Vol.2021-ICS-201 No.2
2021/3/9

料理レシピの構造化による家庭料理の食品ロス削減に向けた献立作成手法の提案

小林 直也[1,a)]　横山 想一郎[2]　山下 倫央[2]　川村 秀憲[2]　関屋 英理子[3]　杉浦 仁志[4]

概要：家庭での調理は外食や弁当の購入に比べ健康面，経済面でメリットがあるが，一方で材料の買い出しや調理の時間を要するため，家庭での調理が難しい場合がある．そこで本研究では調理の効率化を目指し，1 週間で購入した材料をちょうど使い切れる献立の提案と，調理手順のスケジューリングの実験を行った．またこれらの計算を行うため，既存のレシピを参考に木構造を基本形とする「構造化レシピ」を作成した．この構造化レシピを用いて最適化問題を解き，さらに材料の分量変更を行うことで，材料の余りがない 1 週間分の献立を作成した．またこの 1 週間の主菜，副菜の組み合わせの中から，1 週間全体での調理時間が最短となる組み合わせを導出した．

1. 序論

1.1 研究目的

日々の食事は最も基本的な生活の一部である．現代ではスーパーやコンビニ，出前サービス等を介して商品を購入するなど食事の行う方法が多様化しているが，家庭料理は依然として多くの家庭における食事の中心となっている．家庭料理は外食と比べ，栄養バランスの偏りが少なく，コストが抑えられることが一般的に知られている．このようなメリットがある一方で，家庭料理には献立の決定や材料の買い出し，実際の調理時間に多くの時間が掛かる．特に現代では一般的になっている夫婦共働き世帯では家庭料理を行うことの負担が大きい．そこで本研究では家庭料理の調理の効率化を目的とし，構造化レシピを用いて，ちょうど 1 週間で材料を使い切れる献立の選択，材料の分量変更，時間的に最も効率の良い主菜と副菜の組み合わせの選択を行った．

1.2 研究背景

1.2.1 家庭料理を効率的に行うことの難しさ

家庭料理を効率的に行う際に，問題となる要素に以下の 3 点が挙げられる．

(1) 献立の決定

(2) 材料の買い出し

(3) 調理時間

(1) 献立の決定に関して，現在，クックパッドのようなレシピ共有 SNS 等には，プロやアマチュアの調理者が投稿した膨大な数のレシピがあるため，レシピを入手できないことを懸念する必要はない．しかし，無数に存在するレシピの中から調理するレシピを無作為に選択した場合，購入した材料を問題なく使用できる期間に材料を使い切れない場合が考えられる．使用期限の切れた食材の破棄はフードロスとして社会問題となっているほか，経済面からみても効率的でない．そのため，大量のレシピの中から一定の期間のうちに材料を使い切れるようレシピの組み合わせを選択する必要がある．

(2) 材料の買い出しには時間を要する．買い出しに掛かる時間は，買い出しに行く頻度を減少させることで大きく削減できると考えられる．買い出しに行く頻度を減少させるには，長期間の献立を事前に決定し，一度の買い出しで調理に必要な材料を揃えることが必要となる．しかし，長期間の献立を計画する際に献立の選択を誤れば，必要とされる期間内に使い切れない材料が発生する．材料をちょうど使い切ることを想定したレシピを選択することは容易ではない．

(3) レシピの選択の際には，そのレシピの調理時間が重要な選択基準となる．家庭料理の調理では，レシピを 1 つ 1 つ順番に作るのではなく，並列できる調理は可能な限り並列して同時に調理する．単品のレシピの調理時間の目安が与えられても，大量に存在するレシピから，どのレシピ同士の組み合わせが最も調理時間を短縮できるのか素早く

[1] 北海道大学 工学部
[2] 北海道大学 大学院情報科学研究院
[3] 株式会社ニチレイ 技術戦略企画部
[4] 株式会社 ONODERA GROUP ホールディングス Executive Chef

[a)] kobayashi_n@ist.hokudai.ac.jp

ⓒ 2021 Information Processing Society of Japan

判断することは困難である．ここで，最も効率的な調理手順で複数のレシピを作る際の調理時間を計算するツールの需要が生じる．

1.2.2 日本語レシピを扱うことの困難さ

コンピュータを用いて料理レシピを作成する研究は，最新のトピックとして盛んに行われている．これらの研究は多くの場合，Recipe1M+のような英語で書かれた内容が整理されたレシピデータを利用している [1]．これらのレシピは，研究を目的としないネット上に投稿される一般的なレシピと比べデータとして扱いやすいが，日本料理に関するレシピであっても，日本人が想定する一般的な献立とは内容が異なっている．そこで，日本人の一般的な感覚が反映された日本語のレシピを扱う必要があるが，そこには以下のような難点が存在する．

