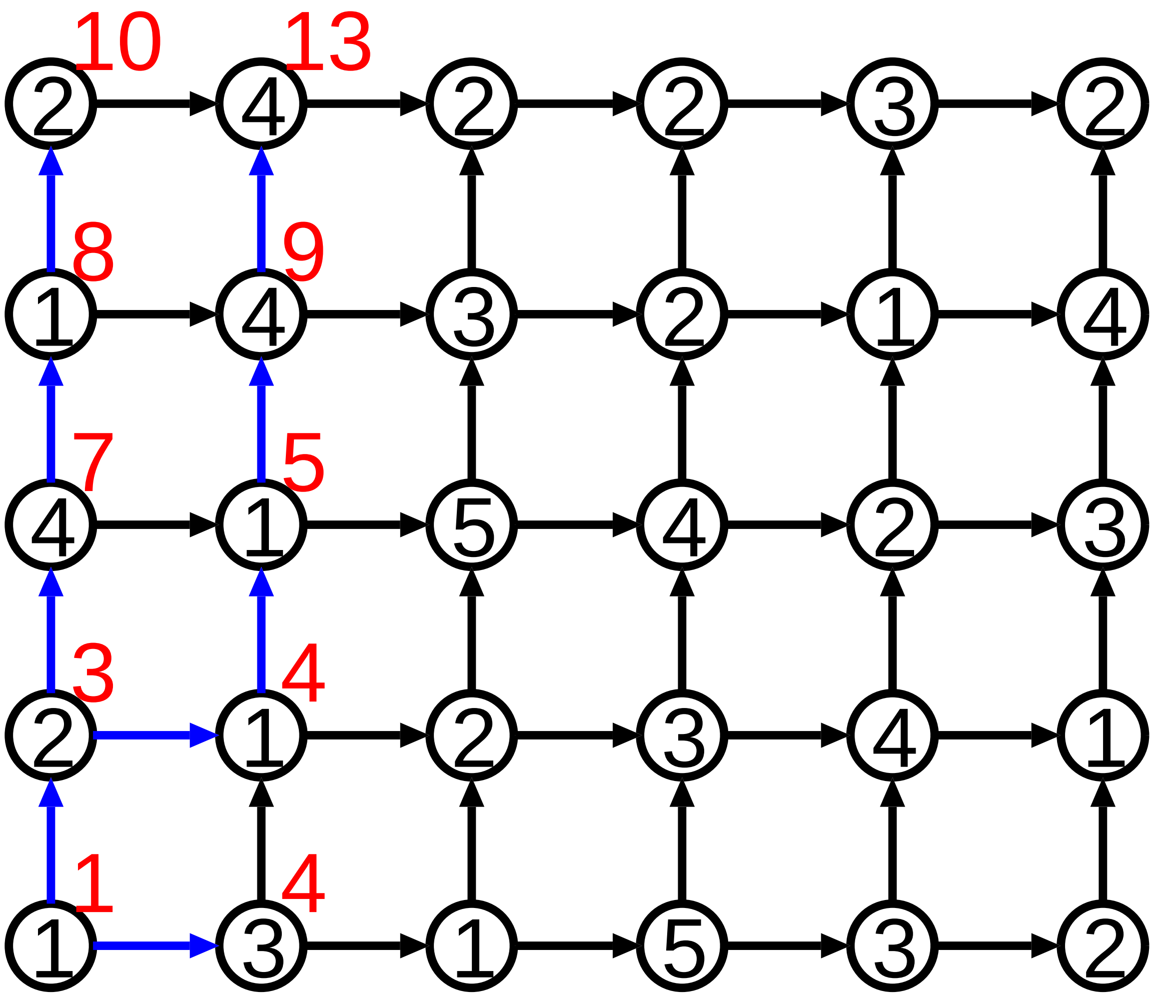


図 7. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 3 列目

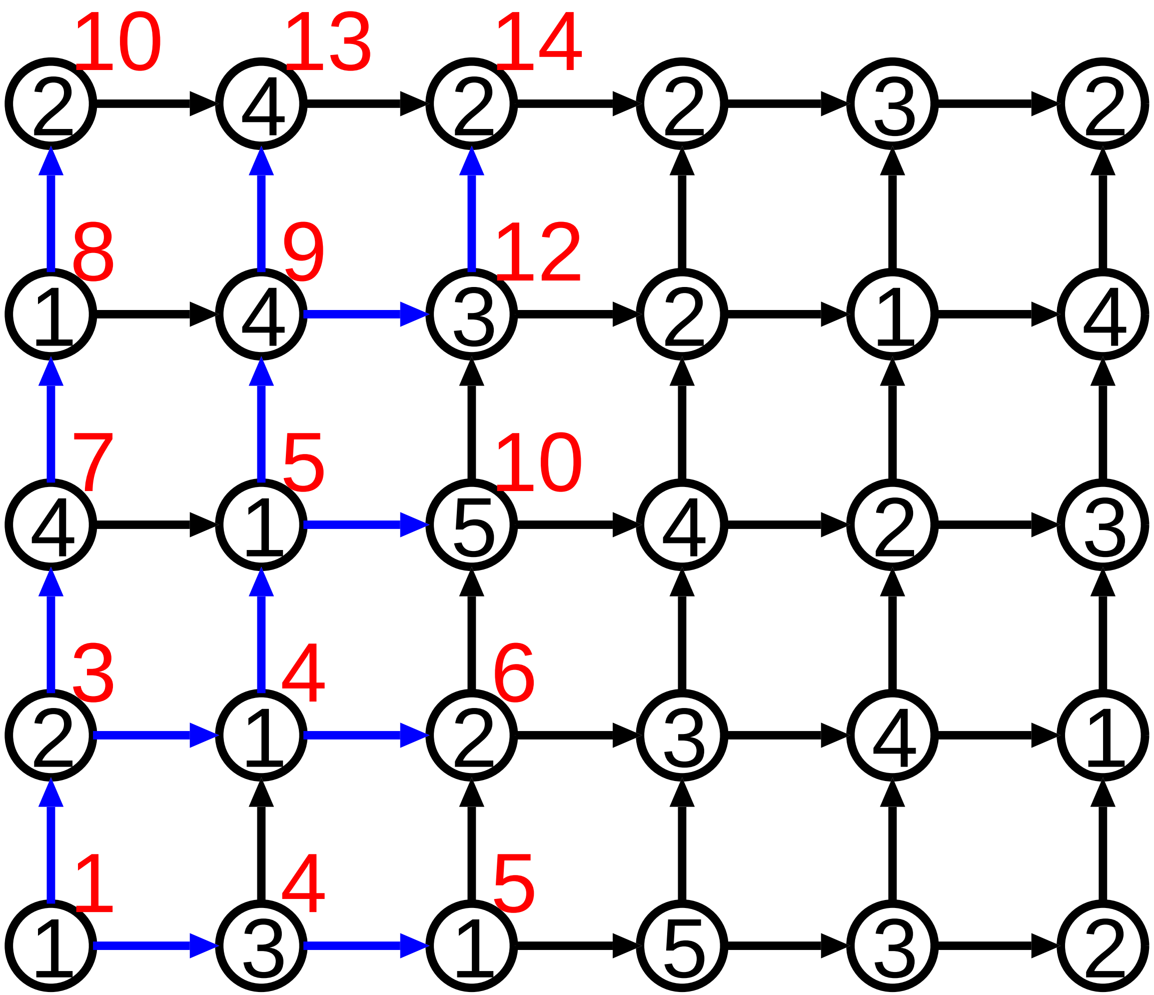


図8. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 4 列目

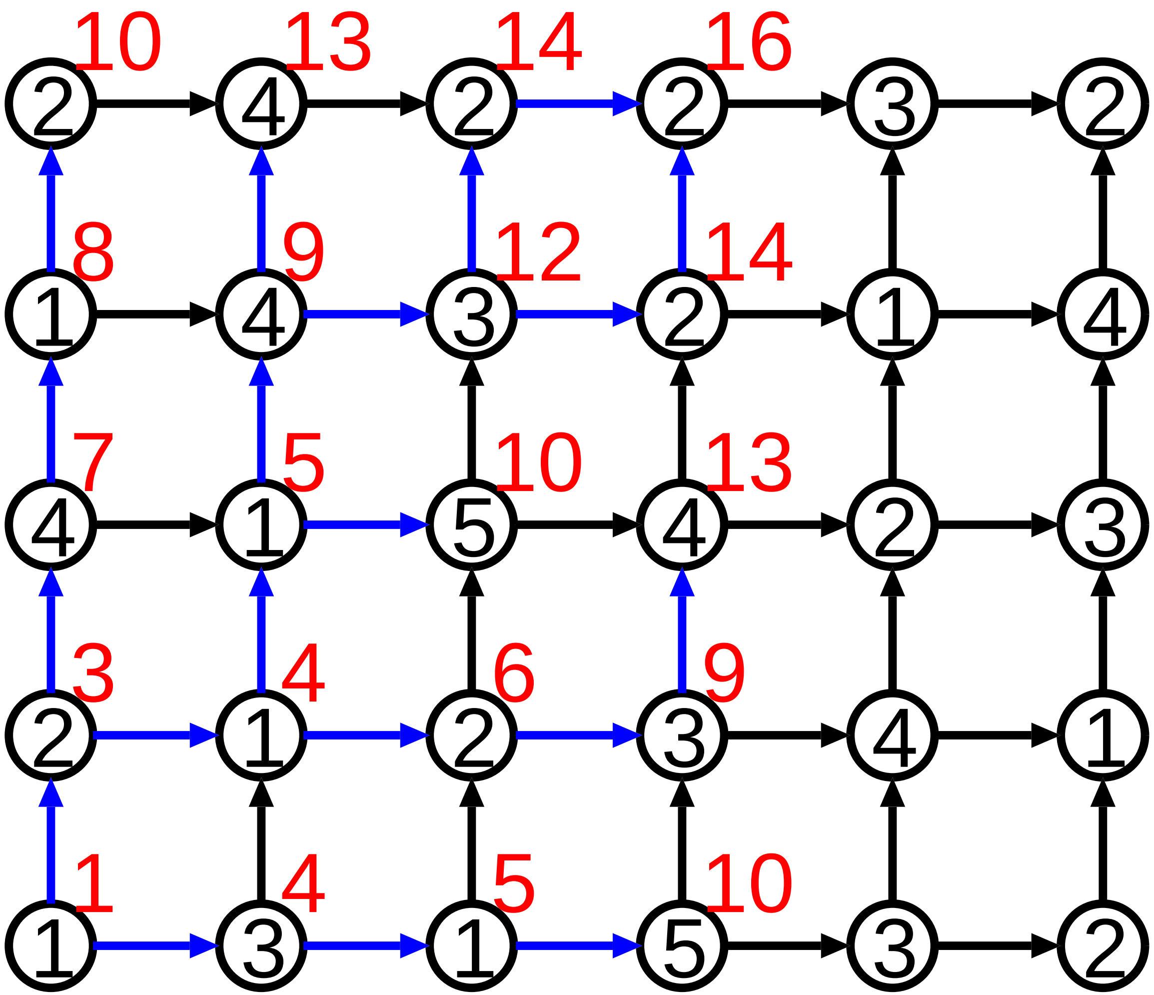


