

# 食育のためのおやつ選択ゲームにおける 選択要素のモンテカルロ法による推薦

但馬 康宏<sup>1,a)</sup> 諏訪 貴洋<sup>1</sup> 藤岡 輝兆<sup>1</sup> 伊藤 祥平<sup>1</sup> 久保田 恵<sup>2</sup> 菊井 玄一郎<sup>1</sup>

**概要：**食生活から健康を維持、向上させる取り組みとして食育が注目されている。本研究では、小学生程度の子供を対象に食育を助ける教材として、おやつを選択ゲームを作成した。このゲームは、スーパーでのおよつちの買物物をモデルとして、教師と子供が対話しながら正しいおよつちの選択を理解する授業の中で使われる。ゲーム中では試行錯誤を繰り返しておよつちの選択を行うが、より低カロリーなおよつちにうまく導く仕組みがあると指導がよりやりやすくなる。そこで本研究では、ゲーム木探索で用いられるモンテカルロ法を適用し、およつち売場のアイテムの並びをより望ましいものが前に並ぶような推薦手法を提案し、実装を行った。本研究ではモンテカルロ法の適用のさせ方により、単純モンテカルロ法、UCB1 法、UCT 法の 3 種類の推薦手法を提案する。さらに、それぞれのアルゴリズムが提示するアイテムの並びを単純な低カロリー順と比較し、その特徴を明らかにした。その結果、UCT 法による推薦が最も低カロリー順に近くかつ多様な順番を提示することが明らかとなった。

## 1. はじめに

健康の増進において食生活の向上は重要な位置を占め、食育という考え方が一般にも浸透してきている。特に現代では、あらゆる嗜好食品、嗜好飲料が流通しており、学童期から間食を多くとりがちなおよつちが多い。したがって、普段食べているおよつちの種類や食べ方を見直し、およつちを上手に食べることを教育する必要がある。この目的のために教師と子供が対話形式でスーパーでのおよつちの買物を例題にした取り組みがなされている。

一方、コンピュータゲームにおける知能処理では、近年統計的判断にもとづく手法が有効であることが知られている。これは、 $k$ -多腕バンディット問題をモンテカルロ法により効率的に探索するアルゴリズム [1] をゲームにおける着手選択に応用し [2]、近年その強さを向上させている。コンピュータ囲碁では、現在強いプログラムはほぼすべてモンテカルロ法をベースにした手法で開発されていると言える [3]。また、カードゲームである大貧民においてもモンテカルロ法によるプログラムは強い結果を残している [4]。さらに  $k$ -多腕バンディット問題は、ゲーム以外の分野でも有用であるという報告がなされている [5]。

本研究では、子供がスーパーでおよつちを買物する状況

を題材に、およつちの正しい選択を教育する電子教材を作成した。さらにその買物ゲームにおいて、およつちをより良い順に表示させる推薦機能を付け加え、子供の選択の手助けとなるようにした。推薦機能はモンテカルロ法をベースに 3 種類の手法を提案し、その特徴を評価した。

## 2. 食育ゲームの概要

本研究では小学生から中学生（6 歳から 14 歳）を対象に、およつちの選択を通じてカロリーの取り過ぎに関して理解を深めるゲームを作成した。ゲームはそれ自体単独で利用することもできるが、教師との対話により教育を行う際の補助ツールとしての利用方法が主な使い方である。

### 2.1 指導方法の流れ

本研究で想定している食育指導は、小中学校における学級活動において、教師と子供が対話形式でゲームを行うことにより行われる。教師との対話は以下のような流れで行われる。

- (1) 子供に、スーパーに来ておよつちを選択して買物する場面であることを想定させる。
- (2) スーパーに並んでいるおよつちが印刷された用紙と、買物をするバスケットに見立てたアンケート用紙を配布する。ここでそれぞれのおよつちには値段が書き込まれている。この値段はそのおよつちのカロリー量と同じ値である。