(1) 自然言語処理が英語ほど発達していない
(2) 口語的でくだけた表現が多く，新聞記事のような表現より扱いづらい
(3) 「玉ねぎ」「タマネギ」といった材料名の表記ブレ
(4) 「少々」「お好み」などあいまいな表現
(5) 「★の材料を炒める」など独自の記号を用いた説明

元より，レシピは料理の画像と，複数の調理手順を含む説明文である．調理を加えることで材料の名称は変化し，見た目も大きく変化したり，姿が見えなくなったりする．このような要素を含む料理レシピをコンピュータで扱うことは，非常に困難なタスクである．

2. 関連研究

ここでは，料理のレシピに関する研究を紹介する．

2.1 コンピュータビジョン分野

料理レシピを扱うコンピュータビジョンの分野では，料理レシピのテキストと画像の両方を取り扱うクロスモーダルなレシピ処理に関する研究が行われている．J. Marn らは，料理レシピのテキストと，対応する料理の画像をそれぞれ Word2Vec と ResNet50 を用いてベクトル化し，対応するもの同士のベクトル表現を近似させる Joint Embedding を行って，レシピのテキストと画像の相互検索を行っている [1]．また J. Marn らは，レシピデータとして作成し利用した Recipe1M+を公開しており，これはコンピュータで料理を扱うさまざまな分野で活用されている．A. Salvador らは，料理の画像を入力し，それを元にその料理の材料と調理手順を記したレシピをテキスト形式で生成し出力する Inverse Cooking を作成し公開している [2]．A. Salvador らの研究でも，料理画像をエンコーダーを用いてベクトル化しており，推定した材料に対してもベクトル化を行っている．エンコーダー，デコーダーには自然言語処理を得意とする RNN の一種である Transformer が用いられている．

2.2 料理の分析に関する分野

Simas らは，材料に含まれる香り成分を調査し，香り成分の共通度が高い材料同士をエッジで結ぶことで材料群をグラフ構造化したフレーバーネットワークを作成している [3]．このフレーバーネットワークを元にして，欧米では共通の香り成分を持つ材料同士の組み合わせが好まれるというフードペアリングについて調査している．また，本研究に類似した研究として，自然言語で書かれたウェブ上のレシピテキストから，主要な調理を自動的にワークフロー化して抽出するというものがある [4]．この研究でレシピテキストから抽出されたワークフローは，本研究と類似した木構造で表現される．本研究では構造化レシピ作成の自動化には焦点を当てず，手動で構造化レシピを作成した上で，構造化レシピの活用法を探ることに焦点を当てた．

2.3 レシピの生成に関する研究

レシピの生成は，レシピデータを元に，データとして存在しないレシピを作るもので，Inverse Cooking のように料理画像に対応するレシピを生成するものと，まだ誰も見たことのないユニークなレシピを開発するものに大別される．特に人工知能 Watson に料理を学習させた Chef Watson は，豚肉を乗せたアップルパイなど，奇抜なレシピを生み出してシェフを驚かせた [5]．また料理レシピのテキストや画像をベクトル化する以外の手段でレシピの生成を扱う研究には，独自の生成文法に従って目新しいカクテルのレシピの生成を行うものがある [6]．

3. レシピの構造化

本研究では，料理レシピを木構造を基礎とする構造化レシピで表現する．自然言語で書かれたレシピを自然言語処理などの技術を用いてコンピュータが扱える表現にするのではなく，構造化レシピの利用に焦点を当てるために，初めからコンピュータで扱いやすい形式のレシピを手動で作成する．そのため構造化レシピには，通常のレシピには含まれていない情報を付与し，レシピの記述における曖昧さをなくすための制約を設ける．

3.1 レシピの基本構成

料理を作る過程は，材料に 1 つの調理 (ここでは「切る」「炒める」などの 1 工程) を行い，元の材料とは性質の異なる中間材料を作るという単位に分割できる．料理の原材料は調理→中間材料→調理→中間材料→……という過程を繰り返し，最終的に完成に至る．この過程を文章化したものが料理レシピである．各調理には一定の調理時間を要する．また，調理には調理者を含む所定の調理資源を使用する必要がある．