図9. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 5 列目

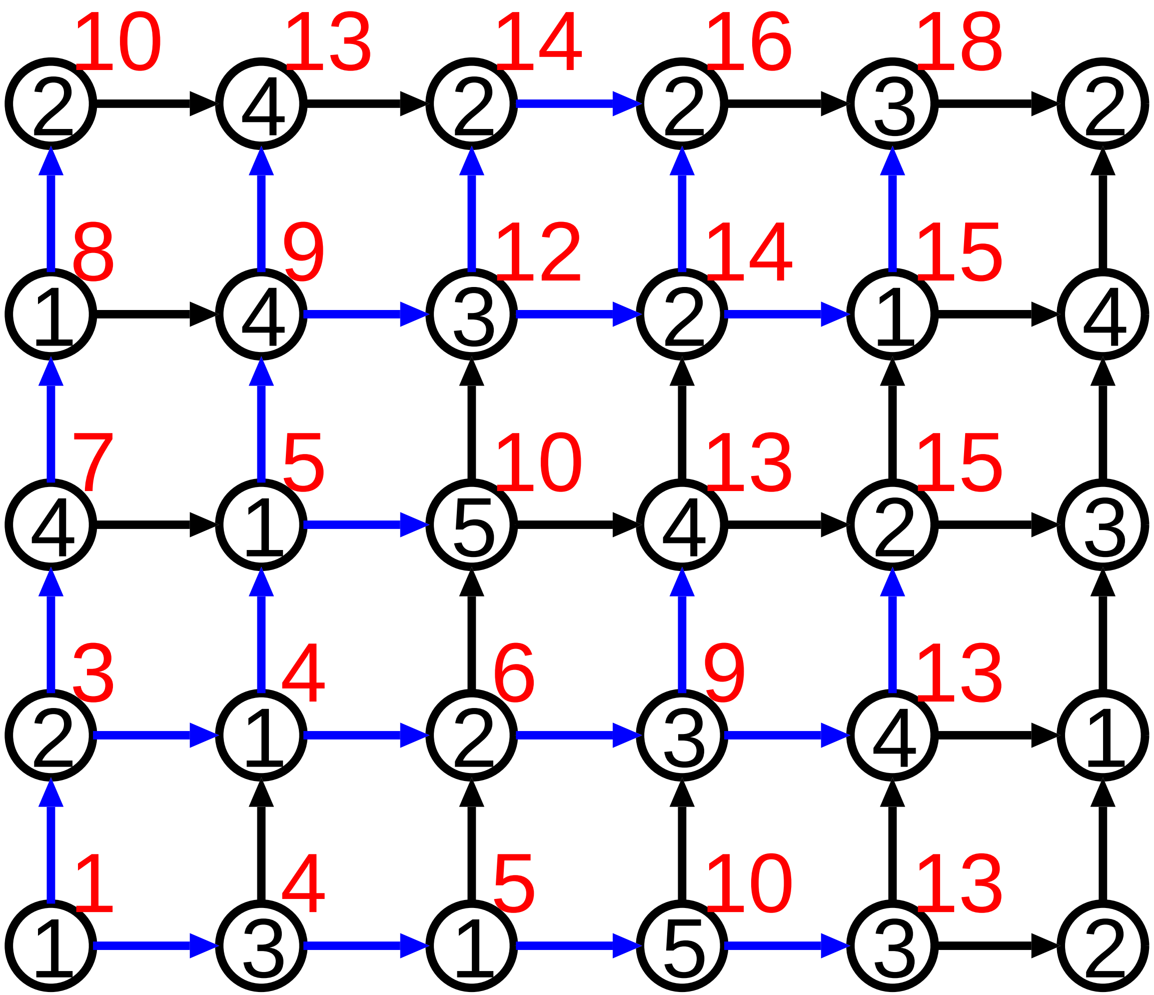


図10. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

単純な動的計画法の原理 – 6 列目

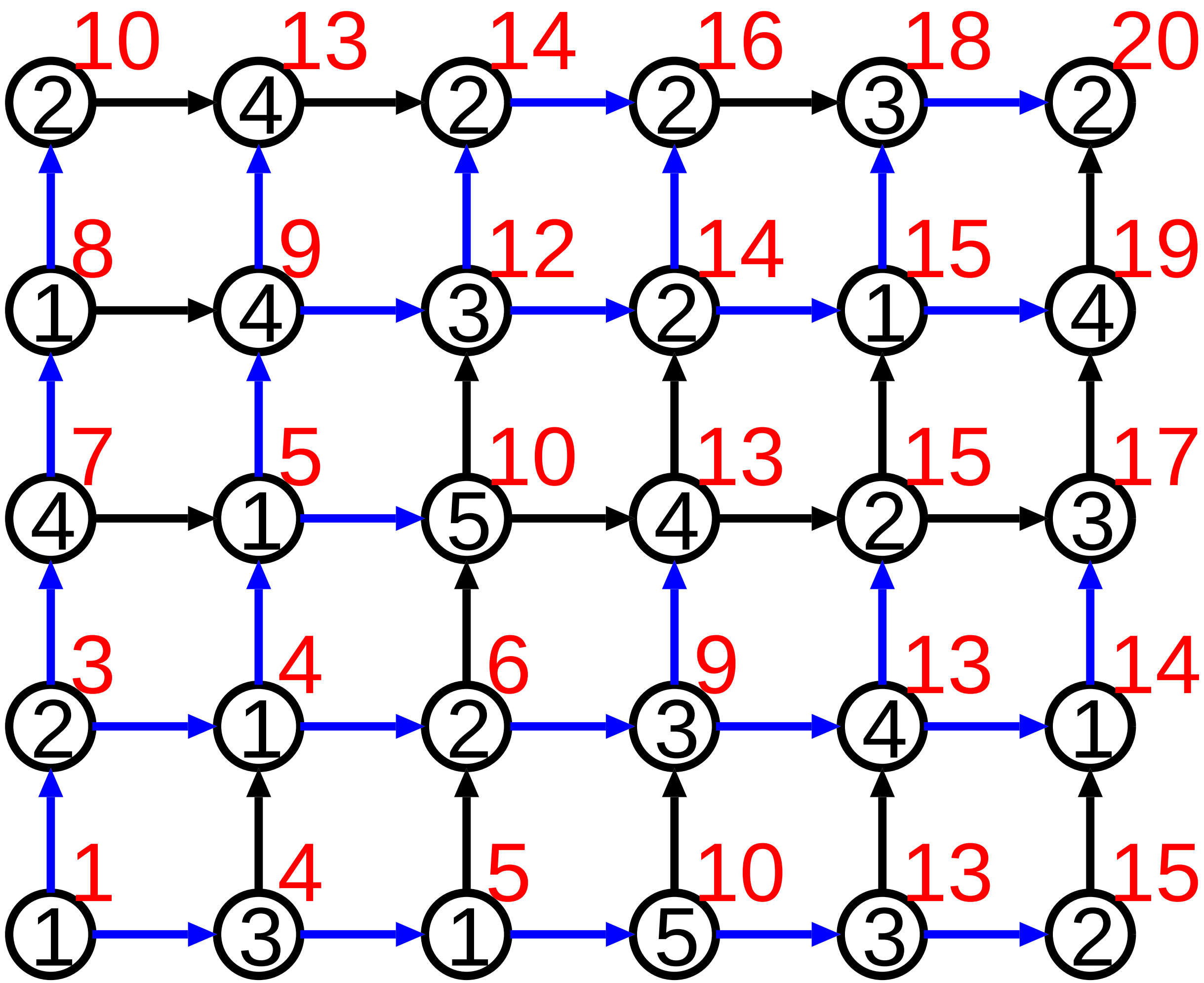


図 11. 動的計画法の例： 左下隅から右上隅に至る数値の総和が最小の経路を求めよ

最適経路決定： トレースバック (traceback)

