

令和 3 年 5 月 29 日現在

機関番号：13901

研究種目：基盤研究(B) (一般)

研究期間：2017～2019

課題番号：17H01721

研究課題名(和文) データベース上の制約プログラミングのためのSQL言語の拡張とその処理系

研究課題名(英文) An extension of SQL language for constraint programming on database and its processing system

研究代表者

酒井 正彦 (Sakai, Masahiko)

名古屋大学・情報学研究科・教授

研究者番号：50215597

交付決定額(研究期間全体)：(直接経費) 8,500,000円

研究成果の概要(和文)：データベース問い合わせ言語SQLが分かるユーザであれば、制約ソルバの知識なしに組合せ最適化問題を容易に記述でき、それを汎用の制約解決ソルバを利用して解くための処理システムCombSQL+を提案・実現し、Webを利用して一般に公開した。その鍵となるアイデアは、関係データベースの理論を集合上の理論に自然に拡張した点にある。これにより、単純な考え方に基づいて問題記述が可能となり、システムの実現が可能になった。

研究成果の学術的意義や社会的意義

関係データベースの問い合わせ言語を、関係データの集合に対して拡張したことで、その有限の表現として制約つきテーブルを提案したという点に学術的意義がある。一方で、実装した組合せ問題解決のためのツールは利用するための敷居が低いため、今後一般向けのユーザインタフェースの構築により、実際の利用が見込まれる。

研究成果の概要(英文)：We have proposed, implemented, and released a system CombSQL+ that solves combinatorial optimization problems by using general purpose constraint solvers according to problem specifications. One of the benefits is that the specifications are easily described by users who know relational database query language SQL without knowledge of constraint solvers. The key idea is the set extension of the theory of relational database, which enables us to design a simple description language as well as its implementation.

研究分野：理論計算機科学

キーワード：組合せ最適化問題 SMTソルバ SQL言語

科研費による研究は、研究者の自覚と責任において実施するものです。そのため、研究の実施や研究成果の公表等については、国の要請等に基づくものではなく、その研究成果に関する見解や責任は、研究者個人に帰属します。

## 様式 C-19、F-19-1、Z-19 (共通)

### 1. 研究開始当初の背景

制約を満たす解を求める問題である組み合わせ問題や、その解のうちで最も優れたものを求める最適化問題は、ネットワーク・LSI・CAD・金融に至るまでありとあらゆる分野での設計・計画・管理・効率化において必要とされ、その効率的な問題解決は非常に重要である。一方、MiniSat に代表される制約の充足可能性判定ソルバ（以下、汎用ソルバと呼ぶ）が近年長足の進化を遂げており、これらの問題の解法としての有効性が認められてきている。

汎用ソルバを利用して問題を解くためには、解きたい問題の制約を汎用ソルバの入力形式（例えば SAT ソルバでは和積標準形の論理式）の充足可能性問題として表現する必要がある。これらは単純な問題であっても膨大な記述量となり、人手で行うことは非現実的であることから、その記述を容易にするための制約記述言語も設計されている。例えば、神戸大学で開発された制約記述言語 Copris は、記述する制約の繰り返し部分を通常のプログラミングで用いられる for 構文などを利用して、プログラムを組む要領で制約記述が可能な優れた言語である。しかしながらそれでも依然として熟練が必要であり、特に汎用ソルバを高速に動作させるために必要な新しい論理変数の導入にはノウハウが必要で、その記述には汎用ソルバの知識と熟練を要する。

これらの理由により、一般ユーザがシステムを利用しようとする専門家には依頼してシステムを構築してもらおうなどの必要が生じ、これらの優れた結果を気軽に利用するには敷居が高いという問題点がある。

### 2. 研究の目的

本研究は、組み合わせ問題を関係データベース上の制約問題として捉えることで問題を容易に記述可能な拡張 SQL 言語を設計すると共に、その問題記述に隠れている構造情報を基とする巧みな変換を与えることで、効率的な処理システムを実現する。これにより、制約ソルバの知識なしに効率良く問題解決するための手法の構築を目的としている。

### 3. 研究の方法

本研究は、種々の検討および試行錯誤の結果、最終的には以下のように研究を進めることが出来た。

(1) 組合せ問題記述のための SQL 言語の拡張: SQL 問い合わせ言語のテーブル（の組）に対する意味論をその集合上に拡張した。その上でテーブルのデータとして制約変数などを導入することで、テーブルの集合を表現する仕組みを導入した。そのうえで最小限の構文要素を追加することで、全ての組合せ問題が表現できる言語拡張を行い、組合せ問題記述言語 CombSQL+ を設計した。具体的に追加した構文は、テーブルからテーブルの集合を生成する CHOOSE 構文、テーブルの集合に名前をつける CREATE SET 構文、集合の内包表現を記述する HAS  $t$  IN 表 SUCH THAT  $P(t)$ 、テーブルの集合から要素を取り出す  $t$  IN 表の集合 MINIMIZING  $Q(t)$  である。ここで、 $P(t)$  および  $Q(t)$  においては  $t$  が通常のテーブルを表す変数で、 $P$  はブール値を  $Q$  は整数値を計算する通常の SQL 問い合わせである。