実際のレシピに書かれた調理には，一部並列して進行可能な調理が存在する．また，一つの調理が複数の (中間) 材

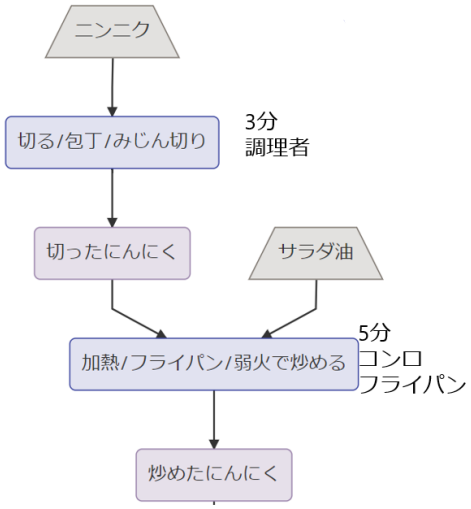


図 1 構造化レシピの例

料を要する場合があり，これは「にんじんとじゃがいもを炒める」というような調理のことである．そのためレシピの表現には，一直線に示した調理過程よりも木構造で示した調理過程が適している．木構造で表した構造化レシピの一部を図 1 に示す．

本研究で扱う構造化レシピでは，木構造によって調理手順が完全に記述されている．木構造においてより後の調理は，前の調理が完了するまで開始できない．別の言い方をすれば，全ての調理は，その調理に必要となる中間材料が揃わない限り開始できない．また木構造において前後の関係にない調理は，必要な調理者や調理資源が揃えば，常に並列に調理可能である．よって，木構造を用いるで調理手順の条件を完全に記述できる．

3.2 構造化レシピ作成例

クックパッドに掲載されている「ブロッコリーとベーコンのにんにく炒め」を参考に，構造化レシピを作成した．構造化レシピを作成するために必須の情報でレシピ中に記載がないものや曖昧なもの，特に調理の所要時間は，筆者が推定して決定した．元レシピと構造化後のレシピを図 2，図 3 に示す．構造化レシピは，材料を灰色の台形で，調理を青の長方形で，中間材料を紫の長方形で表した．また，構造化レシピの実体は，木構造の構成に加え，材料の分量や各調理の所要時間の情報を持った json 形式のデータである．

4. 構造化レシピ作成支援アプリ

自然言語で書かれた通常のレシピから構造化レシピを機械的に作成することは非常に困難であり，人間が直接作成する方が容易である．しかし手動で構造化レシピを作成する場合は，表記ブレや必要情報の欠落などのミスが頻繁に発生する．また，木構造で表現される構造化レシピを逐一



図 2 ブロッコリーとベーコンのにんにく炒めの元レシピ
https://cookpad.com/recipe/2738643

可視化しながら作成することは負担となり，可視化せずに木構造を想像しながら作成するのも困難である．さらに，構造化レシピには既存のレシピには明記されない情報が含まれる必要があるため，レシピの考案者本人が構造化を行うのが理想的である．そこで本研究では構造化レシピ作成支援アプリを開発し，構造化レシピ作成の容易化，ヒューマンエラーの防止と構造化レシピの可視化を行った．

4.1 材料名の表記統一

料理レシピの材料名の表記が一通りに定まらないという問題が存在する．例えば，玉ねぎには「たまねぎ」「タマネギ」などの表記が存在し，どれを使用するべきかについて明確な基準はない．このような表記ブレによって，同一の材料を認識できなくなるといった問題は，テーブルによる変換や簡単な機械学習などの手段で解決できると考えられるが，表記ブレが発生しないように名称を制限することで容易に解決できる．そのため構造化レシピ作成支援アプリでは，材料名は全て選択式とし自由な表記をさせないことで，表記ブレの問題を解決した．