<sup>1</sup> 岡山県立大学 情報システム工学科  
111, Kuboki, Soja, Okayama 719-1197, Japan  
<sup>2</sup> 岡山県立大学 栄養学科  
<sup>a)</sup> tajima@cse.oka-pu.ac.jp

表 1 ユーザ属性と 1 日の摂取目安カロリー

	ふつうの子		よく運動する子	
	男子	女子	男子	女子
6 - 8 歳	1650	1500	1900	1700
9 - 11 歳	1950	1750	2250	2050
12 - 14 歳	2200	2000	2550	2300

- (3) まずは、1 日のおやつとして好きなものを好きなだけ選択させ、アンケート用紙に記入させる。
- (4) 選択が終わったら、各自の選んだおやつと値段の合計金額を求めさせる。
- (5) 合計金額が 1 日の摂取量と比較してどうであるかを教師が説明する。
- (6) あらためて、おやつを選択をし直させ、どのような食品がどのようなカロリーを持ち 1 日の摂取量としての影響を考えさせる。
- (7) さらに、この授業の事前におやつを食べる時間や回数、分量などをアンケート調査しておき、その内容に沿った紙芝居を用いて正しいおやつと選択、食べ方を指導する。

以上が指導の流れであり、

- 事前のアンケート調査：おやつをいつ食べるか、1 日何回食べるか、どの程度食べるかなどの事前調査。
- 授業当日のお買い物ゲーム：普段何気なく選択しているポテトチップスなどがどれほど高カロリーであるかを認識させるゲーム。

の 2 段階から成り立っている。本研究では、この後の段階であるお買い物ゲームを補助する電子教材を作成した。

## 2.2 ゲームの仕様

作成したゲームは、以下の要素から構成される。

- 利用者のプロフィールを入力する部分
- スーパーの売り場に相当するおやつと値段の選択部分
- 買い物カゴに相当するおやつと値段の集積部分
- 買い物合計金額を算出表示する部分

利用者のプロフィールとは、ゲームを行う子供の年齢、性別、1 日の運動量などで摂取基準となるカロリーが異なるため、あらかじめ入力させておくユーザの属性である。本ゲームでは

- 年齢：6-8 歳、9-11 歳、12-14 歳の 3 区分
- 性別：男子、女子の 2 区分
- 運動：ふつう、よく動くの 2 区分

である。1 日の摂取カロリーの目安は表 1 であり、おやつと適量は

1 日のカロリーの 10-15%

である。

スーパーの売り場に相当する部分には、「おかし」、「パンやケーキ」、「のみもの」、「くだもの」の 4 カテゴリーに分けられた全 35 アイテムを準備した。それぞれのアイテムに

表 2 おやつアイテムと値段

カテゴリ名	アイテム名	値段
おかし	あめ 1 個	20
	ガム 1 個	10
	板チョコ 1 枚	370
	ミニチョコ 1 個	50
	ポテトチップス 1 袋	500
	バナナアイス 1 個	230
	せんべい 1 枚	30
	みたらし団子 1 本	100
	プリン 1 個	200
	ヨーグルト 1 個	140
	ゼリー 1 個	30
パンやケーキ	クッキー 1 枚	50
	メロンパン 1 個	450
	あんパン 1 個	400
	クロワッサン 1 個	160
	ショートケーキ 1 カット	350
	チョコレートケーキ 1 カット	300
	パウンドケーキ 1 切	190
	ドーナツ 1 個	200
	おにぎり 1 個	100
	ハンバーガー 1 個	250
	たこ焼き 1 個	30
のみもの	ウーロン茶	0
	紅茶	30
	コーラ	90
	オレンジジュース	100
	カフェオレ	100
	牛乳	140
	くだもの	りんご 1 個
キウイ半分		80
バナナ 1 本		130
梨 1 個		80
みかん 1 個		40
ぶどう 1 粒		10
もも 1 個		60