(2) CombSQL+ からの制約生成法とその正しさの証明: 集合上に拡張された SQL 問い合わせ  $\phi$  から、変換により制約を生成するための SQL 問い合わせ  $\psi$  を生成する方法を提案した。出力された問い合わせ  $\psi$  を要求される性質を満たすデータ処理関数と集計関数とともに問題インスタンスが入れられているデータベースで実行することで、汎用ソルバに入力するための制約が生成される。これらの手法の正しさ、すなわち、生成される制約が集合上の SQL 演算を正しく表していることの証明を与えた。

(3) CombSQL+ の実現、一般公開: 処理系を以下の方針で設計し、これらのシステムを、SQLite3 および z3 をサブシステムとして利用することで、python 言語上で実現し、WEB を通じてシステムを一般公開した

- ① 通常の SQL で実行可能な箇所については、そのまま既存のデータベース処理系で処理する。
- ② それ以外の拡張 SQL は、項目 (2) の変換で得られた SQL を実行することで、線形整数問題の形式で制約を生成し、汎用ソルバ Z3 で求解する。得られた解は、データベースに通常の表として格納する。

#### 4. 研究成果

(1) 研究の主な成果： 組合せ最適化問題を容易に記述可能で、かつ効率的に解くことができるシステム CombSQL+ を提案・実装・公開した。以下で説明するように、SQL を一度でも使ったことがあれば、この言語を理解することは容易であり、最適化問題をツールを使って解くことに対する敷居を大きく下げることになった。当然ながら複雑な SQL を利用しないと書けないような複雑な組合せ問題の記述はそれなりの SQL の経験が必要となるが、導入時の敷居が低いことは大きなメリットである。以下では、これらについて順に説明する。

組合せ最適化問題の記述言語としては、解の候補としてデータベースのテーブルの集合を定義し、テーブルが満たすべき条件を SQL で記述することで解を定義し、また、目的関数も整数を返す SQL で記述することで最適化問題の記述を可能にしている。ごく単純な最適化問題を例題として説明する。表 1 のテーブルを考える。このテーブル Obj は、1,2,3 を要素に持つ集合と解釈できる。このような整数の集合が与えられたときに、「数の合計が 3」になる部分集合のうちで、要素数が最大のもを求める問題 Sel を考える。

表 1 テーブル Obj

num
1
2
3

この問題は図 1 のように CombSQL+ 言語で記述される。1 行目は、テーブル Obj の全ての部分テーブル（すなわち全ての部分集合）からなる集合を SearchSp と定義している。3-4 行目は、集合の内包表現のように、先に定義した集合中のテーブル  $t$  の内で 4 行目を満たす、すなわち、要素の合計が 3 となるテーブルの集合を Solutions と定義している。6-7 行目は、表の集合 Solutions から、7 行目のようにテーブルの行数が最大のテーブルを選んで、solution と定義している。

```

1 CREATE SET SearchSp AS SUBTABLE OF SELECT * FROM Obj;
2
3 CREATE SET Solutions HAS t IN SearchSp SUCH THAT
4   (SELECT TOTAL(num) FROM t) = 3
5
6 CREATE TABLE solution AS t IN Solutions
7   MAXIMIZING(SELECT COUNT(*) FROM t)
    
```

図 1 例題の CombSQL+ による記述

開発した CombSQL+ 処理系は、図 2 に示す構成になっている。システムの入力は、先に説明した図 1 に示すような問題記述と、表 1 に示すような問題のインスタンスが入ったデータベースファイル（テーブル）からなり、その最適解のうちのひとつが格納されたテーブルをデータベースの中に生成する。

