

卒業論文

制約に基づく献立推薦システム

指導教官 村上 陽平 准教授

立命館大学情報理工学部
情報コミュニケーション学科 4 回生
2600160108-2

加藤 聖貴

2019 年度（秋学期）卒業研究 3（2Q）

令和 2 年 1 月 31 日

制約に基づく献立推薦システム

加藤 聖貴

内容梗概

大学の体育会ではチームを強くするために、チーム力、筋力などに加えて、食事に注力して取り組んでいる。食事はただ摂るだけでなく大学の栄養士が栄養素などを管理、計算して提供している。このようなアスリート向けに考えられた食事はアスリート食と呼ばれ、高カロリー、高タンパクなため、メニューの種類が限られる上に、栄養士が意識的にご飯に合う人気メニューや定番メニューを出しているため、食事のメニューが被ることがある。そのため、食事を残す部員がおり、基準の栄養素を摂取できないという問題が生じている。従来の研究では、ユーザのその日のコンディション、好き嫌い、色合いなども考慮している研究が多くある。しかしながら、アスリート食が対象としているのは一人ではなく体育会の大勢の部員である。また、アスリート食は、ユーザが献立の雰囲気などを入力し、推薦された食事を選択することができない。さらに、二週間分の献立を推薦し前週の一週間とは被らないようにすることが必要である。

そこで、本研究では同じメニューを被らないようにすることを目的とし、制約に基づく献立推薦システムを提案する。具体的には、エネルギー量などのアスリートが摂取すべき栄養素の量を制約として記述し、二週間分の献立表を出力する。またその二週間分の献立は前週の一週間とは必ず被らないようにする。本手法の実現にあたり、取り組むべき課題は以下の2点である。

献立推薦の制約充足問題としての定式化

各メニューにはいろいろな栄養素が含まれている。栄養素ごとに制約を満たす献立を探索すると、別の栄養素に影響を与える可能性がある。したがって、単純な数値の割り当てを行う制約充足問題ではなく、メニューというオブジェクトの割り当てを行う制約充足問題として定式化する必要がある。

献立の連続性の考慮

制約を満たす献立を推薦するだけでは、食品データベースが更新されない限り、同じ献立を推薦され続けるようになる。これではメニューが周期的になり、食べ残しを助長する可能性がある。

一つ目の課題に対しては、制約充足問題の解法としてSAT型ソルバーSugarを適用した際、栄養価のデータを参照しサーバーを通して献立を推薦する。また、個々の料理は支持点を用いて、その料理の持つエネルギー量やタンパク

質, 脂質を支持点集合として定義して, 制約最適化問題を解き自動で制約を満たすものを出力する. 二つ目の課題に対しては前週の一週間の献立を重複しないようにするため二週間分の献立を出力する. その二週間分の献立は, それぞれ制約で献立を被らないようにする.

提案手法の有用性を示すために, 提案手法を組み込んだ SAT 型ソルバー Sugar システムを実装し, 動作確認を行った. 本研究の貢献は以下の通りである.

献立推薦の制約充足問題としての定式化

制約充足問題として定式化するために, 各日の主菜, 副菜, デザートを変数として, 主菜, 副菜, デザートに分類される料理をドメインとして, そして各栄養価の必要摂取量および料理の重複確認を制約として定義した. このように献立推薦を制約充足問題として定式化することで, 今後もコストや在庫などの制約を追加することが可能である.

献立の連続性の考慮

前週の一週間と被らないことによりメニューの種類が増え, 選手たちの食事に対する飽きというのが無くなった. また, できれば前週の食べ残し量も考慮できるような拡張が可能である. さらに, 栄養士の献立と提案手法の献立を部員に比較し評価してもらい, 栄養素などはきちんと出来ているがまだ実用性はないと評価された.

Menu recommendation based on constraint

Masaki KATO

Abstract

At the university athletic club, they focus on meals in addition to team power, physical strength. Meals are more than just eating. nutritionists of university calculate and provide nutrients. It called 'Athlete Food' for athlete. This kind of menu limited because it is high calories and high protein, and the nutritionist choices popular and standard menus that go well with rice. Therefore, some members leave meals causing problems with not able to get the standard nutrients. In previous studies, there is a lot of research that consider user's condition, likes and dislikes, color, of the day. 'Athlete Food' targets athletic club of many people, not alone. For 'Athlete Food', the user can't enter the menu atmosphere, and can't select the recommendation meals.

Further, it is necessary to recommend a menu for two weeks and not to overlap the previous week. In this study, we propose a menu recommendation system based on constraints with the aim of avoiding the same menu. In details, describe the amount of nutrients that athlete should intake and output a two-week menu. Also, two-week menu not overlap the week of previous week. There are two issues to be addressed in realizing this method.

Formulation of Menu Recommendation as a Constraint Satisfaction Problem

Each menu contains various nutrients. Exploring menus that meet constraints for each nutrient can affect other nutrients. Therefore, it is necessary to formulate as a constraint satisfaction problem that assigns objects called menus, not a constraint satisfaction problem that assigns simple numerical values.

Considering menu continuity

Recommending menus that meets the constraints will continue to recommend the same menu unless the food ingredients database is updated.

For the first problem, when the SAT solver Sugar is applied as a solution to the constraint satisfaction problem, menus are recommended through a server with reference to nutritional value data. Also, dishes use bearing to define the amount

of energy, protein, fat as a set of bearing. Solve constraint optimization problems and automatically output those that satisfy constraints.

The second problem, it recommends a menu for two weeks and not to overlap the previous week. Two-week menu should overlap from menu restrictions.

The SAT solver Sugar system incorporating the proposed method was implemented and its operation was confirmed. Contributions of this research are as follow:

Formulation of Menu Recommendation as a Constraint Satisfaction Problem

To formulate a constraint satisfaction problem, the main dishes, side dishes, and desserts as variables, the dishes classified as main dishes, side dishes, and desserts as domains, and the required intake and dishes for each nutritive value to formulate as a constraint satisfaction problem Was defined as a constraint. By formulating the menu recommendation as a constraint satisfaction problem in this way, it is possible to add constraints such as cost and inventory in the future

Considering menu continuity

Menu variation has been increased by not overlap previous week and they doesn't get tired of eating. It can also consider the amount left over from the previous week. Ask my club members to evaluate the dietitian's menu and proposed method, but they said it's not so good.