はカロリー量に相当する値段がつけられている。表 2 は、それぞれのアイテムと値段の詳細である。

買い物カゴには、売り場で選択した任意のおやつアイテムを任意の個数保存できるものとした。買い物カゴに入れたおやつアイテムの合計は、いつでも計算でき、その結果が表示部に示される。指導では初めの選択は自由に行わせ、一度選択を終えてから合計金額を見せるようにしているため、合計金額の表示はユーザからの指示がなければ更新されない。

## 3. 実装

実装は AndroidOS を用いて行った。ゲーム画面イメージを図 1 に示す。さらに、図 2 には、おやつアイテムの表示の詳細を、図 3 には、カートにおやつアイテムを積んだところを示す。

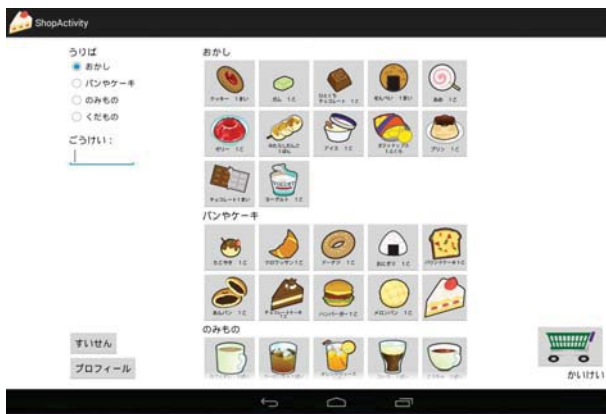


図 1 ゲーム画面



図 4 プロファイル設定

より会計を行う。一度、会計が行われると画面左側の合計欄に合計金額が表示され、以後この値を元にカートの中身を調整することにより、よりよいおやつを選択ができるように指導する。画面左側のプロフィール設定ボタンを押すと図 4 の画面となりユーザのプロファイルを設定することができる。

#### 4. 選択肢の推薦方法

指導において十分な時間をこのゲームに使うことができれば、試行錯誤により正しくかつ好みに合ったおやつアイテムの選択ができるようになる。しかし、多くの時間を割けないときやユーザである子供の性格などによっては、より低カロリーのアイテムを優先的に選ばせて指導に結びつける必要がある。おやつアイテムの並び順を価格の低い順 (= 低カロリー順) に並べても良いが、その場合は何を指導したいのかが小さな子供にもすぐに明らかになってしまう。

そこで本研究では、ゲーム木探索で用いられているモンテカルロ法を使い、より低カロリーのおやつアイテムを適度に順番を変えながら表示を行う「すいせん」機能を実装した。この機能では、現在の買い物カゴの状況に応じて低カロリーなアイテムを高い確率で前の方に表示させ、子供の目につきやすくすることで正しい選択をうながす。モンテカルロ法の利用法により以下の 3 通りのアルゴリズムを実装し、単純に低カロリー順に並べた場合との並び順の比較を行った。

##### 4.1 単純モンテカルロ法による推薦

以後、おやつアイテムの集合を  $X$  で表し、アイテムひとつを  $x_i$  で表す。すなわち  $X = \{x_1, x_2, \dots, x_{35}\}$  である。総アイテム数を  $m$  で表す。本研究の場合は  $m = 35$  である。買い物カゴを  $K$  で表し、カゴの要素を  $(x_i, n_i)$  と表し、アイテム  $x_i$  が  $n_i$  個含まれていることとする。したがって、 $x_1$  が 2 個、 $x_3$  が 1 個、 $x_4$  が 3 個含まれている