ⓒ 2021 Information Processing Society of Japan

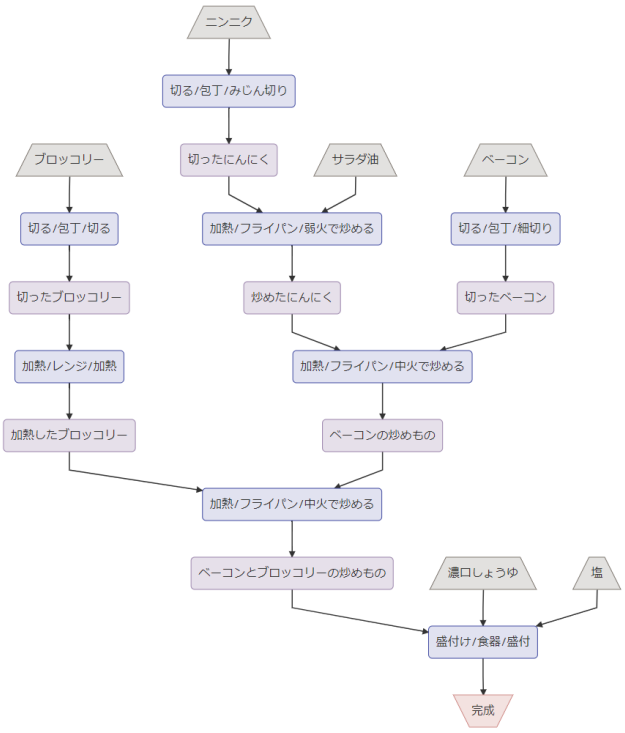


図 3 ブロッコリーとベーコンのにんにく炒めの構造化レシピ

4.2 調理名の表記統一

調理は通常文章表現で記載されるため，調理内容の識別，同定は材料名の識別，同定より格段に難しいタスクになると考えられる．このため構造化レシピ作成支援アプリでは調理の種類もあらかじめ定められた調理の中から選択する方式とした．各調理は必要な調理資源の情報と関連付ける．例えば「加熱/フライパン/炒める」という調理には，フライパンとコンロが必要な調理資源として紐付ける．また，今回作成した構造化レシピでは調理を行うのに必要な要素として，調理者も調理資源に含める．

4.3 材料の分量の情報

クックパッドなどに掲載されているレシピの材料と分量の一覧には，「適量」といった曖昧な表現が頻繁に見られる．今回行う 1 週間で材料の余りが最小となる献立の提案を行う際には，レシピ中の材料の分量が数値で明記されている必要がある．人間には感覚的に理解できる「適量」を機械的に推定することは困難で，不確実である．今回は材料の分量は整数または小数とし，単位を選択式にした．単位を選択式にすることで，全ての材料をグラム換算し材料の余りの計算や分量変更の計算に利用可能にした．

4.4 調理の所要時間

通常のレシピでは，鍋やレンジで加熱する時間の記載やレシピ全体の調理時間の記載はあっても，「大根を切る」「材料を混ぜ合わせる」などの全ての調理に所要時間が記載されていることはない．また，加熱時間についても，「じゃがいもに串が通るまで」など数値以外の条件が記載されている場合がある．しかし，レシピにスケジューリングを適用して，複数のレシピを同時に調理するのに必要な時間を算出し，複数のレシピの最短の調理手順を示すには，全ての調理について所要時間が明記されている必要がある．そのため，構造化レシピ作成支援アプリでは調理に分単位の時間の情報を付加することを必須とした．

5. レシピにおけるスケジューリング

複数のレシピを最も効率の良い手順で調理した場合の最短の調理時間を計算するために，本研究では料理レシピにスケジューリングを適用した．調理におけるスケジューリング問題は，ジョブショップスケジューリングの応用と考えることができる [7]．同時に調理する各レシピはジョブショップスケジューリングのジョブに，「にんじんを切る」「豚肉を焼く」などの各レシピの調理はタスクに相当する．調理者，コンロ，鍋などの，各調理の実行に必要となる調理資源は，ジョブショップスケジューリングにおける機械 (資源) である．タスクに相当する各調理には，その調理を行うのに必要となる調理資源と完了までに必要な時間，開始可能になる条件の情報が付加されている必要がある．レシピのスケジューリングとジョブショップスケジューリングの類似点として，以下のような点が挙げられる．

- 調理者，調理資源に複数ある調理を割り振る
- 各調理には完了するのに一定の時間が必要である
- 各調理を行う順番には制約がある
- 最短時間で全ての調理を完了することを目指す

一方で，以下のような点は典型的なジョブショップスケジューリングとは異なる．

(1) 単一のレシピ内にも並列作業可能な調理が存在する
(2) 調理に必要な器具は複数存在する
(3) 同じ調理が可能な器具は複数存在する

(1) については，「ハンバーグを焼く間にソースを作る」というような調理が例として挙げられる．典型的なジョブショップスケジューリングでは単一のジョブ内のタスクには一通りの開始順番が存在するため並列可能なタスクは存在しないが，レシピのスケジューリングでは単品のレシピにもスケジューリングを適用して最短時間を導出する必要がある．(2) については，例えば「野菜を炒める」という調理は「フライパン」「コンロ」の二種類の調理資源を専有する．この二種類の器具が同時に使用可能な状態になければ，この調理を開始することはできない．(3) は，例えば「野菜を炒める」という調理を行うのに，コンロ A とコンロ B のいずれかが使用可能であれば良いということが挙げられる．これは，あるタスクを担当できる機械はただ一つに定められている典型的なジョブショップスケジューリングと異なる．コンロ A，コンロ B のどちらを使用するかの決定によって調理完了までの最短時間が変化するので，どち