制約に基づく献立推薦システム

目次

第 1 章 はじめに	1
第 2 章 献立推薦システム	3
2.1 ユーザに基づく献立推薦	3
2.2 病院における献立作成	4
第 3 章 献立推薦のモデリング	7
3.1 制約充足問題	7
3.2 制約充足問題として定式化	8
3.2.1 変数	8
3.2.2 ドメイン	8
3.2.3 制約	10
3.3 献立の連続性の考慮	10
第 4 章 システムアーキテクチャ	13
4.1 設計	13
4.2 実装	14
4.2.1 SAT 型ソルバー Sugar	14
4.2.2 献立推薦制御部	16
第 5 章 考察	18
第 6 章 おわりに	21
謝辞	23
参考文献	24

第1章 はじめに

大学の体育会スポーツにおいてチームを強くするために様々な方法があるとされ各大学はそれぞれのやり方で目標に向けて取り組んでいる。チーム力、筋力、技術力など決して欠けてはいけない要素は様々あるが、近年、特に食事に注力している大学が増加してきている。なぜなら、どれだけ高い技術力や完璧な戦術があっても身体のサイズが無ければ当たり前勝つことが出来ない。高校レベルでは通用するかもしれないが、大学レベルのスポーツでは通用しない。そのため食事を摂り身体を大きくし、当たり前負けない身体を作る必要がある。しかし、食事を摂ると言ってもジャンクフードやお菓子や肉だけの様な摂り方では逆に栄養素が偏るためにただ脂肪がつき、スポーツには向いていない。食事を摂るとはただ食べるだけでなくエネルギー量や栄養素を意識しないといけない。例えば立命館大学では、大学の栄養士がおり、その栄養士が栄養素などを計算し管理して提供している。このようなアスリート向けに考えられた食事をアスリート食と呼ばれている。

しかし、食事のメニューが被ることがあり、食事を残す部員もおり基準の栄養素を摂取出来ないという問題が生じている。なぜなら、アスリート食は、高カロリー、高タンパクなため、メニューの種類に限られる上に栄養士が意識してご飯に合う人気メニューや定番メニューを出しているためである。

そこで従来の研究では、ユーザが「風邪予防」や「眼精疲労」など改善したい体調や欲求などのその日のコンディションを選択できる研究がある。そのコンディションに合わせた献立が推薦される。また、ユーザの好き嫌いを考慮しつつ献立の雰囲気を入力しそれに適した献立を推薦される。その推薦された献立を選択することができる。さらに、病院における献立作成では、対象としているのは、多数だが、同じ病気の人などの条件が同じ人によって献立作成を行う。

しかし、従来の研究ではアスリート食の献立推薦には適していない。なぜならアスリート食は、ユーザー一人ではなく体育会の大勢の部員を対象とし、条件が同じである必要がある。また、アスリート食は、ユーザの好き嫌いや献立の雰囲気などを入力し、推薦された食事を選択することもできない。さらに、一週間の献立を推薦するなかで、前週の一週間の献立と被らないようにすることが必要である。これは、メニューが限られている中で被ると飽きが生まれ食べ残しが起こるためである。

このような現状の問題の取り組むべき課題は、以下の 2 点である。

献立推薦の制約充足問題としての形式化

各メニューにはいろいろな栄養素が含まれている。したがって、栄養素ごとに制約を満たす献立を探索すると、別の栄養素に影響を与える可能性がある。したがって、単純な数値の割り当てを行う制約充足問題ではなく、メニューというオブジェクトの割り当てを行う制約充足問題として定式化する必要がある。

献立の連続性の考慮

制約を満たす献立を推薦するだけでは、食品データベースが更新されない限り、同じ献立を推薦され続けるようになる。これではメニューが周期的になり、食べ残しを助長する可能性がある。

本研究では、上記の 2 点の課題に取り組み、最終的には同じメニューが被らないようにするシステムを構築することを目標としている。

以下本論文では、第 2 章で献立推薦に関する関連研究のユーザに基づく献立推薦と大人数を対象とした給食献立推薦を説明する。続いて、第 3 章で提案手法の献立推薦のモデリング、その際に必要となる制約充足問題と献立の連続性の考慮について述べる。その後、第 4 章で提案手法を用いたシステムについての説明を述べる。第 5 章で提案手法の評価を行い提案手法に対する考察を行う。第 6 章で本稿をまとめ今後の展望や課題について述べ結論とする。

第2章 献立推薦システム

本章では、既存の献立推薦システムについて説明する。大別して、ユーザに基づく献立推薦と大人数を対象とした給食献立支援が存在するので、それらを順に説明する。

2.1 ユーザに基づく献立推薦

ユーザに基づく献立推薦は、一人に対して推薦される。個人のコンディションや欲求を考慮した献立推薦を説明する。(図1)

まず、ユーザが目標とする栄養素バランスを設定するにあたり、基準値バランスを設定する[1]。これは、厚生労働省より取り決められている「日本人の食事摂取基準」を参考にする。これで、一般的な年齢や性別などによって違う1日に必要とする栄養素を定める。

次に、ユーザが必要とする栄養素を推薦に反映するために、必要栄養素バランスを調整する。具体的には「風邪予防」や「眼精疲労」、「美肌」などの改善したい体調や欲求を測り、明確な状況とそれに対する必要栄養素を把握する。そして、1日に必要とする栄養素量を2食分の栄養素量を差し引くことで朝食・昼食