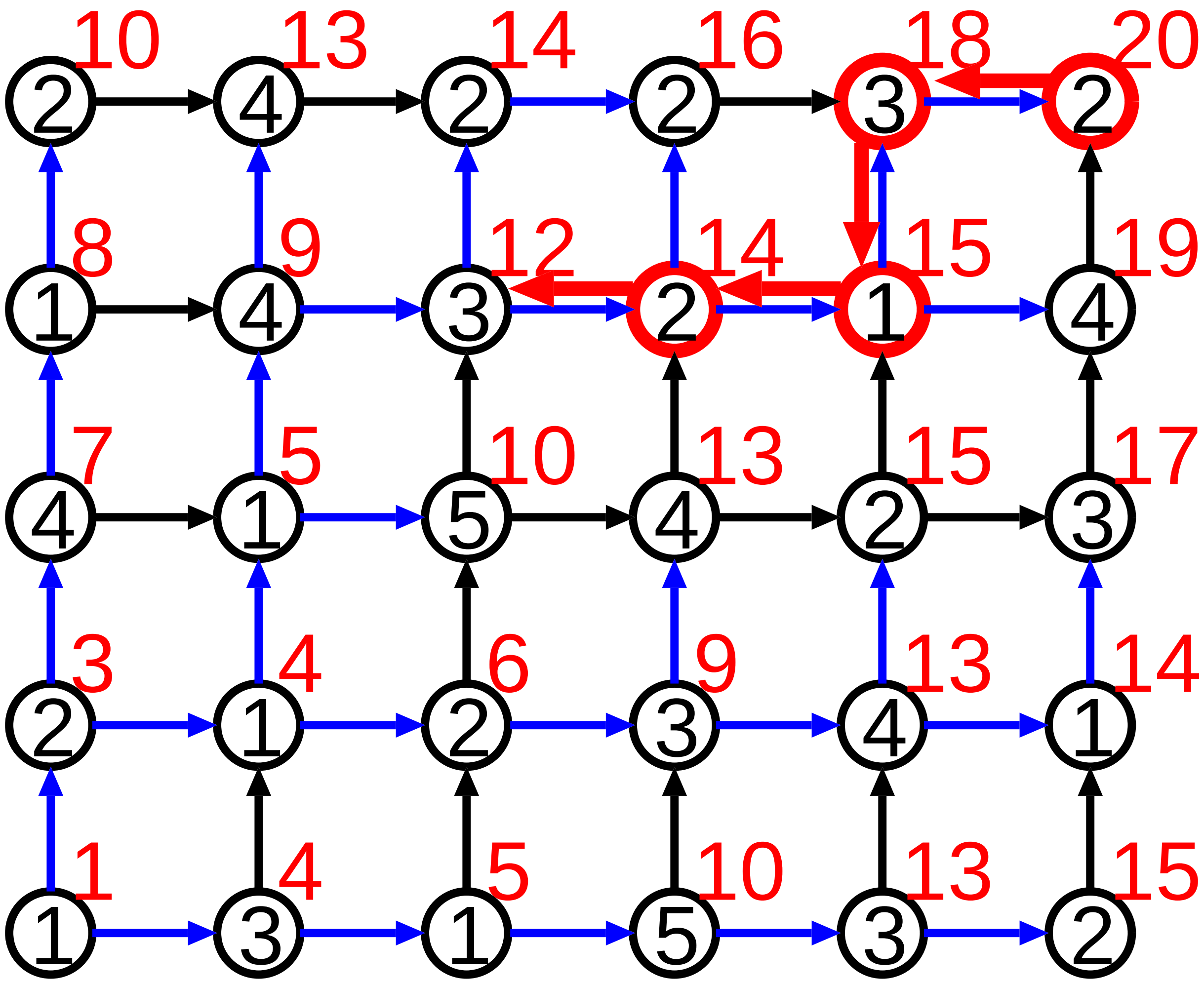


図12. 最後に右上隅から選択された経路(青矢印)を逆に辿って先頭に至ることにより、最適経路が決定される。それまでは最適経路は決まらないことに注意。

最適経路決定：トレースバック (traceback) 続

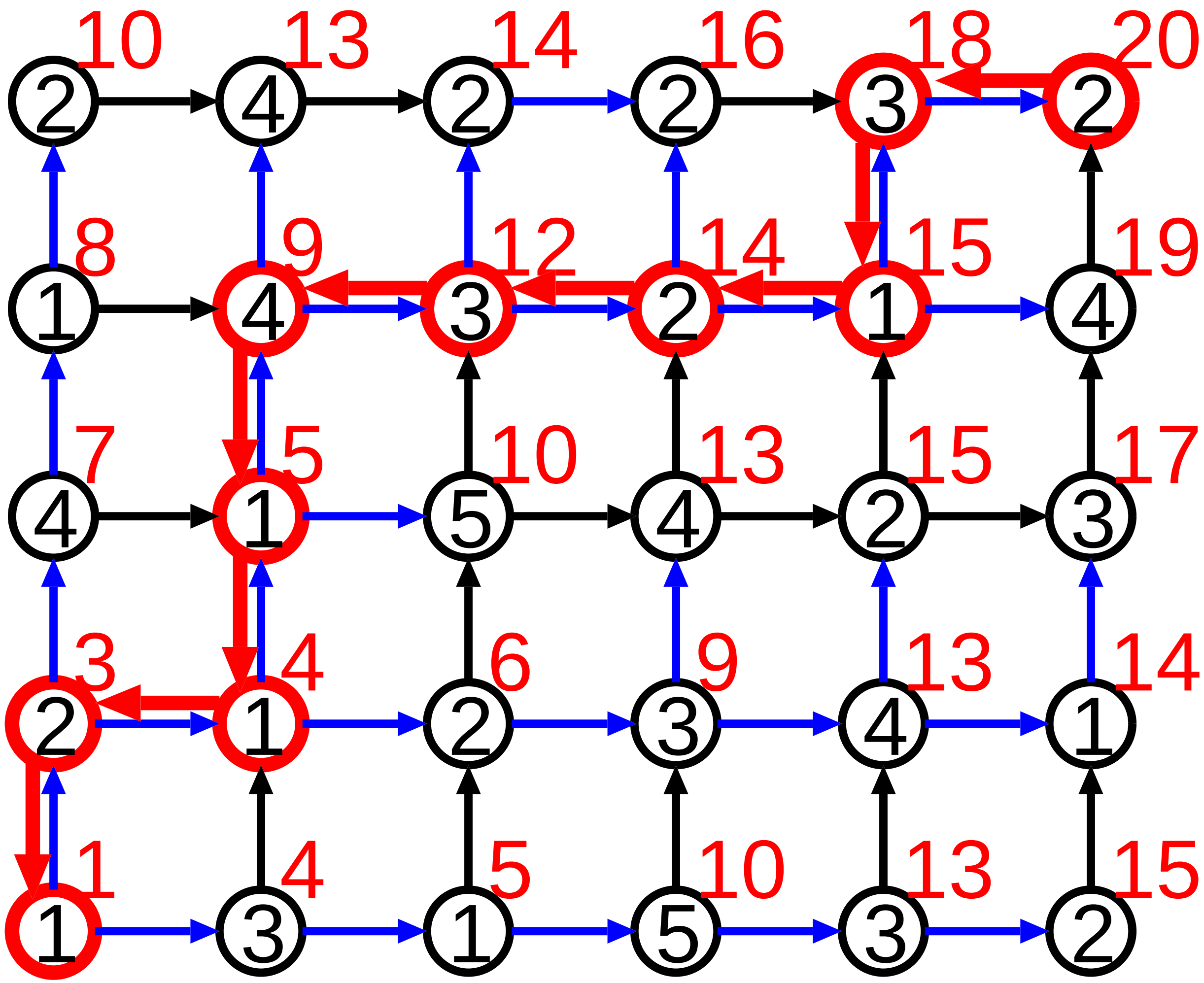


図13. 最適値が決まったあとで、最後に右上隅から選択された経路を辿って先頭に至ることにより、最適経路が決定される。それまでは最適経路は決まらないことに注意。

最適経路：トレースバック (traceback) の結果

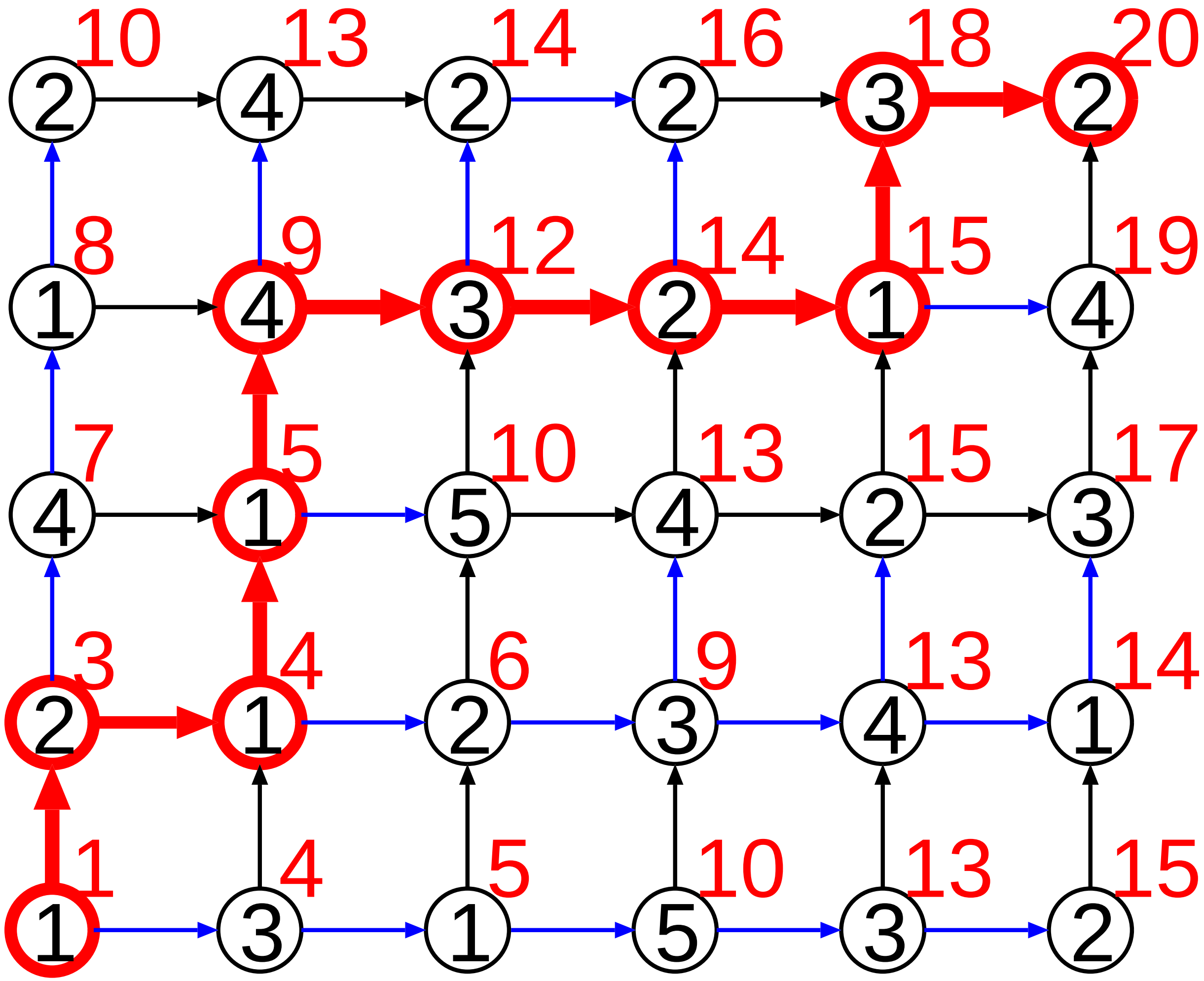


図14. トレースバックを行うためには、青色の経路を辿れるようにバックポインタを全て記憶(book keeping)する必要がある。→ メモリ消費

DP経路における探索木構造