ⓒ 2021 Information Processing Society of Japan

らの器具を使用するかもスケジューリングの対象となる．

6. 実験

初めに，今回使用する 30 の構造化レシピデータの概観を述べる．次に構造化レシピが家庭料理における調理の効率化に寄与するかを確認するため，以下の 3 つの実験を行った．

実験 1. 材料の余りが最小となる 14 品のレシピの選択

実験 2. 材料の余りをゼロにするためのレシピの分量変更

実験 3. 調理時間が最短となる主菜と副菜の組み合わせの選択

ここで家庭料理の効率化とは，材料の買い出しの効率化と調理時間の効率化の 2 点を指す．実験 1 と 2 は買い出しの効率化に関して，実験 3 は調理時間の計算に関して構造化レシピの有用性を調べる実験である．

6.1 構造化レシピデータ

実験では，既存のレシピを参考にして構造化レシピを 30 品作成した．実験で行う 1 週間分の献立提案では主菜・副菜のグループの中から 7 品ずつ選択するので，この 30 品を筆者の主観で主菜 15 品と副菜 15 品に分類した．実験に先立ち，この 30 品の構造化レシピの基本情報を表 1，表 2，表 3 に示す．

表 1 主菜候補

レシピ名	最短調理時間 (分)
あんかけうどん	32
とんかつ	34
クリームシチュー	55
グラタン	19
コロッケ	34
ビーフシチュー	56
ポテトマカロニグラタン	57
マーボー春雨	37
ロールキャベツ	30
回鍋肉	46
油淋鶏	30
焼きめし	14
生姜焼き	36
酢豚	42
野菜餃子	44

6.2 実験 1 材料の余りが最小となる 14 品のレシピの選択

週の初めに 7 日間の献立を決定することで，買い物に行く回数を減らすことができ効率的である．しかし，1 週間分の材料を一度に購入したとき，1 週間後に最初に決定した献立をつくり終えたときに，使い切れず無駄になってしまう材料が大量にある場合，経済的な観点から効率的とは言えない．ここで，材料は献立に使う分量をちょうど購入することができない場合があることに注意が必要になる．店舗で購入できる材料は，例えばにんじんなら一本単位，豚肉なら 200g 単位など，購入できる最小単位が決まっている．そのため，各材料について (店舗で購入できる分量) × (材料が不足しない最小の自然数)-(献立で使用する分量) が 1 週間後に余る分量になる．実験 1 では，1 週間の献立で使用する全ての材料について，1 週間後に余る材料を最小にするようなレシピの選択を行った．発生する材料の余りは定式化して，Google OR-Tools の CP-SAT solver を使用し最小化した [7].

表 2 副菜候補

レシピ名	最短調理時間 (分)
きゅうりの中華漬け	42
こんこんてりてり	35
なすのなめろう	24
エビチリ	18
スコッチエッグ	35
タコのカルパッチョ	23
ハムエッグ	8
ピーマンの肉詰め	39
ブロッコリーの炒めもの	31
卵とエビの春巻き	29
大根とベーコンの炒め物	35
治部煮	43
肉じゃが	48
長芋と鶏肉の酢豚風	18
鯖の南蛮漬け	19

表 3 構造化レシピの基本情報

主菜最短調理時間平均	37.7 分
副菜最短調理時間平均	29.8 分
調理ステップ数平均	8.93
材料数平均	4.03
材料質量平均	632g

6.2.1 実験設定

以下のように実験を設定した．

- 主菜候補 15 品，副菜候補 15 品から 7 品ずつ選択する
- 同じレシピは 2 回以上選択しない
- 材料の余りはグラム換算で計算する
- 店舗で購入できる分量は材料の種類と 1:1 対応とした
- 食材の購入前には食材を全く所持していないこととする
- 調味料や牛乳，米などは考慮すべき材料から予め除いた