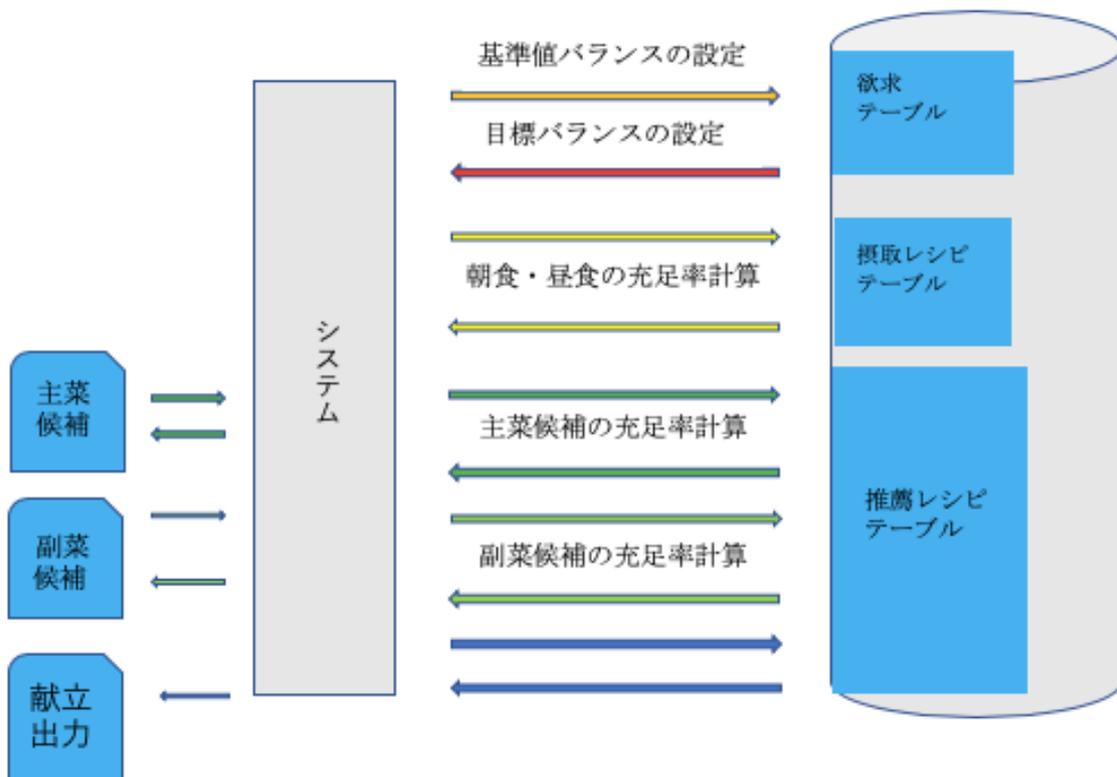


図1 ユーザに基づく献立推薦システム

で目標栄養素量に充足される割合を求め、夕食で必要となる栄養素量を算出する。最後に夕食で必要となる栄養素量を元に推薦レシピの充足率を算出する。1つのレシピが含む栄養素は、そのレシピで使われている食材とその分量を用いて食品成分表を参考にレシピで補える栄養素を算出する。レシピデータを主菜・副菜のカテゴリに分類する。主菜・副菜をそれぞれ推薦するにあたり、不足している栄養素を補っている上位5品を候補として提示し、ユーザが選択できるようにする手法を提案した。提案された手法は、評価としては良いが体調や欲求を測ることができず、また不足している栄養素を補うレシピを選択することができない。

2.2 病院における献立作成

病院における献立作成は、人に合わせるのではなく、同じ病気を抱える患者群ごとに対して献立を作成する[2]。

病院食は入院している患者を対象に、傷病の治療、健康の保持、増進を目的に医療の一環として行われるため、治療に有効である。また、患者個々の性別・年齢・体格・生活活動強度・症状などからエネルギーおよび栄養素量を決定し、さらに食欲などを考慮した質の高い内容の食事を提供しなければならない。また、献立作成において疾病の種類、症状などを踏まえ、治療に適したエネルギーおよび栄養素量の供給と使用食品などの配慮が必要である。病院の栄養基準には疾病別分類があるが、疾病別分類とは、糖尿病食、腎臓病食というように、疾病別に食事の種類を分類する方法で、この研究では、従来の疾病別分類の栄養基準を栄養食事療法の方針に基づき、エネルギーコントロール食・たんぱく質コントロール食・脂質コントロール食などの栄養成分の特徴で治療食を分類した栄養成分別分類に変更し、同時に治療食の水準を保ちながら、献立作成を行う。その際に献立のパターンを作成する。献立のパターン化とは、同じ傾向の栄養成分と形態の食種12種類の基本献立としてまとめ、基本的には主食の増減で栄養基準を満たすようにしたものである。パターン化した図2に示す。エネルギーコントロール食1400kcal常菜を基本献立とし、その主菜と副菜の料理は変えずに主食量を交換し、エネルギーコントロール食1200kcal常菜、エネルギーコントロール食1600kcal常菜、一般食1400kcal常菜、一般食1600kcal常菜の栄養基準を満たすようにした。つまり、1種類の献立で5種類の食種に対応できるように規則性を持たせる。