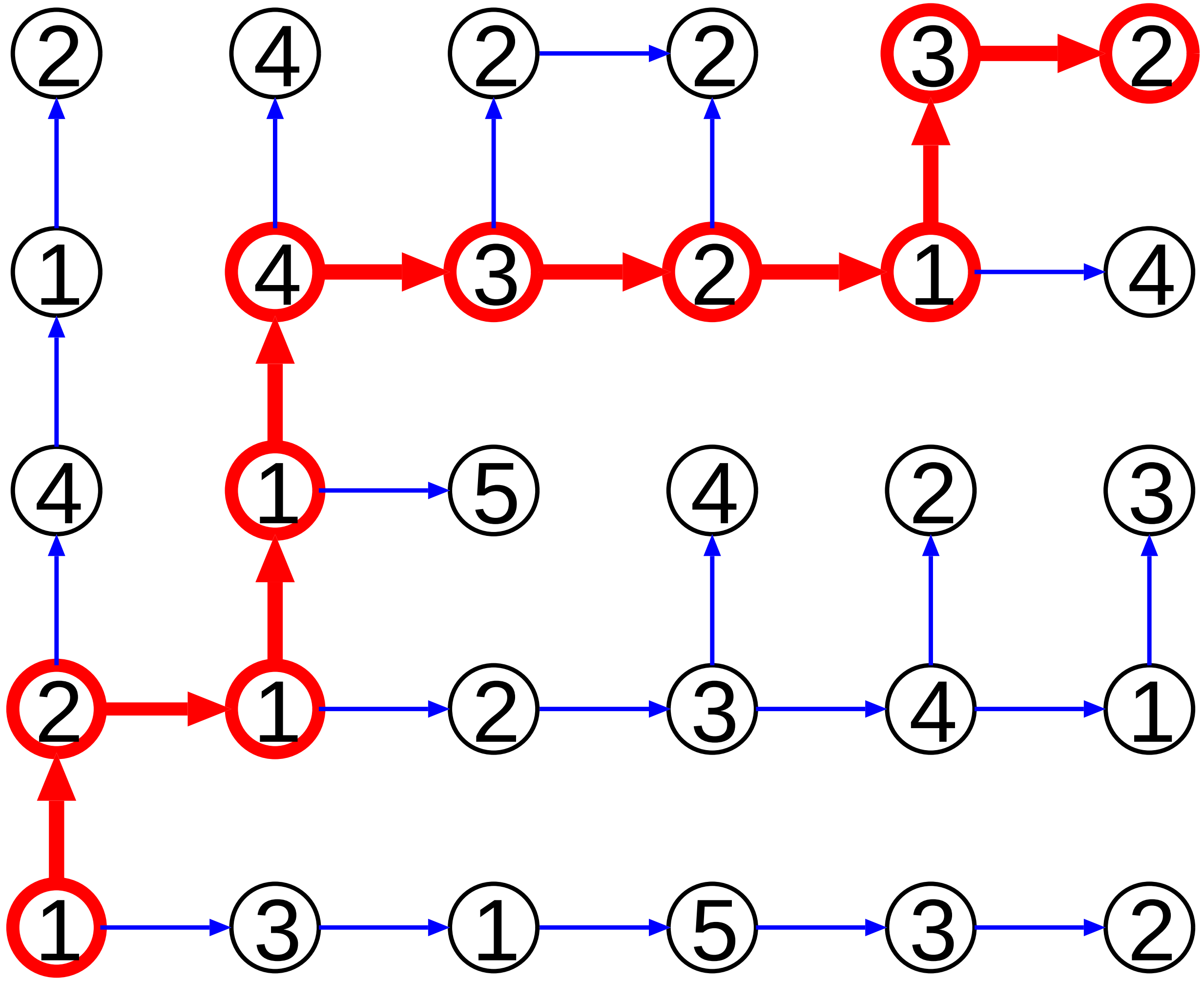


図15. 全ての経路は出発点を根とした木構造をなす。

N-best 問題：第 N 位まで経路を求める

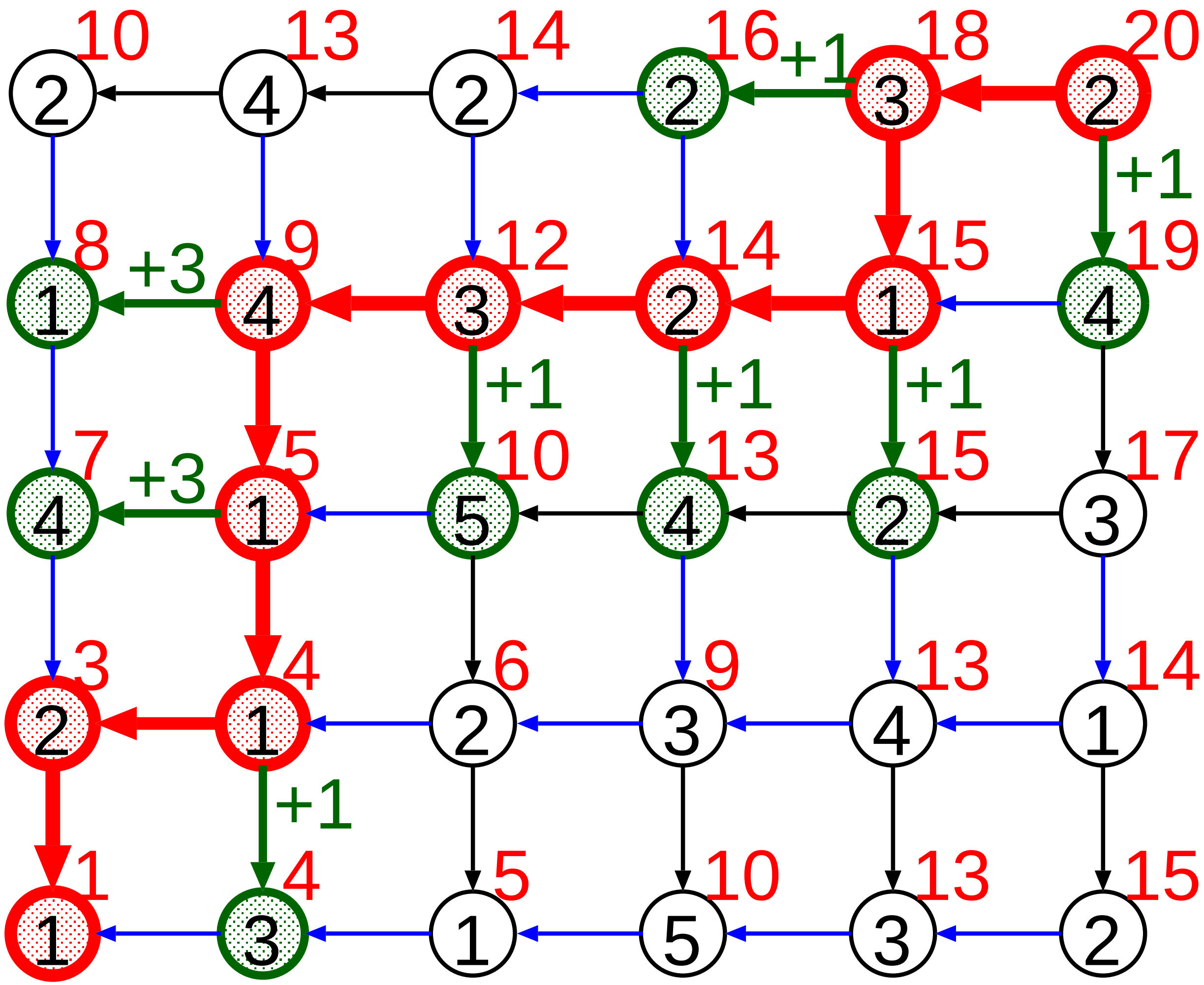


図16. 最適経路から外れる箇所はすべての節点。そのコストは簡単に計算できる。

N-best 問題：第 N 位まで経路を求める

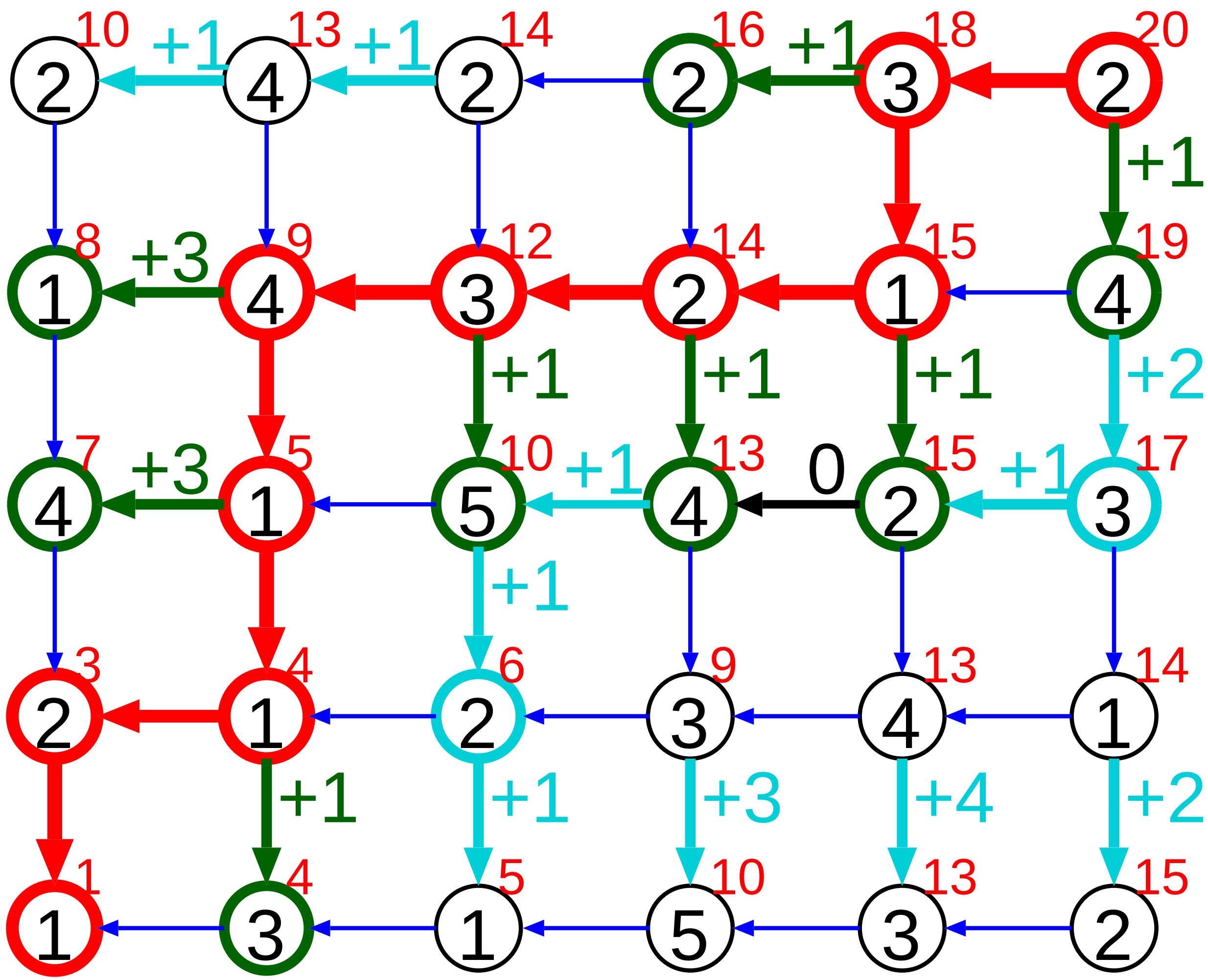


図17. 終点から遡る経路は追加コストを与える木構造をなす。これから任意の個数の上位最適経路が求められる。

計算順序 縦方向優先の場合

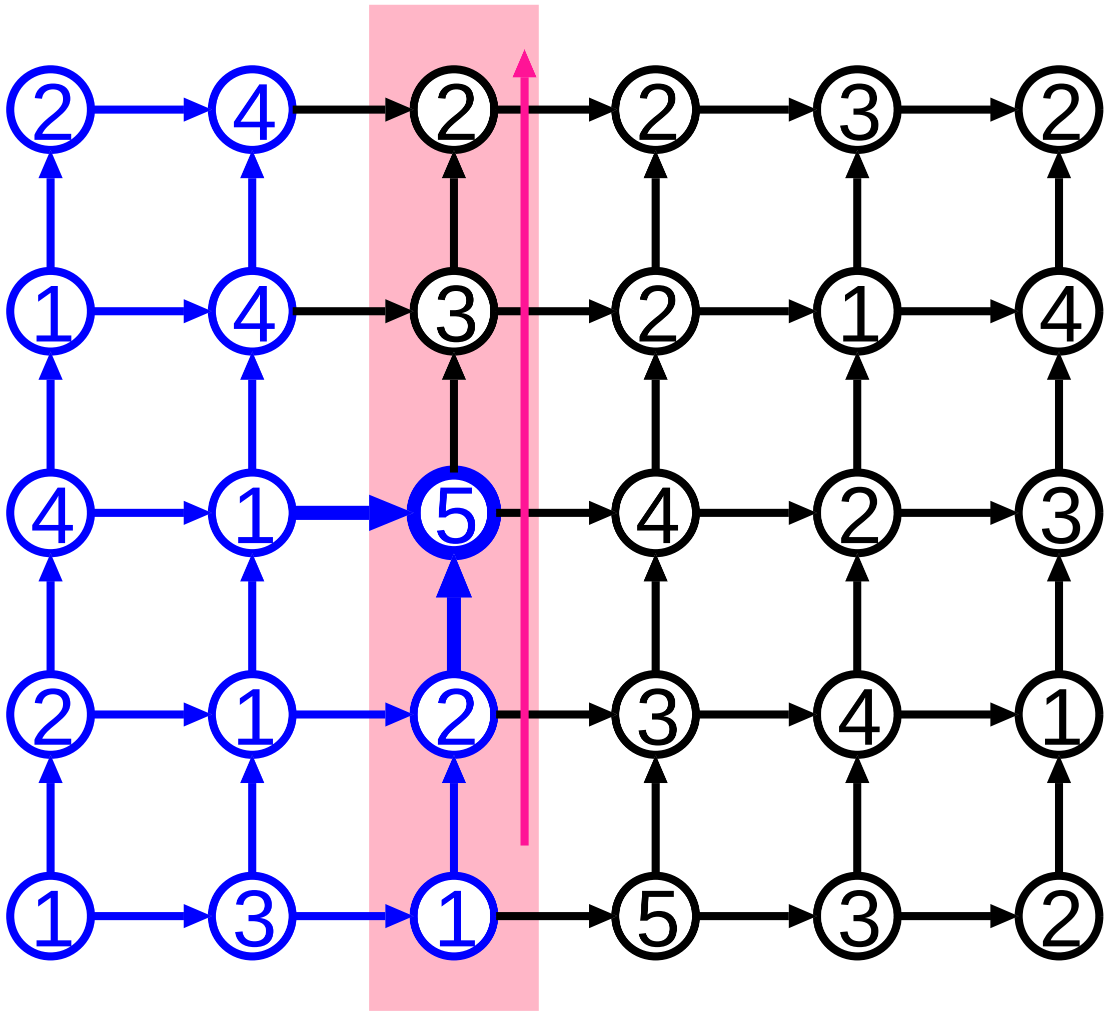


図18. 動的計画法の計算順序(1)： 下から上へ、左から右へ

計算順序 横方向優先の場合

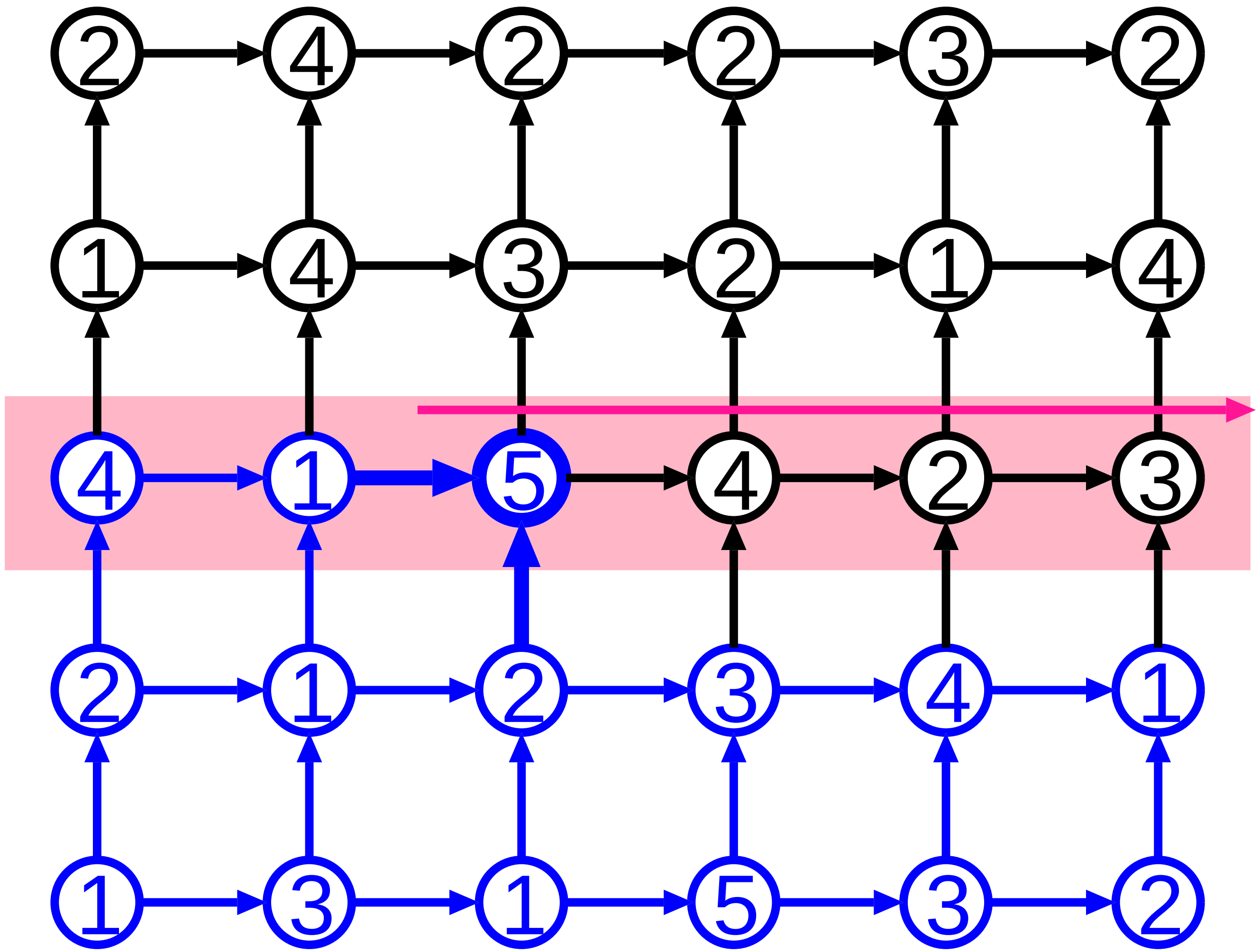


図 19. 動的計画法の計算順序 (2)： 下から上へ、左から右へ