6.2.2 定式化

1 週間後に発生する材料の余り L を最小化する最適化問題の定式化を行った．献立 **b** は，全レシピデータの長さを持つリストで，b_i は 0 か 1 の値を取る．これが i 番目のレシピを献立として採用するかしないかを意味している．

$$\mathbf{b} = [b_0, b_1, ..., b_R] \tag{1}$$

R: レシピの種類数
レシピ i を献立に含める場合: $b_i = 1$
レシピ i を献立に含めない場合:$b_i = 0$

全構造化レシピに含まれる材料の種類数の長さを持つ材

料リストを $\mathbf{m}_i$ とする．$\mathbf{m}_i$ は，i 番目のレシピが使用する材料の質量のリストである．

$$\mathbf{m}_i = [m_{i,0}, m_{i,1}, ..., m_{i_M}] \tag{2}$$

M: データで使用した全材料の種類数
$m_{i,j}$: レシピ i に必要な材料 j の量 [g](0 を含む)

$\mathbf{s}$ は店舗で購入できる材料の最小単位の質量を表すリストで，m_i と同じ長さである．

$$\mathbf{s} = [s_0, s_i, ..., s_M] \tag{3}$$

s_j: 材料 j の販売単位 [g]

これらを用いて，材料の余り L を以下のように定義する．材料の余り L は，それぞれの材料を購入できる単位の自然数倍から，献立として採用したレシピで使用する材料の分量を引いたものを，全ての材料について総和をとったものである．

$$L = \sum_{j=0}^{M}(s_j n_j - \sum_{i=0}^{R} b_i m_{i,j}) \tag{4}$$

$s_j n_j$: 購入する材料 j の量 [g]
ただし，n_j は

$$s_j n_j - \sum_{j} b_i m_{i,j} \geq 0 \tag{5}$$

を満たす任意の自然数とする．このとき，$\sum_i b_i m_{m,j}$ は献立 b で使用する材料 j の量 [g] である．

ここから，L を最小化する n_j, b_i の最適化を行う．

6.2.3 結果と考察

CP-SAT を用いて，0.84 秒で最適解が得られた．選択された主菜を表 4 に，副菜を表 5 に示す．また，今回選択した献立で発生する材料の余りは表 6 のようになった．また，比較対象としてランダムに 50 回献立の選択を行い，その結果を表 7 に示した．ランダムなレシピを選択した結果の材料の余りの平均と比較して，今回得られた 1 週間分の主菜・副菜では材料の余りを約 4 分の 1 程度に抑えられた．

表 4 選択された主菜

主菜
とんかつ
ビーフシチュー
マーボー春雨
ロールキャベツ
回鍋肉
焼きめし
生姜焼き

材料の余りを最小化した場合でも，今回の主菜・副菜の選択では 1 週間で 576g の材料の余りが発生した．この理由は，今回使用した構造化レシピの数が少なかったため，全てのレシピ中に一度しか登場しない材料が多数あったためであると考える．

表 5 選択された副菜

副菜
きゅうりの中華漬け
なすのなめろう
エビチリ
ハムエッグ
ブロッコリーの炒めもの
卵とエビの春巻き
鯖の南蛮漬け

表 6 材料の余り

材料名	余る材料と分量 (g)
にんじん	65
キャベツ	50
ミニトマト	80
ベーコン	50
ロースハム	51
きくらげ	30
食パン	100
豚バラ肉	150
合計	576

表 7 ランダムな選択結果

	余る材料の分量 (g)
平均値	2440.7
最小値	1173
最大値	3594.25

6.3 実験 2 材料の余りをゼロにするためのレシピの分量変更

実験 1 の結果から，材料の余りが最小になるようなレシピの選択を行っても，今回行った献立の選択では 1 週間で 576g の材料の余りが発生することが分かった．この材料の余りをすべて使い切るために，各レシピにその材料が使用されている割合に応じて余った材料を割り振り，レシピ中の材料の分量を増加させる実験を行った．

6.3.1 実験設定

今回のアルゴリズムを用いて厳密な計算を行えば，余った材料を全て使い切ることが可能である．ただし，厳密な計算結果は無限小数を含むことがあるため，変更後の材料の分量が桁数の非常に多い小数となる場合がある．これでは変更後の材料の分量が実践可能なものとならないため，今回は材料を増加させることができる単位に以下のような条件を設定した．

- レシピ中で個数表記 (本，枚など含む) である分量は 0.1 個単位で増量

- グラムで表記されている分量は 1g 単位で増量
- 上の条件を満たすため小数を切り捨て

これら条件により，計算結果はわずかな誤差を含むものとなる．ただし，誤差は各材料につき 1g または 0.1 個未満となるため，実際の調理ではその材料を使う最終日に全て使い切ることによって，材料の余りはゼロとなる．