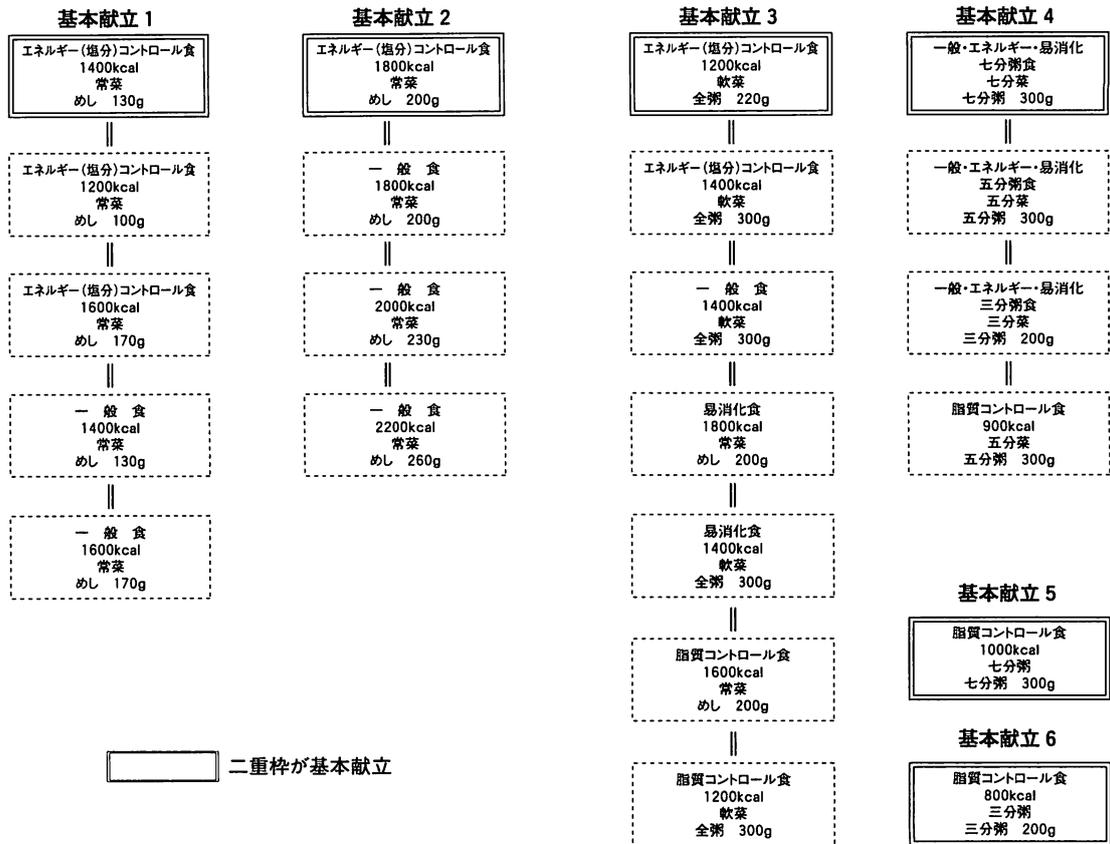


図2 献立作成のパターン たんぱく質コントロール食

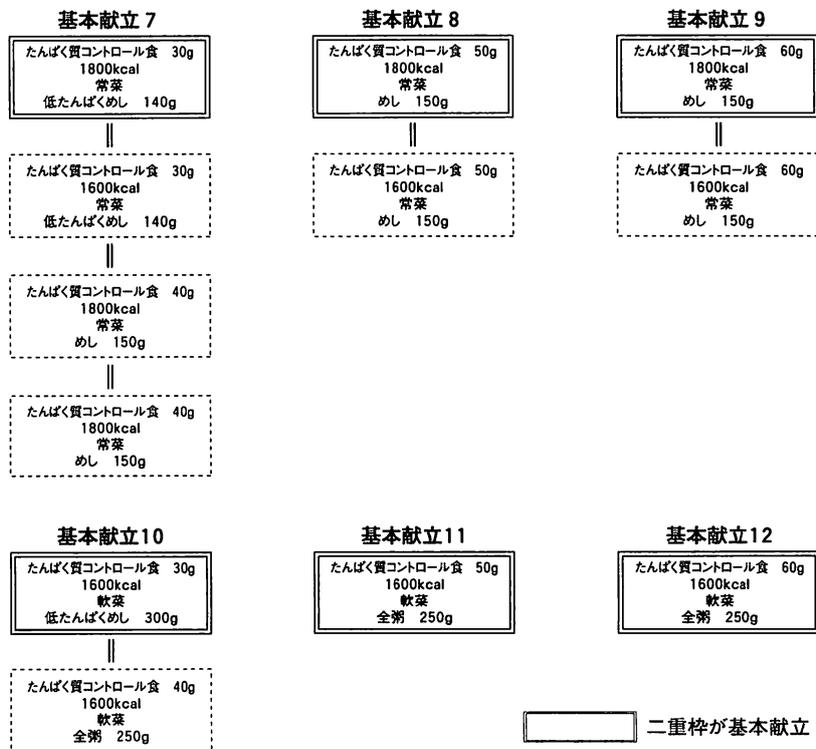


図2 献立パターン

は別の基本献立を立て、主食量の変更で一般食 1800kcal 常菜，一般食 2000kcal 常菜，一般食 2200kcal 常菜も対応できるようにした。

次に、軟食エネルギーコントロール食 1200kcal 軟菜を基本献立とし、エネルギーコントロール食 144kcal 軟菜，一般食 1400kcal 軟菜，易消化食常菜・軟菜，脂質コントロール食常菜・軟菜にも対応した。

分粥菜は、一般食・エネルギーコントロール食・易消化食それぞれの七分菜と五分菜と同じ主菜と副菜にした。脂質コントロール食七分菜とその三分菜は脂質の制限が厳しいため別々の献立を立てた。

たんぱく質コントロール食 30g・1600kcal 常菜と軟菜，たんぱく質 50g・1600kcal 常菜と軟菜，たんぱく質 60g・1600kcal 常菜と軟菜は主食のみ交換では対応できないので、それぞれ献立を立てた。たんぱく質コントロール食 40g・1600kcal 常菜と軟菜は、たんぱく質コントロール食 30g・1600kcal の主菜・副菜と同じであるが、30g の主食が治療用特殊食品の低タンパク質ご飯を使用しているため、普通のご飯に交換することでたんぱく質量が1日 10g 増加し、40g の栄養基準を充足することとした。

以上のことから基本献立は12種類に設定し、食種数が増え患者個人の年齢・性別・体格・摂取量などに合わせた選択が行えるようになったが、基本献立が12種類であれば本研究の目的である前週と被らない14日間の献立推薦に満たしていない。そのため飽きなどが生じてしまう可能性がある。

第3章 献立推薦のモデリング

3.1 制約充足問題

制約充足問題は、各変数に与えられたドメインから値を割り当てることで、与えられた制約の全てを満たすことができるかどうかを判定する問題である。すべての制約を満たす値割当てが存在する場合、元の制約充足問題は充足可能であるといい、その値割当てが解となる。値割当てが存在しない場合、もとの制約充足問題は充足不能であるという[3]。

制約充足問題は、一般には実数や集合などさまざまなドメイン上で展開されるが、実用上多くの応用を含む整数の有限領域上の制約充足問題を対象とする。

整数有限領域の制約充足問題は、形式的には以下のように定義できる。定義としては、制約充足問題は以下を満たす組 $\text{CSP}(X, \text{Dom}, C)$ である。
1 つ目に、 X は整数変数の有限集合 $\{x_1, x_2, \dots, x_n\}$ である。2 つ目に、関数 $\text{Dom}: X \rightarrow \text{Fin}(\mathbb{Z})$ は、各変数の取り得る値集合（ドメイン）を定める。（ $\text{Fin}(\mathbb{Z})$ は整数 \mathbb{Z} の有限部分集合の全体を表す）。すなわち、各変数 $x_i \in X$ について $\text{Dom}(x_i) \subset \mathbb{Z}$ であり、 $\text{Dom}(x_i)$ は有限集合である。3 つ目に、 C は X 上の制約の有限集合 $\{C_1, C_2, \dots, C_m\}$ であり、制約の連言を表す。

制約には、算術論理演算などで条件が記述されている内包的制約、制約を満足する（あるいは制約に違反する）値の組の集合が陽に与えられている外延的制約、そして **alldifferent** などに代表されるいわゆるグローバル制約がある。

内包的制約は、通常の算術演算、算術比較に加え、否定 (\neg)、連言 (\wedge)、選言 (\vee)、含意 (\Rightarrow)、同値 (\Leftrightarrow) などの論理演算を用いて条件を表したものである。例えば、 $(x_1 + 2 \leq x_2) \vee (x_2 + 3 \leq x_1)$ は内包的制約である。

外延的制約では、制約を満足する値の組の集合である支持点集合あるいは制約に違反する値の組の集合である違反点集合が与えられる。例えば、支持点集合 $R = \{(0, 0), (1, 1), (2, 2)\}$ とし、整数変数 x_1, x_2 のドメインを $\{0, 1, 2\}$ とする。このとき、外延的制約 $R(x_1, x_2)$ は $x_1 = x_2$ を表す。同様に、 $R = \{(0, 0), (1, 1), (2, 2)\}$ が違反点集合の場合、外延的制約 $R(x_1, x_2)$ は $x_1 \neq x_2$ を表す。

グローバル制約は、複数の変数に対する複雑な（意味のある）条件を簡単に表すために導入された。例えば **alldifferent**(x_1, x_2, \dots, x_n) は、 x_i が互いに異なる事を表す。 $x_i \neq x_j$ を個別に記述するよりも簡潔になり、また効率良い解法アルゴリズムの

存在がある.

上記の手法を用いて献立における制約の記述を参考にして取り組む.

3.2 制約充足問題として定式化

各メニューにはいろいろな栄養素が含まれている. 栄養素ごとに制約を満たす献立を探索すると, 別の栄養素に影響を与える可能性がある. したがって, 単縦な数値の割り当てを行う制約充足問題ではなく, メニューというオブジェクトの割り当てを行う制約充足問題として定式化する必要がある. そのため 3.1 で説明した手法を厳密に議論するために本論で扱う対象を厳密に定式化する.

3.2.1 変数

日にちごとの料理のカテゴリ (主菜, 副菜, デザート, 主食) 別の献立を格納するための変数 $x_{ij} \in X$ を定義する. i は i 日目の料理であることを示し, j は料理のカテゴリを表す. 具体的には, 主菜のカテゴリは「1」, 副菜のカテゴリは「2」, デザートのカテゴリは「3」, 主食のカテゴリは「4」とする. このようにすることで, 日にち毎の料理カテゴリ別のカロリーやたんぱく質などの栄養素に制約を設けることができる.