計算順序 斜め方向の場合

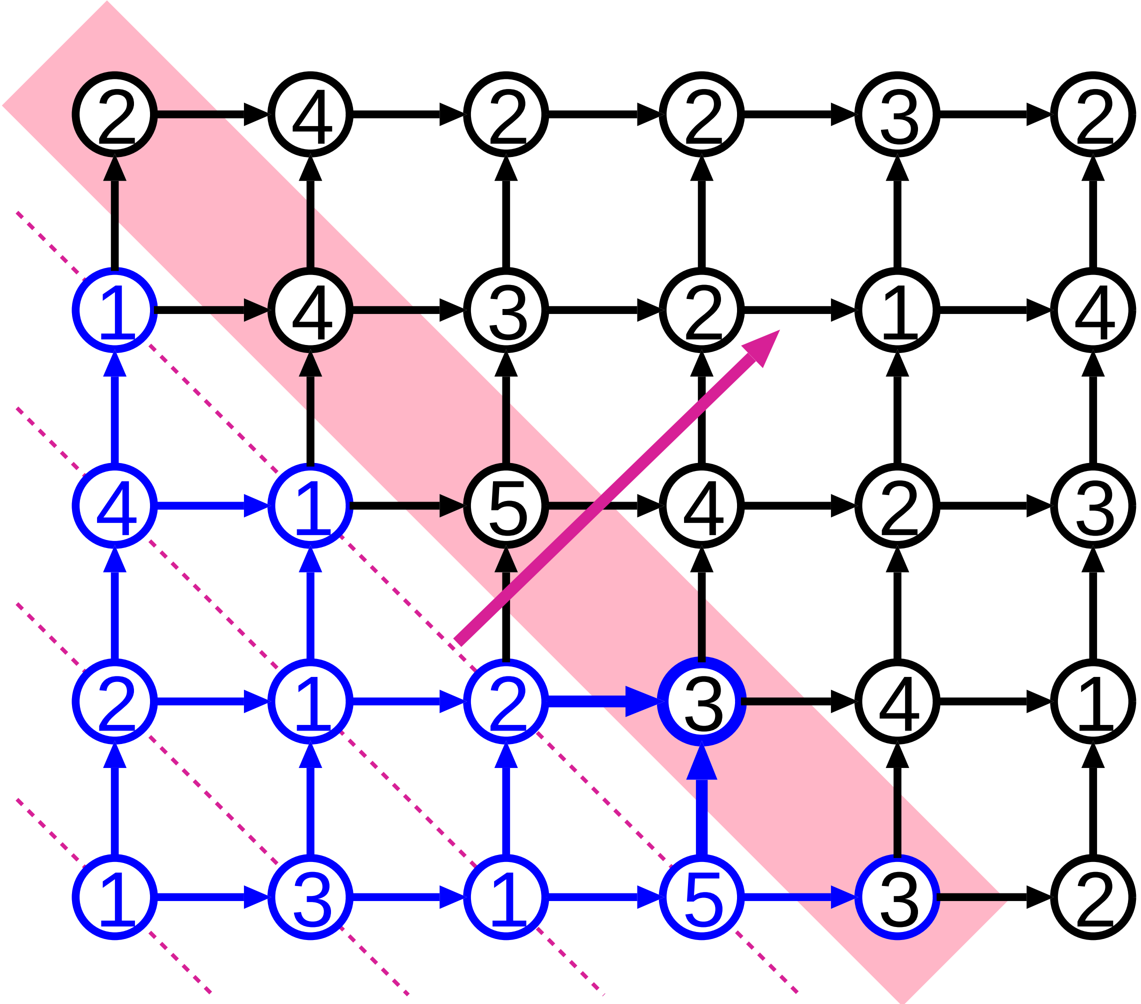


図20. 動的計画法の計算順序(3)： 左下から右上へ

経路にコストがある動的計画法の例

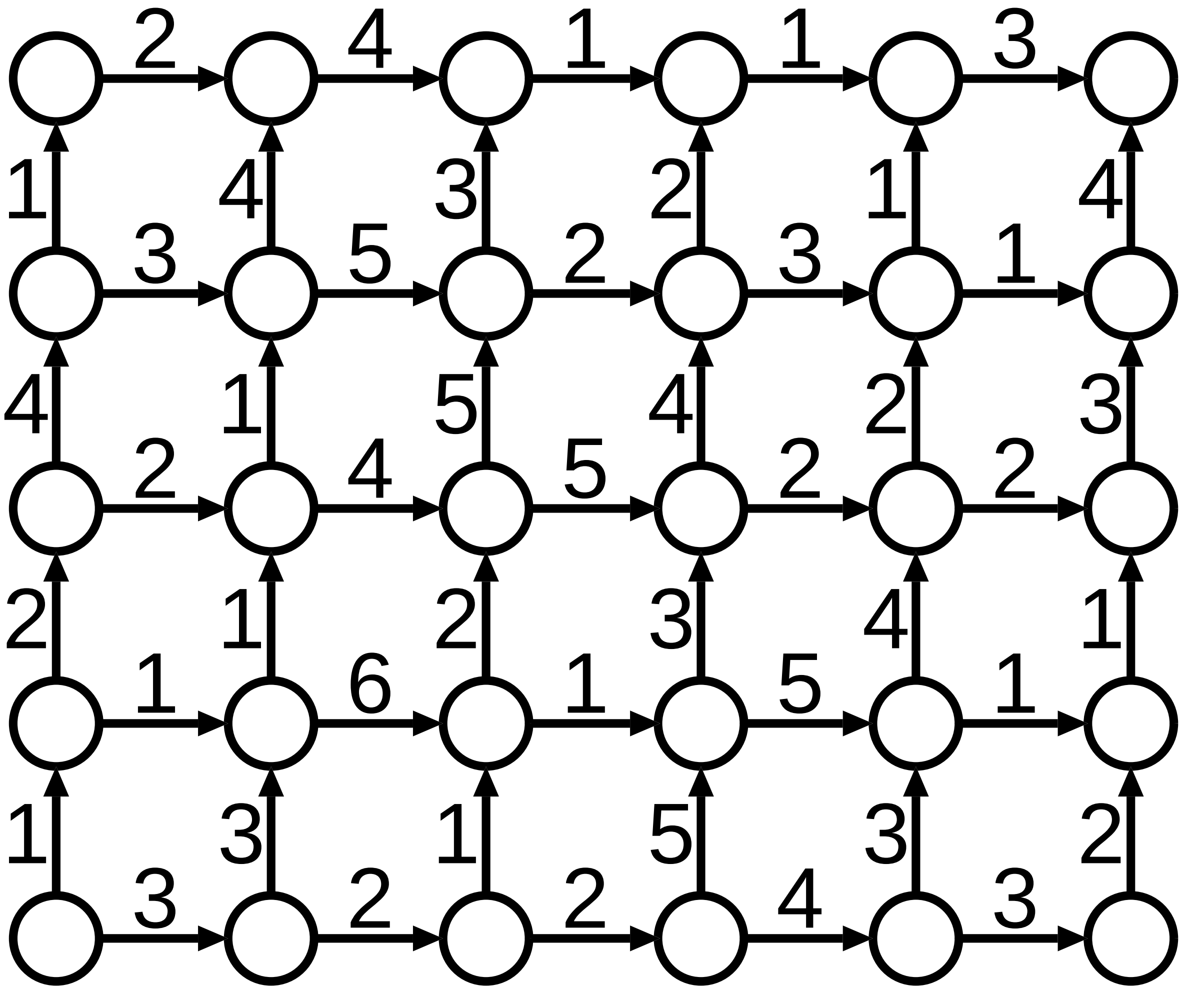


図21. 考え方は同じ：部分最適化問題へ分解できる

節点と枝の両方にコストがある動的計画法の例

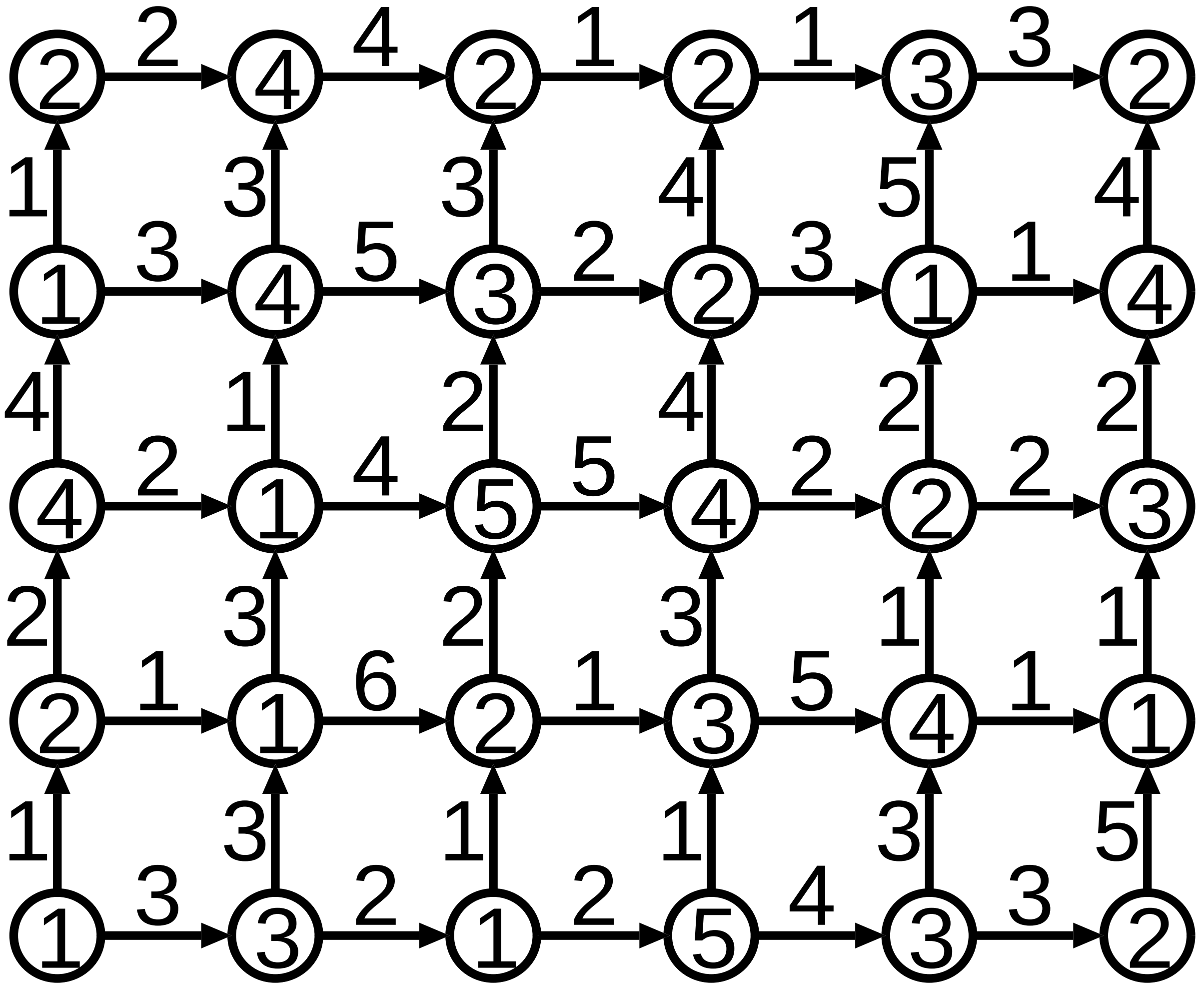


図22. 考え方は同じ：部分最適化問題へ分解できる

斜め経路を許す動的計画法の例

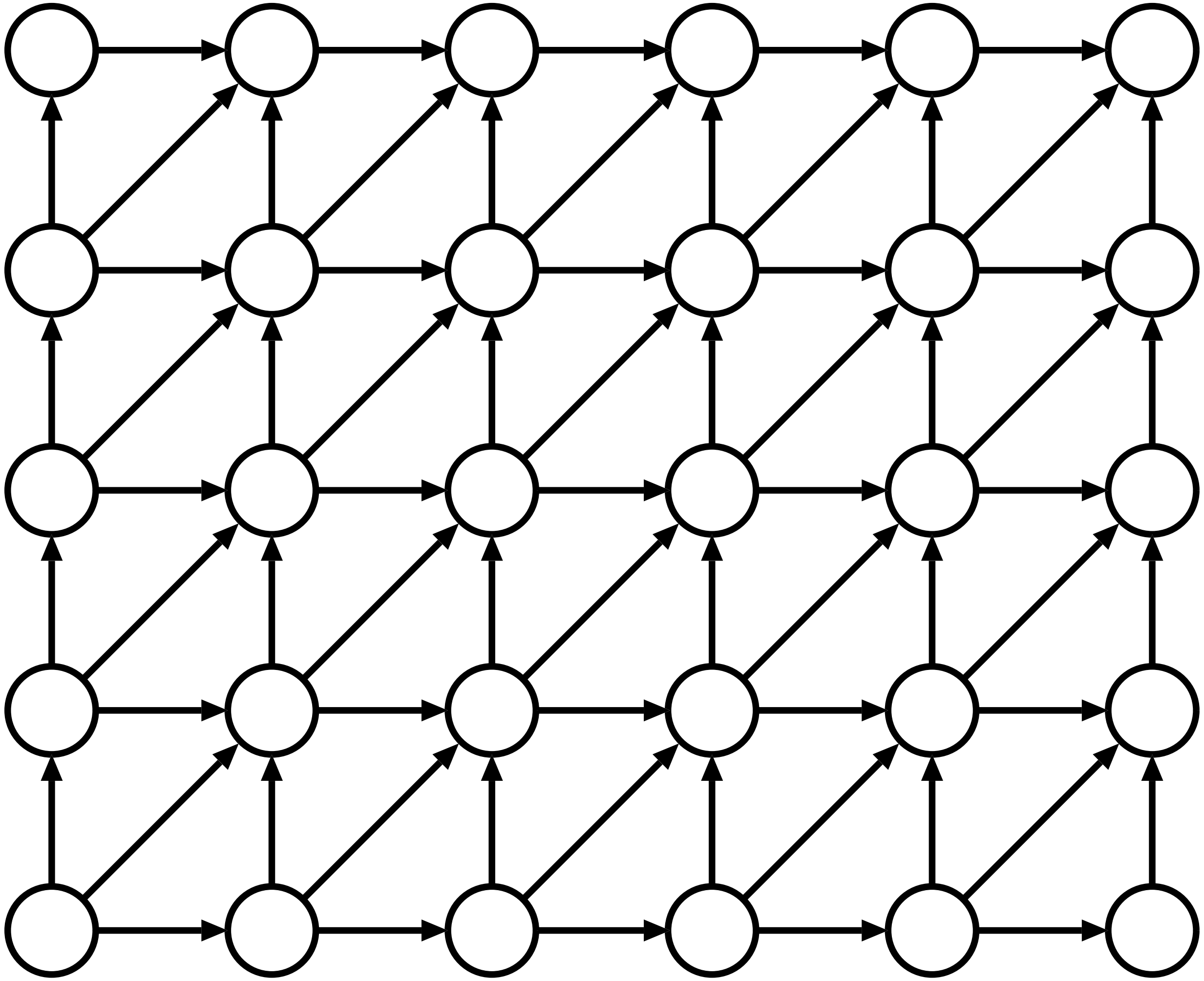


図23. 重み２の斜め経路を許す

DP経路制限の例

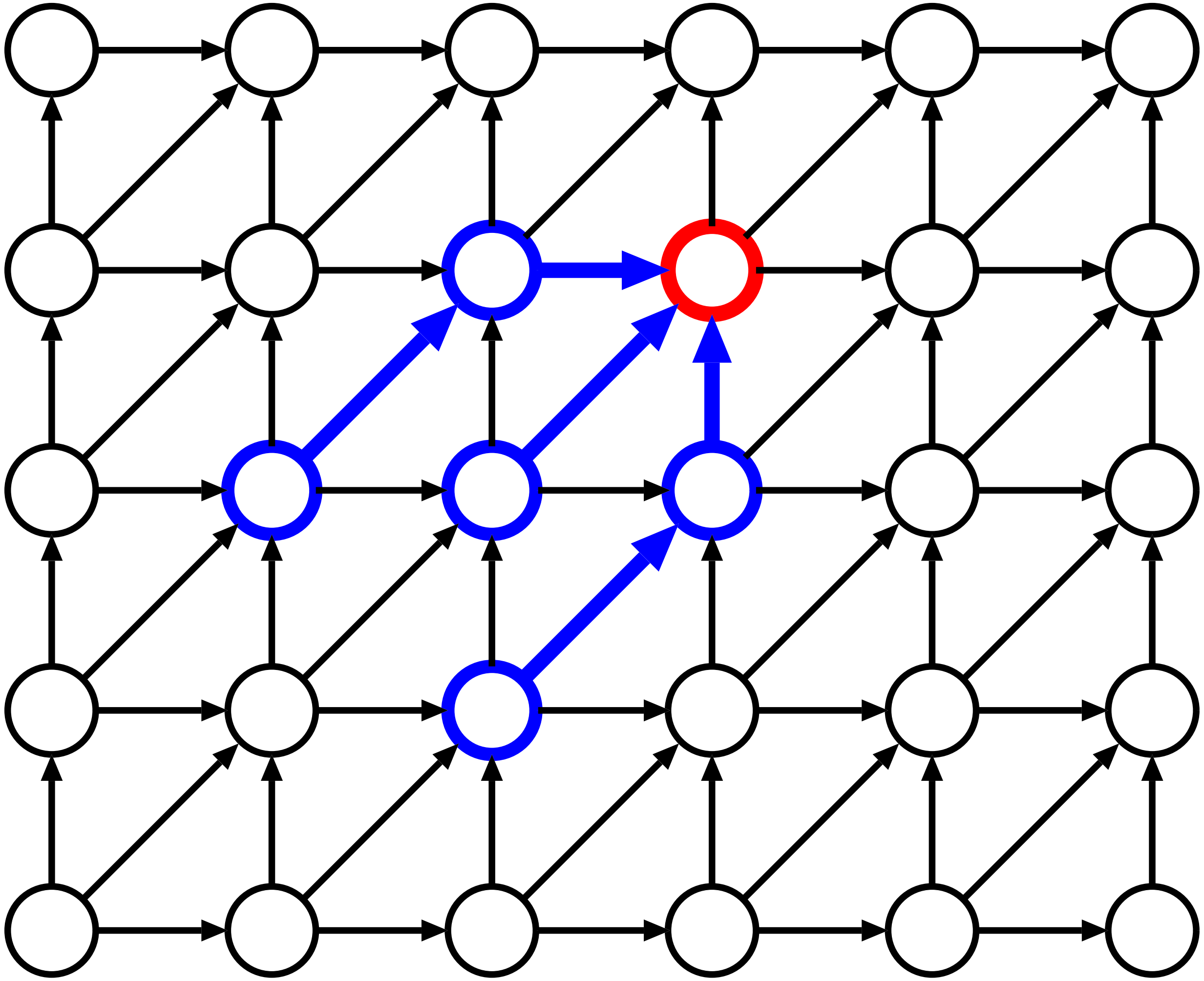


図24. 1/2 から 2 までの傾き制約を与えた経路制限

非対称経路の例