6.3.2 結果と考察

分量変更の結果を表 8 に示す．

分量の変更について，この結果が実用上の問題がある箇所を赤で示した．問題点には以下の二種類がある．

(1) 変更前との分量の差が大きすぎる
(2) 小数での分量表現が非実用的である

(1) の原因には，今回の実験での構造化レシピデータ数が少なく，データ中の一品のみに使用される材料の割合が高くなったことが考えられる．今回のアルゴリズムでは，提案された献立の一品にしか使用されていない材料が大量に余った場合，その全てを一品に割り振るため，このような問題が発生した．(2) の原因には，構造化レシピデータ中で個数単位で分量が表現されている材料については，0.1 個単位の増量を許容したことが考えられる．ただし，「にんじん 0.5 本」といったように小数を用いた分量表現は通常のレシピにも頻出するので，今回は 0.1 個単位での増量を採用した．この問題の解決策としては，材料ごとに増量可能な単位を決定するテーブルを作ることを想定している．

6.4 実験 3 調理時間が最短となる主菜と副菜の組み合わせの選択

実験 1 で選択された主菜 7 品と副菜 7 品について，1 週間全体での調理時間が最小化できる主菜と副菜の組み合わせを選択した．はじめに主菜と副菜の組み合わせ計 49 通りについて，それらのレシピを同時に調理した場合の最短の調理時間をスケジューリングにより求めた．その結果を元に，1 週間での主菜と副菜の組み合わせ 7! 通りの中から，1 週間全体での調理時間が最小となるような主菜と副菜の組み合わせを導出した．またこの結果を比較するために，7! 通り全ての組み合わせ方についても計算を行った．レシピを同時に調理する際の最小時間は，Google OR-Tools の CP-SAT solver を用いてスケジューリング問題を解くことで導出した．

6.4.1 実験設定

スケジューリングを行うにあたって，調理環境は以下のように設定した．

- 調理者 1 人
- フライパン 1 個
- コンロ 2 台
- 保存容器 1 個
- ボール 1 個

また，この実験では主菜と副菜の組み合わせ方に注目して，どの組み合わせを何日目に作るかは考慮しない．

6.4.2 結果と考察

1 週間での総調理時間が最小化できる主菜と副菜の組み合わせは表 9 のようになった．これを 7! 通り全ての主菜・副菜の組み合わせ方について計算した表 10 と比較すると，1 週間で約 30 分の調理時間短縮を行えていることが確認できる．また，本論文の執筆時点では構造化レシピデータに含まれる材料名や調理の文字列から，最短の調理手順を説明するレシピ文章を生成するアルゴリズムを作るまでに至らなかったが，複数のレシピを同時調理する際の最短の調理手順もスケジューリング結果から得られる．これは以下のように，調理者と各調理資源に，それぞれが担当する調理と開始時間，終了時間を割り当てたものである．

調理手順

【調理者】
0-3 分 豚バラ肉に片栗粉をまぶす
3-5 分 きゅうりに塩をまぶす
5-8 分 キャベツを洗う
8-10 分 洗ったキャベツを手でちぎる
……
【フライパン】
3-13 分 片栗粉をまぶした豚肉を中火で炒める
13-23 分 柔らかくしたキャベツを中火で炒める
23-33 分 焼いたキャベツ, 甜面醤を入れた豚肉, 長ネギを中火で炒める
……

7. 結論

本研究では初めに構造化レシピを提案し，レシピ構造化支援アプリを用いて構造化レシピデータを作成した．構造化レシピを作成する目的は，日本語レシピを自動的にコンピュータで扱える形式に変換することを回避し，コンピュータで分量の計算やスケジューリングが実施可能なように設計したレシピを利用することで，その活用方法に焦点を当てることであった．

本研究では作成した構造化レシピの有用性を示すため，構造化レシピを家庭料理の効率化に利用する実験を実施した．構造化レシピにおいて統一された材料名と明記された分量から，材料が余りにくい 1 週間の献立提案を行った．これにより，材料を余らせることなくまとめ買いをすることが可能となった．また構造化レシピにより明確になった調理手順と所要時間を利用して，レシピのスケジューリングを行った．その結果，選択された 1 週間の献立の中から最短時間で調理可能な主菜・副菜の組み合わせを算出し，家庭料理の調理時間短縮に利用可能となった．