3.2.2 ドメイン

主菜, 副菜, デザート, 主食のカテゴリ別に分類された料理インスタンスをドメイン D として定義する. 料理集合 D 中の料理インスタンスは各栄養素を属性としてもつため, 料理インスタンスを $d_i \in D$ とすると, この料理インスタンスのカロリーは $d_i.calorie$, タンパク質は $d_i.protein$, 脂質は $d_i.fat$, 炭水化物は $d_i.carbohydrate$ で表される. 本研究で用いた料理インスタンスの食品成分データを図 3 に示す. このリストでは料理を主菜, 副菜, デザート, 主食で分類して料理 ID を振り分けることも可能であるが, データを並べ替えの作業量が増え, また料理を追加した場合も ID の振り直しが必要になる. そのため, 主菜, 副菜, デザート, 主食を分類するために図 3 のようにタイプ属性を追加した. 具体的には, 主菜は「main」, 副菜は「sub」, デザートは「dessert」, 主食は「rice」に分類した. このようにするとデータの並べ替えの必要がなく, また, 料理の追加が容易になる. また, **relation** と呼ばれる支持点を用いる定義がある. この **relation** をタイプ別に分類した 4 つのカテゴリを **relation** の支持点として定義することで, 制約などがありデータを読み込み探すのではなく, **relation** でタイプ別に分

料理名,ID,タイプ,カロリー,タンパク質,脂質,炭水化物←

うどん,1,rice,105,2.6,0.4,21.6←

そうめん,2,rice,127,3.5,0.4,25.8←

中華めん,3,rice,149,4.9,0.6,29.2←

スパゲッティ,4,rice,165,5.4,0.9,32.0←

生パスタ,5,rice,247,7.8,1.9,46.9←

こめ,6,rice,983,16,4,2.7,213←

そば,7,rice,132,4.8,1.0,26.0←

がんもどき,8,sub,228,15.3,17.8,1.6←

糸引き納豆,9,sub,200,16.5,10.0,12.1←

挽きわり納豆,10,sub,194,16.6,10.0,10.5←

トマト,11,sub,19,0.7,0.1,4.7←

いちご,12,dessert,34,0.9,0.1,8.5←

いちじく,13,dessert,54,0.6,0.1,14.3←

かき,14,dessert,60,0.4,0.2,15.9←

オレンジ,15,dessert,46,0.9,0.1,11.8←

グレープフルーツ,16,dessert,38,0.9,0.1,9.6←

パイナップル,17,dessert,51,0.6,0.1,13.4←

ハスカップ,18,dessert,53,0.7,0.6,12.8←

パッションフルーツ,19,dessert,64,0.8,0.4,16.2←

ぶどう,20,dessert,59,0.4,0.1,15.7←

ブルーベリー,21,dessert,49,0.5,0.1,12.9←

もも,22,dessert,40,0.6,0.1,10.2←

りんご 皮むき,23,dessert,57,0.1,0.2,15.5←

りんご 皮つき,24,dessert,61,0.2,0.3,16.2←

べにざけ,25,main,177,28.5,6.0,0.1←

さんま,26,main,270,23.9,17.9,0.2←

ぶり,27,main,304,26.2,20.4,0.3←

ベーコン,28,main,405,12.9,39.1,0.3←

ウインナーソーセージ,29,main,321,13.2,28.5,3.0←

図3 食品成分データ

類している料理 ID が当てはまる。この料理 ID に対応して栄養素も relation 内にあるため容易に献立を探しやすくなる。

3.2.3 制約

献立から高タンパク質ではなくただの高カロリーの料理などを除くためにカロリー、タンパク質、脂質、炭水化物のある程度の基準を設けることでお菓子など献立推薦にふさわしくないものを除外するための制約を設ける。また、各日目の主菜のカロリー、たんぱく質、脂質、炭水化物の最大摂取量および必要摂取量と料理の重複確認を制約として定義した。なお、必要摂取量および最大摂取量は管理栄養士から規定された数値を参照している。

具体的には、以下の通りである。

- ① $x_{i,\text{main}}.\text{calorie} + x_{i,\text{sub}}.\text{calorie} + x_{i,\text{dessert}}.\text{calorie} + x_{i,\text{rice}}.\text{calorie} \geq 1800$,
1回の食事のカロリーの合計が 1800kcal 以上
- ② $x_{i,\text{main}}.\text{protein} + x_{i,\text{sub}}.\text{protein} + x_{i,\text{dessert}}.\text{protein} + x_{i,\text{rice}}.\text{protein} \geq 70$,
たんぱく質は 70g 以上
- ③ $x_{i,\text{main}}.\text{fat} + x_{i,\text{sub}}.\text{fat} + x_{i,\text{dessert}}.\text{fat} + x_{i,\text{rice}}.\text{fat} \geq 50$,
 $x_{i,\text{main}}.\text{fat} + x_{i,\text{sub}}.\text{fat} + x_{i,\text{dessert}}.\text{fat} + x_{i,\text{rice}}.\text{fat} \leq 55$,
脂質は 50g~55g

として管理栄養士が献立を作成しているためその内訳は変えずに定義する。

また、料理が重複しないように、以下の制約も導入する。

- ① $x_{1,\text{main}} \neq x_{2,\text{main}} \neq x_{3,\text{main}} \neq x_{4,\text{main}} \dots$
主菜が被らない
- ② $x_{1,\text{sub}} \neq x_{2,\text{sub}} \neq x_{3,\text{sub}} \neq x_{4,\text{sub}} \dots$
副菜が被らない
- ③ $x_{1,\text{dessert}} \neq x_{2,\text{dessert}} \neq x_{3,\text{dessert}} \neq x_{4,\text{dessert}} \dots$
デザートが被らない

主食は、基本ご飯であり被っても問題がないため、重複に関する制約は設けない。なお、主食はタイプ別で「rice」と分類しているが、データの中には、うどんやカレーなども含まれている。

3.3 献立の連続性の考慮

この節では、献立の連続性の考慮について説明する。まず、一週間毎日違う献

立を推薦するのは前節で説明した通りである。それでは連続性の考慮とは、一週間献立を推薦し、次の週の献立推薦では、前週の献立と被らないようにすることである。図4で示すように、連続性の考慮の考え方は、14日間の献立を作成する。14日間の献立の場合であれば課題である前週と被らないという問題は解決される。

次に、図5で示すように、21日間の献立を考える。

21日間の献立の場合であれば前週と被らないという課題は解決されているが、三週間も空くと一週目の献立を食べたいということが起き、逆にストレスがか



図4 14日間の献立



図5 21日間の献立

かってしまう。これは実際に選手たちがあのメニューが食べたいなど三週間前のメニューが食べたいという声があった。そのため14日間であれば丁度ストレスのかからない程度であるため今回は14日間の献立推薦で行う。

第4章 システムアーキテクチャ

第3章で説明したモデリングを用いた献立推薦システムの実装の構成を説明する。このシステムは、大きく二つのコンポーネントからなる。1つ目は、推薦の実行を担う献立推薦制御部、2つ目は、制約充足問題を解決するための手法であるSAT型ソルバーのSugarである。システムの全体の構成図を図6に示す。これから設計、実装について説明する。

4.1 設計

献立推薦を行う上でどのような設計をするべきか図7で示しながら説明する。まず、食品データベースのリストを読み込む。読み込んだリストから「料理変数」「栄養素変数」の定義を行う。その変数の定義から「relation」「relationの支持点」を定義する。「制約」の定義をし、制約を満たしている献立が揃えば「推薦済み献立リスト」を作成し、CSPファイルに変換して献立を出力する。