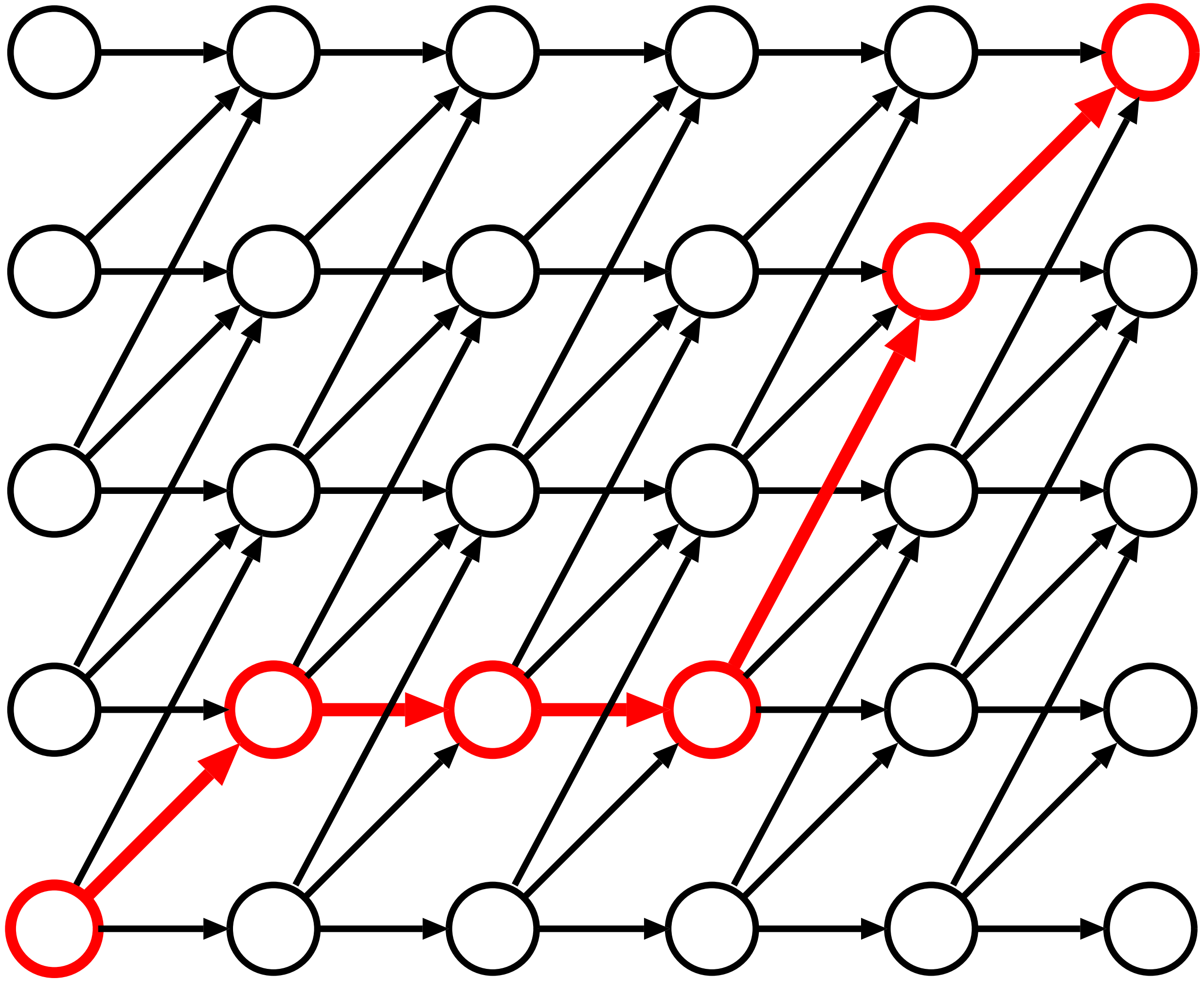
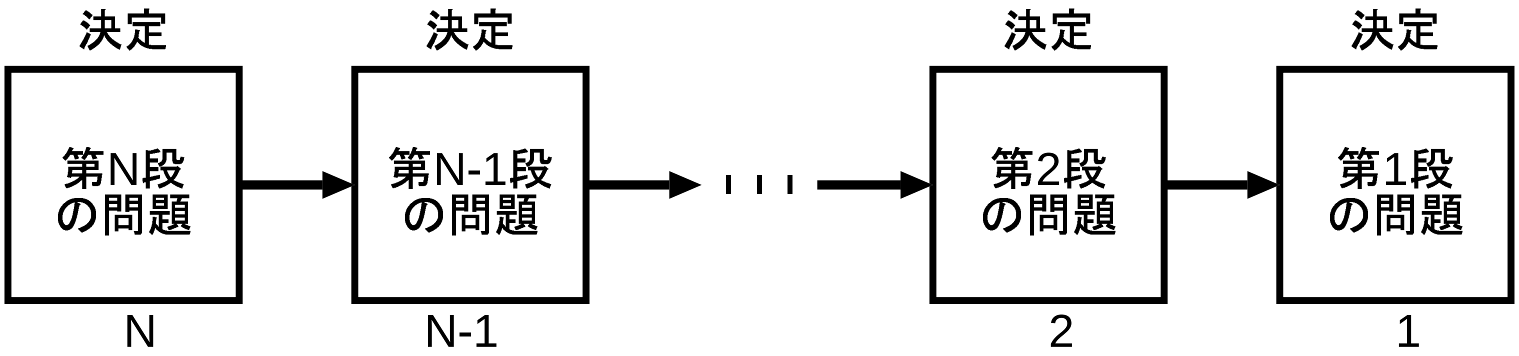


図25. 1/2 から 2 までの傾き制約を与えた経路制限 – 横軸方向の格子点数が和の個数

動的計画法とは

動的計画法の考え方

最適性原理に基づき，決定すべきN個の最適決定を順次行うことによって問題全体の最適解を模索



最適性原理(Principle of Optimality)

A→C(上): ある問題の最適経路
B: 最適経路の通過点
B→C(下): 部分最適経路とする

↓

矛盾

つまり，最適経路は部分最適経路を含まなくてはならない→最適性原理

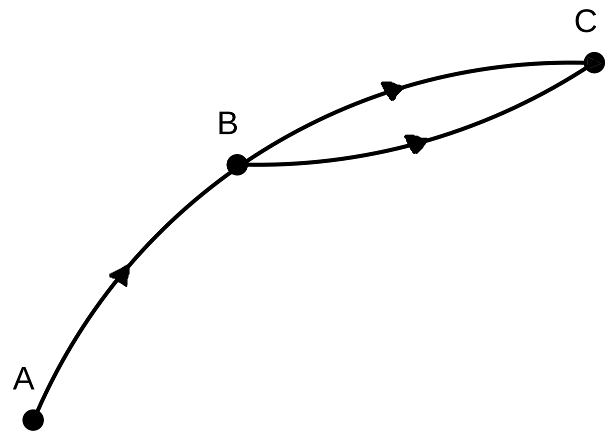


図26. 最適性原理

動的計画法の原理 – 解析的な例題

問題例

「$x+y+z=1$ という条件のもとで $x^2+y^2+z^2$ を最小にしなさい」

解法

1. 数式で表現し、以下のように書き換えると、

$$\min_{x+y+z=1}\{x^2+y^2+z^2\} = \min_{w+z=1}\{\min_{x+y=w}\{x^2+y^2\}+z^2\}$$

2. まず「$x+y=w$ という条件のもとで x^2+y^2 を最小にせよ」という問題を解く。これは簡単に解けて、$x=y=w/2$ の場合に最小値 $w^2/2$ になる。
3. 次に「$w+z=1$ という条件のもとで $w^2/2+z^2$ を最小にせよ」という問題を解く。これも簡単に解けて最小値は $w=2/3$ となり、結局 $x^2+y^2+z^2$ を最小にするのは $x=y=z=1/3$ の場合という結論が導かれる。
4. n 変数の場合は以下のように書き換えることができる。

$$f(n,1) = \min_{x_1+x_2+\ldots+x_n=1}\{{x_1}^2+{x_2}^2+\ldots+{x_n}^2\} = \min_{w+x_n=1}\{f(n-1,w)+{x_n}^2\}$$