今後の課題として，この結果をより現実の家庭料理によ

ⓒ 2021 Information Processing Society of Japan

表 8 材料の分量変更

レシピ名	材料名	変更前分量	変更前分量 (g)	変更後分量	変更料分量 (g)
ビーフシチュー	にんじん	1 本	150	1.2 本	180
マーボー春雨	にんじん	10g	10	12g	12
マーボー春雨	きくらげ	70g	70	100g	100
ロールキャベツ	食パン	2 枚	100	4 枚	200
ロールキャベツ	ベーコン	10 枚	200	12 枚	240
回鍋肉	キャベツ	250g	250	263g	263
回鍋肉	豚バラ肉	250g	250	400g	400
焼きめし	ロースハム	5 枚	85	6.6 枚	112
エビチリ	ミニトマト	2 個	20	10 個	100
ハムエッグ	ロースハム	4 枚	68	5.3 枚	90
ブロッコリーの炒めもの	ベーコン	50g	50	60g	60
鯖の南蛮漬け	にんじん	0.5 個	75	0.6 個	90

表 9 選択された主菜・副菜の組と調理時間

主菜	副菜	調理時間の和 (分)	同時調理時間 (分)
とんかつ	鯖の南蛮漬け	53	38
ビーフシチュー	卵とエビの春巻き	95	56
マーボー春雨	ハムエッグ	45	40
ロールキャベツ	ブロッコリーの炒めもの	61	39
回鍋肉	きゅうりの中華漬け	88	46
焼きめし	エビチリ	32	22
生姜焼き	なすのなめろう	60	40

表 10 1 週間の調理時間の平均値・最大値・最小値

	1 週間の調理時間 (分)
平均	311
最大値	345
最小値	281

り近いものとすることが挙げられる．実際に調理者によって行われる厳密な調理時間と，本研究にて作成した構造化レシピにおいて仮定した時間は一致しないことが予想される．このため，より実際の調理者が体験する状況に近いモデルを構築するため，実際の家庭が献立として計画するレシピの数に近い構造化レシピを用意する必要がある．また，複数の調理を並行して行うことが簡易かどうかを実際に調理を行うことで検証することが望ましい．その際は，熟練した調理者と調理の初心者の両者で比較し，得られた結果を参考にするなどの実験を行うことが考えられる．一方で買い出しの効率化に関しては，1 週間の献立で余る材料の量はランダムに献立を選択する場合と比較して 1/4 程度に削減できたことから，本実験の方針で一定の成果が得られた．また調理時間とは異なり，予め用意された材料が途中で変更されることや，調理者の能力による差異が生じず，思わぬ変更が発生しづらいため実際の状況に対応していると考えられる．店舗で購入しやすい分量にバリエーションを追加することで，材料が余りにくい献立も，より実際の家庭料理に近づけることが可能であると考える．

参考文献

[1] Marn, J., Biswas, A., Ofli, F., Hynes, N., Salvador, A., Aytar, Y., Weber, I. and Torralba, A.: Recipe1M+: A Dataset for Learning Cross-Modal Embeddings for Cooking Recipes and Food Images, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 43, No. 1, pp. 187–203 (2021).

[2] Salvador, A., Drozdzal, M., Giro-i-Nieto, X. and Romero, A.: Inverse Cooking: Recipe Generation From Food Images, *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10445–10454 (2019).

[3] Simas, T., Ficek, M., Diaz-Guilera, A., Obrador, P. and Rodriguez, P. R.: Food-Bridging: A New Network Construction to Unveil the Principles of Cooking, *Frontiers in ICT*, Vol. 4, p. 14 (2017).

[4] Yamakata, Y., Imahori, S., Maeta, H. and Mori, S.: A method for extracting major workflow composed of ingredients, tools, and actions from cooking procedural text, *2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 1–6 (2016).

[5] Pinel, F.: What's Cooking with Chef Watson? An Interview with Lav Varshney and James Briscione, *IEEE Pervasive Computing*, Vol. 14, No. 4, pp. 58–62 (2015).

[6] Pagnutti, J. and Whitehead, J.: Generative Mixology: An Engine for Creating Cocktails, *Proceedings of the Sixth International Conference on Computational Creativity (ICCC 2015)*, Park City, Utah, Brigham Young University, Brigham Young University, p. 212–219 (2015).

[7] Developers, G.: Google — About OR-Tools, `https://developers.google.com/optimization/introduction/overview`. (Accessed on 02/02/2021).

ⓒ 2021 Information Processing Society of Japan