図 2 おやつアイテムの表示

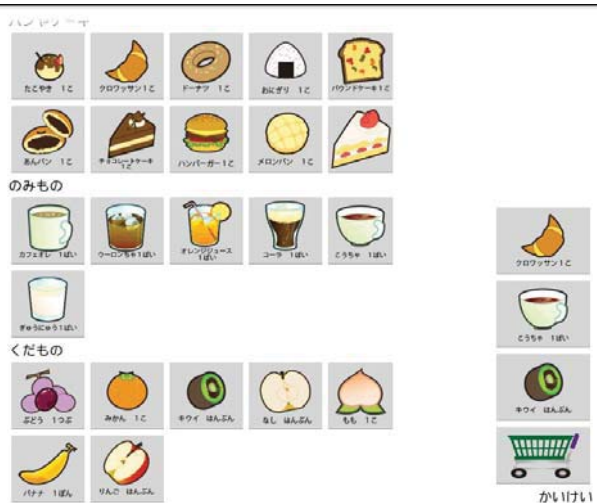


図 3 カートにおやつを入れたところ

画面左側は利用者のプロフィール選択および売り場へのショートカットである。画面中央におやつアイテムが並べられ、それぞれがボタンとなっている。選択はアイテムをタップすることにより行われる。タップされたアイテムは画面右側のカートに積まれ、カートのボタンを押すことに

```
[simple montecarlo]
 $r_i := 0, w_i := 0, g_i := 0$  for all  $i = 1, \dots, m$ 
for maxgame times
do
   $K_r := K$ 
  ランダムにひとつ  $x_j$  を選び  $K_r$  に加える
  while ( $K_r$  中のアイテムの合計金額  $t_r < u$ )
  do
    ランダムにひとつ  $x_k$  を選び  $K_r$  に加える
  done
  if ( $|K_r| > l$ ) then
     $w_j = w_j + 1, g_j = g_j + 1$ 
  else
     $g_j = g_j + 1$ 
  fi
done
 $r_i = w_i/g_i$  for all  $i = 1, \dots, m$ 
 $r_i$  値が高い順に  $x_i$  を売り場に並べる
```

図 5 単純モンテカルロ法

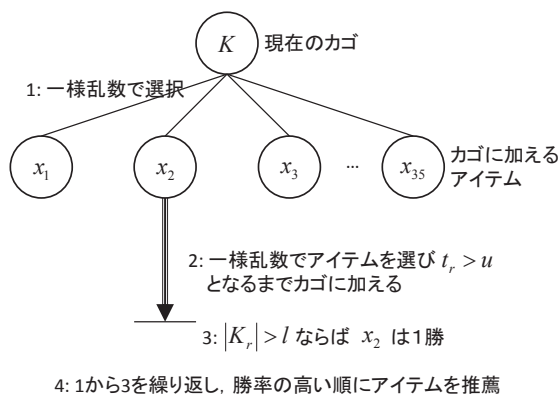


図 6 単純モンテカルロ法の動き

カゴは,  $K = \{(x_1, 2), (x_3, 1), (x_4, 3)\}$  と表される. ゲームをプレイ中のユーザのプロファイルから得られたおやつのおよその合計金額の上限を  $u$  とする. さらに, カゴの中身の合計金額を  $t$  とする.

理想的なカゴの中のアイテム数に相当する数を  $l$  とする. この値は, ゲーム作成者の側であらかじめ決めておく. モンテカルロ法のゲーム木探索におけるプレイアウト数に相当する数  $maxgame$  もあらかじめ決めておく. これは利用する機器の計算性能と推薦に利用できる計算時間から調整する必要がある. モンテカルロ法によるアイテム  $x_i$  の推薦度合い  $r_i$  は, 図 5 のアルゴリズムで定められる. これは, 理想とするバスケット内のアイテム数  $l$  をランダムに選んだアイテムで越えられるかどうかを調べ, 越えられれば勝ち, 越えられなければ負けとみなして勝率を求めている. 推薦はこの勝率の高い順にしたが行われる.