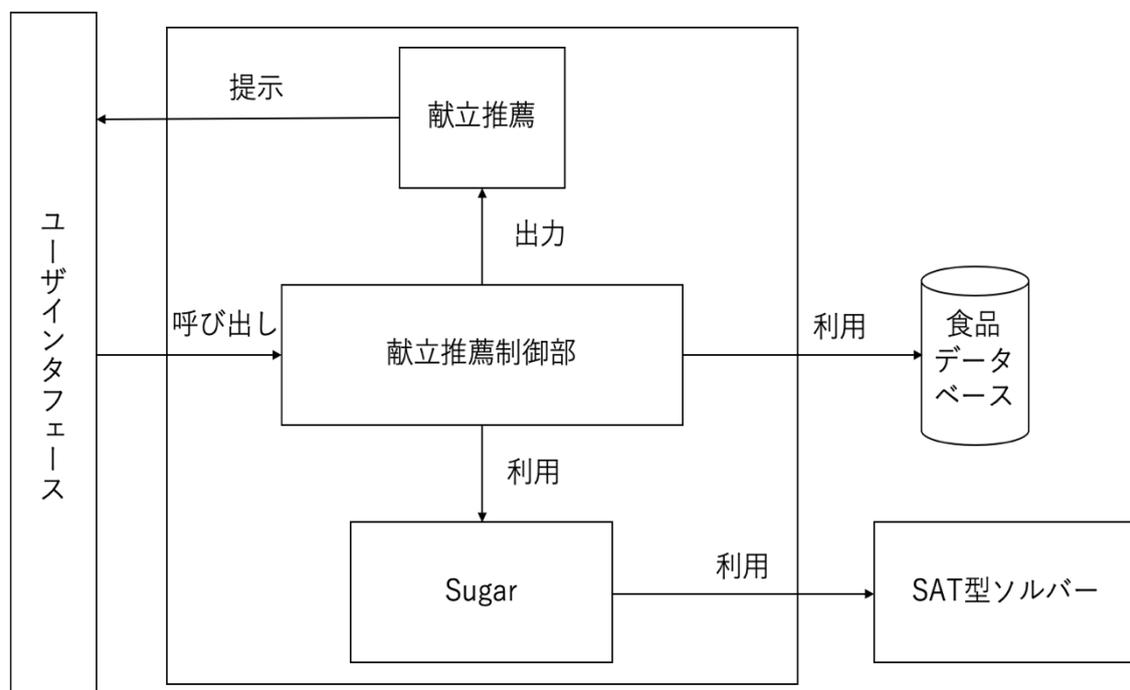


図6 推薦システムのアーキテクチャ

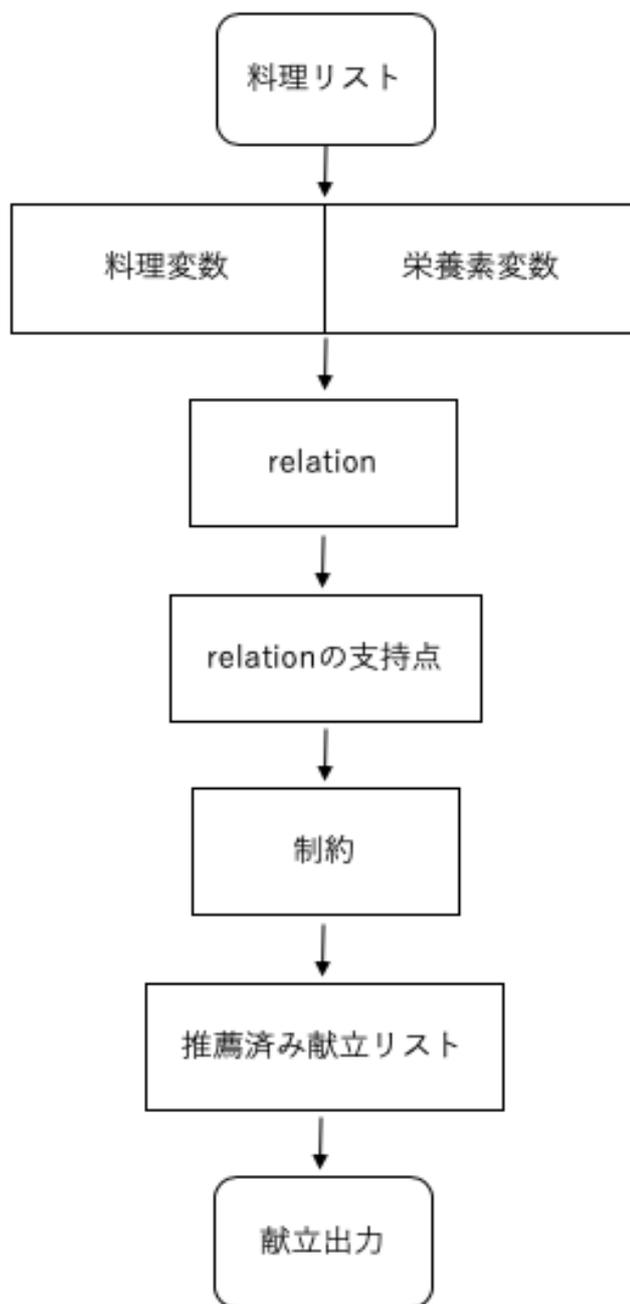


図7 献立推薦制御部のフローチャート

4.2 実装

4.2.1 SAT 型ソルバーSugar

SAT は、与えられた命題論理式を真にするような命題変数への真理値割当て



図8 SAT型ソルバー

が存在するか否かを判定する問題である[4]。真にする真理値割当てが存在すればもとの命題論理式は充足可能と呼ばれ、存在しなければ充足可能と呼ばれる。与えられた SAT が充足可能な場合、その問題を真にする真理値割当てが解となる。

命題論理式は連言標準形で与えられる。すなわち、全体の論理式はいくつかの節の連言 (AND) であり、各節はいくつかのリテラルの選言 (OR)、各リテラルは命題変数あるいはその否定である。

SAT 型制約ソルバーは、与えられた制約充足問題を SAT 問題に符号化し、SAT ソルバーを用いてもとの問題の解を求めるシステムである (図 8)。

制約充足問題の SAT 符号化にはさまざまな方法が存在する。

直接符号化法は、最も広く用いられている SAT 符号化法である。この符号化法では、各整数変数 x と各整数定数 $a \in \text{Dom}(x)$ に対して、 $x = a$ を意味する命題変数 $p(x = a)$ を用いる。制約については、それを違反点集合として表現し、各要素を節に符号化する。

多値符号化法は、直接符号化法の改良である。多値符号化法では、制約を違反点集合として表現する場合、各整数変数がただ一つの値をとるという条件が不要である。このとき、整数変数 x について、複数の $p(x = a)$ が真となる場合が生じるが、いずれの値を選んでももとの制約充足問題の解となる。