動的計画法

- **動的計画法の特徴のまとめ**
  - 与えられた問題を最適性原理に基づいて関数再帰方式として解く．
  - N 段の決定過程において，$N-1$ 段に対する最適解だけを調べるだけでよい．
  - 全探索の場合と比較して，計算量が非常に少なくて済む．

- **動的計画法が用いられる問題**
  - 最大化・最小化問題
  - 経路選択問題
  - ナップザック問題
  - 資源配分問題

DPマッチング

DPマッチングの特徴

- 動的計画法を利用した、長さが異なるパターン間のマッチング法
- パターンを伸縮しながらマッチングを行う
- 全体としての距離が最小となるような対応づけをする

DP マッチング、変数定義

- **テンプレート系列：** $A(a_i) = a_1, a_2, a_3, \ldots, a_I$
- **入力系列：** $B(b_j) = b_1, b_2, b_3, \ldots, b_J (I \neq J)$
- **対応変換関数：** $F = c(1), \cdots, c(k), \cdots, c(K)$
  $c(k) = (i(k), j(k))$**,** i, j **平面でとる経路**
- **特徴ベクトル間距離：** $d(c(k)) = d(a_{i(k)}, b_{j(k)})$
  F **上各点での特徴ベクトル間の距離**
- F **上の各点の間の距離 ：** $w(k) = c(k)$ **と** $c(k+1)$ **の間の距離**
- A, B **の間の距離：** $D(A, B)$

DP マッチング、系列間の距離

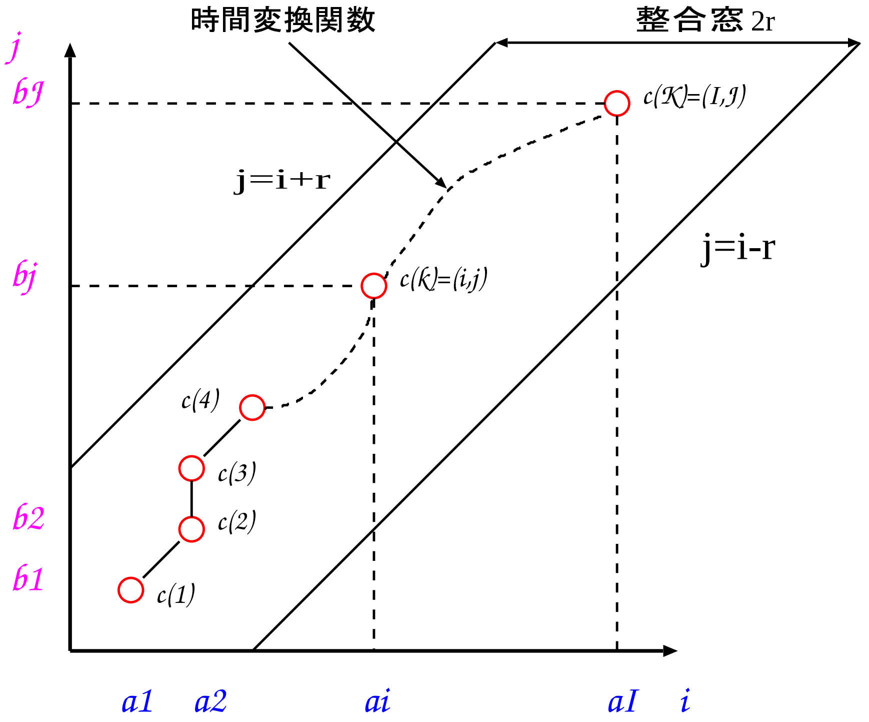


図27. 二系列の対応づけ

経路Fにおける距離の総和は

$$H(F) = \frac{\sum_{k=1}^{K} d(c_k) \cdot \omega_k}{\sum_{k=1}^{K} \omega_k}$$

二つのパターンの最小距離は

$$D(A, B) = \min_F \frac{\sum_{k=1}^{K} d(c_k) \cdot \omega_k}{\sum_{k=1}^{K} \omega_k}$$

ただし、経路の取り方には制限がかかる。

経路制限

- **単調性：** $i(k-1) \leq i(k), j(k-1) \leq j(k)$
  後戻りしない
- **連続性：** $i(k) - i(k-1) \leq 1, j(k) - j(k-1) \leq 1$
  格子点を飛び越さない
- **境界条件：** $i(1) = 1, j(1) = 1, i(K) = I, j(K) = J$ **入力となる2つの系列の長さ以上に探索空間を広げない**
- **整合窓の条件：** $|i(k) - j(k)| \leq r$ **（極端な伸縮を防ぐため、** r **：定数）**

荷重係数

二つのパターンの最小距離

$$D(A,B) = \min_F \frac{\sum_{k=1}^{K} d(c_k) \cdot \omega_k}{\sum_{k=1}^{K} \omega_k}$$

ここで荷重係数の設定によって $D(A,B)$ **の分母を定数とすることでこの最小化問題は簡単化できる。**

- **対称形**

$$\omega_k = i_k - i_{k-1} + j_k - j_{k-1}$$

このとき分母は $I + J$ **となる**

- **非対称形**

$$\omega_k = i_k - i_{k-1} = 1$$
$$0 \leq j_k - j_{k-1} \leq 2$$

このとき分母は I**、分子は** $\Sigma_{i=1}^{I} d(i, j_i)$ **となる**

動的計画法の応用

■ 対称型

$$\omega_k = i_k - i_{k-1} + j_k - j_{k-1}$$

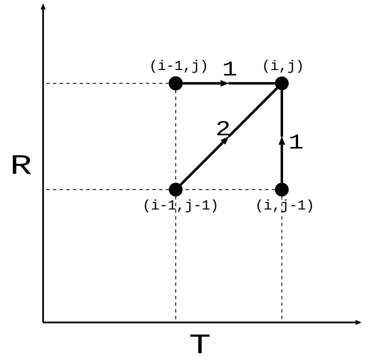


図28. 対称型

■ 非対称型

$$\omega_k = i_k - i_{k-1} = 1$$
$$0 \leq j_k - j_{k-1} \leq 2$$

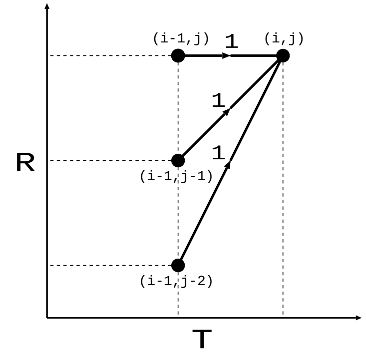


図29. 非対称型

部分和 $g(i,j)$ は，

$$g(c_k) = g(i,j) = \min_{c_1,\cdots,c_k} \sum_{t=1}^{k} d(c_t) \cdot \omega_t$$

ここで，動的計画法が使える．

$$g(c_k) = g(i,j) = \min_{c_{k-1}} [g(c_{k-1}) + d(c_k) \cdot \omega_k]$$

動的計画法の応用

ω_k として，非対称型のものを用いたとすると，

$$g(i,j) = \min \begin{Bmatrix} g(i-1,j) \\ g(i-1,j-1) \\ g(i-1,j-2) \end{Bmatrix} + d(i,j)$$

と書ける．

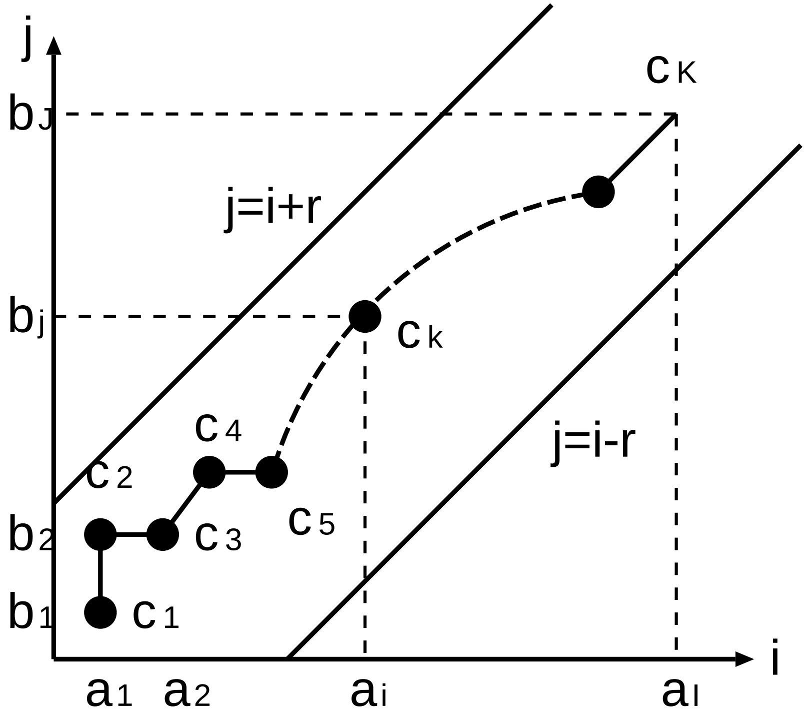


図30. A，B 両系列の対応づけ

単純なＤＰマッチングの例

Sample Pattern								
7					18	9	9	9
6				10	13	7	9	10
5			11	8	9	6	9	
1		9	10	7	6	10		
3	5	5	5	4	5			
6	5	3	2	4				
5	2	2	3					
Input Pattern	3	5	6	4	2	5	8	7

図31. 絶対値差の累積値が最小の経路を求める例：横軸 i, 縦軸 j

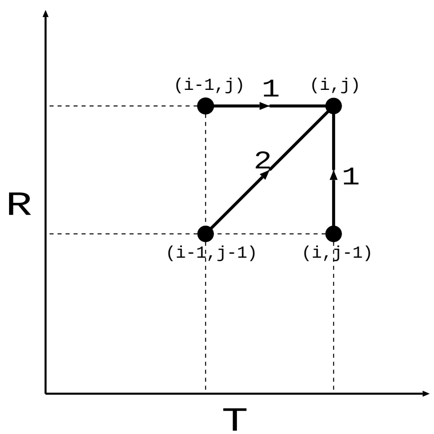


図32. 用いた荷重係数

整合窓は $|i(k) - j(k)| \leq 3$ とした

DPマッチングのまとめ

- 要件
  - 入力：2つの特徴ベクトル系列
  - 出力：2つの特徴ベクトル系列の間の距離と対応づけ
  - 特徴ベクトル間の距離が定義されていること。
  - 格子点間の移動コスト：探索空間における格子点間の移動コストが定義されていること。
  - 包含性：探索空間において、全体問題の最小解を与える経路が部分問題の最小解を与える経路を含んでいること。
- 要点
  - 再帰的に部分最適化問題へ帰着できる
  - 前向き経路探索で最小累積距離が得られる
  - 後向き経路追跡(トレースバック)により最適経路決定
  - 経路制限により傾きなどを制御
  - 整合窓により整合範囲を制御 (極端な整合を避ける)
  - 経路の木構造からN-best解が得られる