繰り返し回数  $maxgame$  を十分大きな値に定めれば, 低

```
[UCB1 ordering]
 $r_i := 0, w_i := 0, g_i := 0$  for all  $i = 1, \dots, m$ 
for maxgame times
do
   $K_r := K$ 
  UCB1 アルゴリズムで  $x_j$  を選び  $K_r$  に加える
  while ( $K_r$  中のアイテムの合計金額  $t_r < u$ )
  do
    ランダムにひとつ  $x_k$  を選び  $K_r$  に加える
  done
  if ( $|K_r| > l$ ) then
     $w_j = w_j + 1, g_j = g_j + 1$ 
  else
     $g_j = g_j + 1$ 
  fi
done
 $r_i = w_i/g_i$  for all  $i = 1, \dots, m$ 
 $r_i$  値が高い順に  $x_i$  を売り場に並べる
```

図 7 UCB1 法

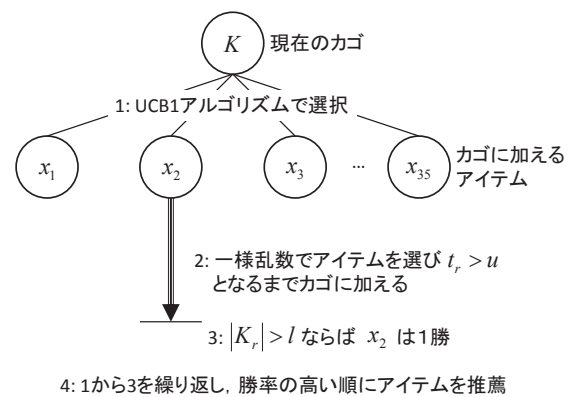


図 8 UCB1 法の動き

カロリーなアイテムの順に並ぶが, 計算時間の都合からいくつか順番が入れ替わるにより推薦の単調さを防いでいる. 図 6 に動作の模式図を示す.

#### 4.2 UCB1 アルゴリズムを用いた推薦

単純モンテカルロ法に対して計算時間の短縮の視点から, 勝率を求めるアイテムの選択を UCB1 アルゴリズムとした推薦手法である. 図 7 と図 8 にそれぞれアルゴリズムとその動作の模式図を示す. UCB1 法のアルゴリズムと単純モンテカルロ法の相違点は,  $x_j$  選択のアルゴリズムのみである.

ここで UCB1 アルゴリズム [1] とは, k-armed バンデット問題に対して真の勝率から外れた腕を選ぶ確率が十分小さく抑えられるアルゴリズムであり, このゲームに適用すると, 以下のように  $x_j$  を選択する.

(1)  $k = \sum_{i=1}^m g_i$  すなわち繰り返しの総数とする



```
[UCT ordering]
 $r_i := 0, w_i := 0, g_i := 0$  for all  $i = 1, \dots, m$ 
すべての  $x_i$  から子節点  $x_{i,i'}$  ( $i' = 1, \dots, m$ ) を生成し
 $r_{i,i'} := 0, w_{i,i'} := 0, g_{i,i'} := 0$  とする
for maxgame times
do
   $K_r := K$ 
  UCB1 アルゴリズムで  $x_j$  を選び  $K_r$  に加える
  UCB1 アルゴリズムで  $x_{j,j'}$  を選び  $K_r$  に加える
  while ( $K_r$  中のアイテムの合計金額  $t_r < u$ )
  do
    ランダムにひとつ  $x_k$  を選び  $K_r$  に加える
  done
  if ( $|K_r| > l$ ) then
     $w_j = w_j + 1, g_j = w_j + 1$ 
     $w_{j,j'} = w_{j,j'} + 1, g_{j,j'} = g_{j,j'} + 1$ 
  else
     $g_j = g_j + 1$ 
     $g_{j,j'} = g_{j,j'} + 1$ 
  fi
  すべての  $i = 1, \dots, m, i' = 1, \dots, m$  について
   $r_i = w_i/g_i, r_{i,i'} = w_{i,i'}/g_{i,i'}$  とする
done
 $r_i$  値が高い順に  $x_i$  を売り場に並べる
```