支持符号化法は、直接符号化法と同様に各整数変数 x と各変数定数 $a \in \text{Dom}(x)$ に対して、 $x = a$ を意味する命題変数 $p(x = a)$ を用いる。直接符号化法では制約の違反点集合に着目したが、支持符号化法では制約の支持点集合に着目する。なお、制約については 2 変数間の制約のみを対象とする。

対数符号化法は、各整数変数 x の 2 進数表現に着目し、 x の i 桁目が 1 に等しいことを表す命題変数を用いる。なお、変数のドメインの下限は 0 以上とし、ドメインに含まれない値は、それらを除外する節を追加する。制約については、直接符号化法と同様に制約を違反点集合として表現し、違反点集合中の各組を節に符号化する。

対数支持符号化は、対数符号化法と同様に各整数変数 x の 2 進数表現について、 x の i 桁目に対応する命題変数を用いる。ドメインに含まれない値に対して、それらを除外する節を追加する点も同様である。制約については支持符号化法と同様に支持点集合に着目して符号化する。

順序符号化法は、各整数変数 x と各変数定数 $a \in \text{Dom}(x)$ に対して、 $x \leq a$ を意味する命題変数 p を用いる。制約については、制約に違反する点ではなく違反する範囲を符号化する。

本研究では、支持点符号化法と順序符号化法を使って行う。また、Sugar を使い書いたプログラムが以下のようなになる。Sugar は性能および信頼性が高い制約ソルバーであることを前提とする[5]

4.2.2 献立推薦制御部

本研究では、支持点符号化法と順序符号化法を使って行う。また、Sugar を使い書いたプログラムが以下のようなになる。

まず、3.2.1 で説明した何日目どのカテゴリであるか示したものである。int 型にすることで自然な制約の記述、動的な制約の追加、他のプログラムとの連携などが簡単に行える (図 9)。

次に、Sugar を使った制約のプログラムを説明する。3.2.3 で説明した制約のように 1 回の食事のカロリーの合計が 1800kcal 以上、たんぱく質は 70g 以上、脂質は 50~55g としているためこのようになる。カロリーの合計が 1800kcal のように①ある数値以上など 1 つの制約であれば \geq など書けば良いが、②ある数値以上ある数値以下のような制約の場合、ある数値以上で 1 つ書き、ある数値以下で 1 つ書かないといけない (図 10)。

この制約は、①主菜が被らないように左の日数だけ数値は増えるが、右のカテゴリは、そのままにしている。②、③も同様に副菜、デザートも被らないようにしている (図 11)。

```
(int x_11 1 5) ;1日目の主菜の料理ID
(int x_12 1 5) ;1日目の副菜の料理ID
(int x_13 1 7) ;1日目のデザート of 料理ID
(int x_14 1 3) ;1日目の主食の料理ID

(int x_21 1 5) ;2日目の主菜の料理ID
(int x_22 1 5) ;2日目の副菜の料理ID
(int x_23 1 7) ;2日目のデザート of 料理ID
(int x_24 1 3) ;2日目の主食の料理ID

(int x_31 1 5) ;3日目の主菜の料理ID
(int x_32 1 5) ;3日目の副菜の料理ID
(int x_33 1 7) ;3日目のデザート of 料理ID
(int x_34 1 3) ;3日目の主食の料理ID
```

図 9 Sugar のプログラムの例 1

```
(>= (+ x_11_kcal x_12_kcal x_13_kcal x_14_kcal) 1800) ...①
(>= (+ x_11_pro x_12_pro x_13_pro x_14_pro) 70)
(<= (+ x_11_fat x_12_fat x_13_fat x_14_fat) 55) ...②
(>= (+ x_11_fat x_12_fat x_13_fat x_14_fat) 50) ...②
```

図 10 Sugar のプログラムの例 2

```
(alldifferent x_11 x_21 x_31) ...③
(alldifferent x_12 x_22 x_32) ...④
(alldifferent x_13 x_23 x_33) ...⑤
```

図 11 Sugar のプログラムの例 3

第5章 考察

前章において、献立推薦のシステムの実装の構成を説明した。実際に実装する前に `sample` ファイルを仮に作成し簡単にすることでプログラムの実装のエラーを早めに対処できるようにした。この `sample` ファイルは、料理名やその料理の栄養価も架空で作成し、2日間の献立推薦を行なったのを図 12 で示す。

しかし、実際に文部科学省の日本食品標準成分表を今回の栄養価データベースとして実装を行なったところ、空集合が値として返ってきた。これは、アスリート食の栄養価を満たす料理がないということが考えられる。これは、基準の食品 100g や主菜、副菜の固形物 1 個の場合であったためである。実際にアスリート食を食べる際は、主菜、副菜の固形物 2 個提供されることがあるため、図 13 のように栄養価データベースの主菜、副菜、主食を基準の 2 倍に設定することにした。デザートは、1 つしか提供されないため基準値とすることとする。また、ご飯は元々基準値である 600g で設定しているため変更しない。基準の 2 倍にして実行したが、これも空集合が値として返ってきた。この結果から栄養価のデータベースの料理数が少ないためであると考えられる。文部科学省の食品成分表標準成分表では今回の制約を満たす料理は無い。そのため文部科学省のデータベースでなく、より料理数があるデータベースを使うことでアスリート食に適した献立推薦を行えると考える。

```
1日目の main はハンバーグです。
1日目の sub はコロッケです。
1日目の rice はカレーです。
1日目の dessert はみかんゼリーです。
2日目の main はチキン南蛮です。
2日目の sub はカニクリームコロッケです。
2日目の rice は白飯です。
2日目の dessert はプリンです。
```