図 9 UCT 法

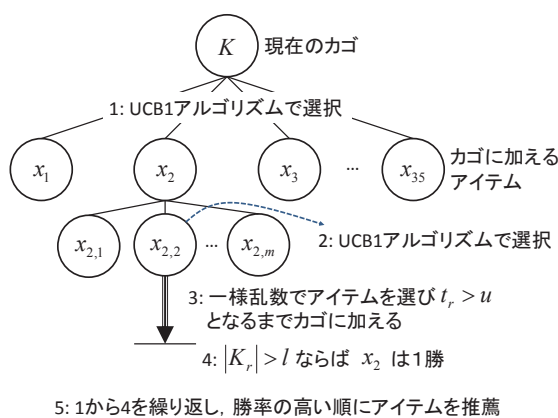


図 10 UCT 法の動き

(2) すべての  $x_i$  について  $r_i + \sqrt{2 \log k/g_i}$  を求め、この値が最も高いものを  $x_j$  とする

### 4.3 UCT アルゴリズムを用いた推薦

現在の買い物カゴの状態から 2 アイテムを選択し、カゴに加えた状態までをゲーム木の要領で展開し、木の中での選択は UCB1 で行い、木の葉からは一様乱数でアイテムを選択する手法である。ゲーム木探索において UCT アルゴリズム [2] と呼ばれている手法に類似しているため、以後 UCT 法と呼ぶ。図 9 と図 10 にそれぞれアルゴリズムとその動作の模式図を示す。

UCT 法の特徴は、買い物カゴに入れる 2 つ目のアイテムをランダムではなくそれまでの統計情報から選択する点にある。これにより、1 つ目のアイテムの勝率が高いものほどより高い値にシフトするため、推薦順位がより極端に定まる傾向がある。これは、単調な推薦状況を回避する上で有効な手法であると言える。

## 5. 推薦の内容の比較

単純にカロリーが低い順番にアイテムを並べたものを Low Cost Order (LCO) と呼ぶ。推薦に用いたアルゴリズムを単純モンテカルロ法を Random と呼び、UCB1 法と UCT 法をそれぞれ UCB1, UCT と呼ぶ。それぞれのアルゴリズムを用いて作成されたアイテムの並び順をそれぞれ Random Order (RO), UCB1 Order (UO), UCT Order (TO) と呼ぶ。

RO, UO, TO と LCO との違いを以下の評価基準で比較する。

LCO でのアイテム順序を  $C_1, C_2, \dots, C_m$  ( $C_i \in X$ ) とし、RO, UO, TO でのアイテム順序をそれぞれ  $R_1, \dots, R_m$  と  $U_1, \dots, U_m$  および  $T_1, \dots, T_m$  と表す。ここで各要素はおやつアイテム  $R_i \in X, U_i \in X, T_i \in X$  ( $i = 1, \dots, m$ ) である。ある  $x_i \in X$  について、 $x_i = C_a$  となる  $1 \leq a \leq m$  を  $C(i)$  と表す。すなわち LCO における  $X_i$  の順位である。同様に RO における順位を  $R(i)$ , UO における順位を  $U(i)$ , TO における順位を  $T(i)$  と表す。LCO と RO の順位類似度  $D(LCO, RO)$  を以下のように定義する。

$$D(LCO, RO) = \sum_{i=1}^m |C(i) - R(i)|$$

同様に

$$D(LCO, UO) = \sum_{i=1}^m |C(i) - U(i)|$$

$$D(LCO, TO) = \sum_{i=1}^m |C(i) - T(i)|$$

と定義する。上記の値をそれぞれ 20 回推薦を行なって求めた結果を示す。表 3 が maxgame = 500 とした場合であり、表 4 が maxgame = 1000 とした場合である。ここで理想のアイテム数  $l$  は 3 である。

どちらの結果においても  $D()$  の値は TO, UO, RO の順で大きくなっている。これは LCO との違いも TO, UO, RO の順で大きくなっていることを示している。しかし、標準偏差の値は RO, UO, TO の順で大きくなっており、UCT 法による推薦順位の方が単純モンテカルロ法よりもばらつきが大きいことを示している。これは、より多様な順序で表示されることを示しており、本研究の目的である単調さを感じさせない推薦が達成されていると言える。繰り返しの回数 maxgame の違いによる差は、標準偏差は

表 3 推薦順位の比較 (maxgame=500)

	$D(LCO, RO)$	$D(LCO, TO)$	$D(LCO, UO)$
	566	594	556
	558	598	580
	588	584	590
	584	600	578
	602	596	580
	598	574	588
	586	552	592
	600	576	580
	582	558	590
	564	556	542
	562	564	558
	602	524	596
	592	596	600
	562	590	554
	594	586	576
	594	506	568
	570	578	586
	560	610	552
	590	552	556
	584	558	598
ave	581.9	572.6	576.0
dev	14.96	25.91	17.04

表 4 推薦順位の比較 (maxgame=1000)

	$D(LCO, RO)$	$D(LCO, TO)$	$D(LCO, UO)$
	606	556	600
	576	554	604
	606	544	586
	608	576	586
	590	594	590
	606	568	584
	592	592	604
	586	598	602
	604	594	598
	604	584	584
	594	578	580
	592	604	594
	598	584	612
	596	582	610
	606	604	572
	604	568	594
	598	604	600
	598	604	602
	594	600	602
	596	564	612
ave	597.7	582.6	595.8
dev	7.96	18.13	10.82

1000 回の繰り返しの方が小さくなっているが、平均値は逆に大きくなっており、さらなるデータの取得が必要と言える。

## 6. おわりに

本研究において、食育の授業で利用可能なおやつを選択ゲームを作成し、その中でモンテカルロ法にもとづいたおやつアイテムの並べ替え手法を提案した。提案した 3 種類の手法のうち、UCT にもとづいた手法が最も多様な順序でかつ、カロリーの低い順番との相違が少ないものであることが明らかとなった。いずれの手法も携帯端末やタブレットなど本ゲームを実行する機器の計算性能に合わせて試行回数を調整することにより、推薦順位と計算結果が出るまでの待ち時間を調整することができる。また、協調フィルタリングなどの推薦手法と違い、事前に設定した評価要素にもとづいて動作するため、利用者の統計情報などが必要ない点も特徴である。

今後の課題として、各おやつアイテムについてカロリーのみでなく他の栄養成分値 [6] を含めた最適化への発展、利用者の嗜好を取り込んだ方法の開発などが挙げられる。

## 参考文献

- [1] P. Auer, N. Cesa-Bianchi, P. Fischer, Finite-time analysis of the multiarmed bandit problem, *Machine Learning*, 47(2,3), pp.235-256, 2002.
- [2] L. Kocsis, C. Szepesvari, Bandit based Monte-Carlo planning, *LNCS 4212*, pp.282-293, 2006.
- [3] R. Coulom, Efficient selectivity and backup operators in Monte-Carlo tree search, *LNCS 4630*, pp. 72-83, 2007. (Proc. of Int. Conf. on Computer and Games 2006)
- [4] 小沼 啓, 西野 哲朗, コンピュータ大貧民に対するモンテカルロ法の適用, *情報処理学会 ゲーム情報学研究会報告*, SIG-GI-25, no.3, pp.1-4, 2011.
- [5] 下田 敬祐, 江口 浩二, 多腕バンディットによる検索結果の多様化に関する大規模クリックスルーログを用いた評価, 第 4 回データ工学と情報マネジメントに関するフォーラム (DEIM2012), E6-6, 2012.
- [6] 文部科学省科学技術学術審議会資源調査分科会, 五訂増補 日本食品標準成分表, 国立印刷局, 2008.