図 12 `sample` ファイルで実行した際の献立推薦

料理名,ID,タイプ,カロリー,タンパク質,脂質,炭水化物	料理,ID,タイプ,カロリー,タンパク質,脂質,炭水化物
うどん,1,rice,105,2.6,0.4,21.6	うどん,1,rice,210,5.2,0.8,43.2
そうめん,2,rice,127,3.5,0.4,25.8	そうめん,2,rice,254,7.0,0.8,51.6
中華めん,3,rice,149,4.9,0.6,29.2	中華めん,3,rice,298,9.8,1.2,58.4
スパゲッティ,4,rice,165,5.4,0.9,32.0	スパゲッティ,4,rice,330,10.8,1.8,128.0
生パスタ,5,rice,247,7.8,1.9,46.9	生パスタ,5,rice,494,15.6,3.8,93.8
こめ,6,rice,983,16.4,2.7,213	こめ,6,rice,983,16.4,2.7,213
そば,7,rice,132,4.8,1.0,26.0	そば,7,rice,264,9.8,2.0,52.0
がんもどき,8,sub,228,15.3,17.8,1.6	がんもどき,8,sub,456,30.6,35.6,3.2
糸引き納豆,9,sub,200,16.5,10.0,12.1	糸引き納豆,9,sub,400,33.0,20.0,24.2
挽きわり納豆,10,sub,194,16.6,10.0,10.5	挽きわり納豆,10,sub,388,33.2,20.0,21.0
トマト,11,sub,19,0.7,0.1,4.7	トマト,11,sub,38,1.4,0.2,9.4
いちご,12,dessert,34,0.9,0.1,8.5	いちご,12,dessert,34,0.9,0.1,8.5
いちじく,13,dessert,54,0.6,0.1,14.3	いちじく,13,dessert,54,0.6,0.1,14.3
かき,14,dessert,60,0.4,0.2,15.9	かき,14,dessert,60,0.4,0.2,15.9
オレンジ,15,dessert,46,0.9,0.1,11.8	オレンジ,15,dessert,46,0.9,0.1,11.8
グレープフルーツ,16,dessert,38,0.9,0.1,9.6	グレープフルーツ,16,dessert,38,0.9,0.1,9.6
パイナップル,17,dessert,51,0.6,0.1,13.4	パイナップル,17,dessert,51,0.6,0.1,13.4
ハスカップ,18,dessert,53,0.7,0.6,12.8	ハスカップ,18,dessert,53,0.7,0.6,12.8
パッションフルーツ,19,dessert,64,0.8,0.4,16.2	パッションフルーツ,19,dessert,64,0.8,0.4,16.2
ぶどう,20,dessert,59,0.4,0.1,15.7	ぶどう,20,dessert,59,0.4,0.1,15.7
ブルーベリー,21,dessert,49,0.5,0.1,12.9	ブルーベリー,21,dessert,49,0.5,0.1,12.9
もも,22,dessert,40,0.6,0.1,10.2	もも,22,dessert,40,0.6,0.1,10.2
りんご 皮むき,23,dessert,57,0.1,0.2,15.5	りんご 皮むき,23,dessert,57,0.1,0.2,15.5
りんご 皮つき,24,dessert,61,0.2,0.3,16.2	りんご 皮つき,24,dessert,61,0.2,0.3,16.2
べにざけ,25,main,177,28.5,6.0,0.1	べにざけ,25,main,354,51.6,12.0,0.2
さんま,26,main,270,23.9,17.9,0.2	さんま,26,main,540,47.8,35.8,0.4
ぶり,27,main,304,26.2,20.4,0.3	ぶり,27,main,608,52.4,40.8,0.6
ベーコン,28,main,405,12.9,39.1,0.3	ベーコン,28,main,810,25.8,78.2,0.6
ウインナーソーセージ,29,main,321,13.2,28.5,3.0	ウインナーソーセージ,29,main,321,13.2,28.5,3.0

図 13 左が基準値,右が2倍にした基準値

また、これからの大学アスリート食の献立推薦の機能拡張が様々考えられる。まず、材料の仕入れコストを考慮した献立推薦が考えられる。今回の献立推薦は、仕入れ時のコストを考慮していない。そのため献立推薦されてもコストがかかれば負担が増える。今のアスリート食は 850 円とされているため、それを超える可能性は大いにある。よって、同じ栄養素でコストが少ない業務用材料などを仕入れることでコストが少ない材料で栄養価も担保できる献立推薦することができると考える。

さらに、アスリート食の残飯量を考慮した献立推薦も可能であると考えられる。食べ残しをする部員にも好き嫌いやその日の体調で変動することはある。そのため、もし食べ残しをした場合、その捨てられた残飯量を計算する。その残飯量で

どの料理が人気でどの料理が不人気なのかがわかる。それを次の献立推薦に活かすようなフィードバックを繰り返し行うことで、人気な料理を知ることができ、その料理に類似した料理を推薦することで料理が被ることがなく、人気料理を推薦することができる。それだけでなく残飯量も減らすことができる献立推薦を提案することができるため無駄なコストなども減ると考える。

第6章 おわりに

献立推薦が今後さらに発展していくためには、材料の仕入れコストの考慮や残飯量のフィードバックを踏まえた献立推薦が必要である。個人に焦点を当てた献立推薦は数多くあるが、同じ条件で大人数に対する体育会などのスポーツに対する研究は少ないため様々な大学などの問題を聞き、本研究の対象となった立命館大学だけでなく全国の大学に対応できるようになれば日本のスポーツの強化にもつながると考える。本研究では、アスリート食のメニューが被ることなく、また身体作りに必要な栄養価も担保する献立推薦の提案を目的とし、アプローチを試みた。

献立推薦の制約充足問題としての定式化

制約充足問題として定式化するために、各日の主菜、副菜、デザートを変数として、主菜、副菜、デザートに分類される料理をドメインとして、そして各栄養価の必要摂取量および料理の重複確認を制約として定義した。このように献立推薦を制約充足問題として定式化することで、今後もコストや在庫などの制約を追加することが可能である。

献立の連続性の考慮

前週の一週間と被らないことによりメニューの種類が増え、選手たちの食事に対する飽きというのが無くなった。また、できれば前週の食べ残し量も考慮できるような拡張が可能である。さらに、栄養士の献立と提案手法の献立を部員に比較し評価してもらい、栄養素などはきちんと出来ているがまだ実用性はないと評価された。

これまでは、献立の連続性の考慮や制約による栄養価の担保について述べてきたが、今後は、栄養士に実際に評価をしてもらいその評価や問題点を献立推薦システムに組み込む。それらのフィードバックを繰り返すことでこの献立推薦システムを使用しアスリート食を推薦してもらうことで、体育会のさらなる強化に繋がる可能性があると考え。また、栄養価のデータの追加が必要不可欠である。これらのデータから推薦された材料から新しい料理の開発などもできれば同じ材料でも違う料理ができ、同じ材料だが違う料理として飽きなども無くなり、コストも考慮しながらより栄養価の担保が容易になると考えられる。さらに実際に選手らにアンケートを取ることで栄養士が考える人気メニューの齟齬があるかもしれないためアンケートも必要であると考え。献立推薦

の開発に取りかかるとともに、コストや残飯量などを考慮するシステムを開発することを検討することを考えている。

謝辞

本研究を行うにあたり，熱心なご指導，ご助言を賜りました村上陽平准教授に深謝申し上げます。また，普段からお世話になっている社会知能研究室の皆様
に心より感謝申し上げます。

参考文献

- [1] 長村玲奈, 波多野賢治. (2011). 個人のコンディションや欲求を考慮した献立推薦システムの実装とその評価. 第 73 回全国大会講演論文集, 2011 No. 1, pp. 623-624.
- [2] 加藤哲子, 新谷恵子. (2005). 病院における献立作成と調理作業の効率化に関する研究. 山形県立米沢女子短期大学紀要, Vol. 40, pp. 79-86.
- [3] 田村直之, 丹生智也, 番原睦則. (2010). 制約最適化問題と SAT 符号化. 人工知能学会誌, Vo. 125, No. 1, pp. 77-85.
- [4] 井上克巳, 田村直之. (2010). SAT ソルバーの基礎 (〈特集〉最近の SAT 技術の発展). 人工知能学会誌, Vol. 25, No. 1, 57-67.
- [5] 田村直之, 丹生智也, 番原睦則. (2010). SAT 変換に基づく制約ソルバーとその性能評価. コンピュータ ソフトウェア, Vol. 27, No. 4, 4_183-4_196.