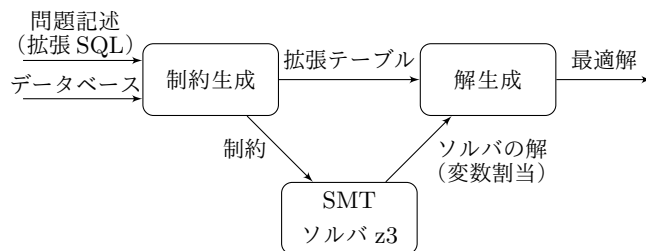


図 2 CombSQL+ システムの構成

制約生成部は、問題記述中の SQL 式から変換で得られる SQL 問合せを、問題のインスタンスが格納されているデータベースのもとで実行することにより線形整数制約 (QFLIA) に変換し、SMT ソルバ z3 に渡す。ここで、変換後の SQL 問合せの実行では DB エンジンとして、副作用をもつユーザ関数の利用が可能な SQLite を用いており、変換後の SQL 中の制約生成関数の副作用により制約を SMT ソルバに送っている。

目的関数に基づく解の最適化については以下のように処理されている。SMT ソルバの最初の実行により解のうちのひとつが求められると、その目的関数値  $v$  が得られるようになっている。さらに、得られた目的関数値より良い解のみに制限するための阻止制約を SMT ソルバに追加して実行することを繰り返すことで、その最適解を求めている。

本システムの核は制約生成の部分であり、以下ではこれについて説明する。テーブルの集合は、拡張テーブル  $R^+$  と制約  $\psi$  の組からなる制約付きテーブルで保持する。ここで拡張テーブル  $R^+$  は、通常の値の他に制約変数を持つことができ、かつ、プール型のカラム ext を持つテーブルである。また、制約  $\psi$  は拡張テーブルが持つ制約変数上の論理式である。制約変数への値割当  $\alpha$  と拡張テーブル  $R^+$  から定まるテーブル  $\alpha(R^+)$  は、 $R^+$  の変数に  $\alpha$  の値を割り当てた上で、カラム ext の値が偽となる行をすべて取り除き、さらにカラム ext を削除して得られるテーブルである。制約付きテーブル  $\langle R^+, \psi \rangle$  は、制約  $\psi$  を満たすよう

な割当  $\alpha$  で定まるテーブル  $\alpha(R^+)$  すべてからなる集合を表す。

例えば SearchSp は、表 2 の制約付きテーブルと恒真な制約の組  $(R_{\text{SearchSp}}^+, \text{true})$  により表現される。また、割当  $\alpha = \{x_1 \mapsto \text{true}, x_2 \mapsto \text{true}, x_3 \mapsto \text{false}\}$  に対して、 $\alpha(R_{\text{SearchSp}}^+)$  は表 3 に示されるテーブルである。

表 2 制約付きテーブル  $R_{\text{SearchSp}}^+$

num	ext
1	$x_1$
2	$x_2$
3	$x_3$

表 3  $\alpha(R_{\text{SearchSp}}^+)$

num
1
2

制約付きテーブルの定義により、問題記述中の SQL 問い合わせから、評価により線形整数制約

を生成する SQL 問い合わせの変換の定義、ならびに、正しさの証明ができる。さらに、システムの設計および実装をスムーズに行えた。先の例題においては、変換で得られた SQL の実行により次の線形整数制約が得られる。

$$\psi: y_1 \Leftrightarrow \text{ite}(x_1, 1, 0) \wedge y_2 \Leftrightarrow \text{ite}(x_2, 2, 0) \wedge y_3 \Leftrightarrow \text{ite}(x_3, 3, 0) \wedge y = y_1 + y_2 + y_3 \wedge z \Leftrightarrow y = 3$$

ここで、ite は if 式、すなわち、第 1 引数が真のときに第 2 引数を表し、そうでないときに第 3 引数を表す関数である。汎用ソルバ z3 により解  $\{x_1 \mapsto \text{true}, x_2 \mapsto \text{true}, x_3 \mapsto \text{false}\}$  が得られるが、この割当  $\alpha$  と表 2 の制約付きテーブルから、表 3 の解が得られる。

システム評価の結果、汎用ソルバでは if 式のようにブール値と整数値の間での橋渡しがある場合に特に効率が悪化することが判明した。そこで、線形整数制約に加えて擬ブール制約 (Pseudo-Boolean Constraint) の形式で表現可能な場合に、この制約を使うことで汎用ソルバでの求解速度が向上し、問題によっては 10 倍の高速化が認められ、その場合には人が直接コーディングした制約を汎用ソルバで解くのと遜色のない性能が確認された。例えば、上の論理式  $\psi$  は、この制約の形式では  $z \Leftrightarrow 1y_1 + 2y_2 + 3y_3 = 3$  のように表現される。

## (2) 得られた成果の国内外における位置づけとインパクト :

組合せ最適化問題用の解法のために、OPL, EaCL, NCL, ESRA, ESSENCE, MiniZinc, Copris など国内外に複数のツールが存在し、その速度にしのぎを削っている。本研究の手法は、それとは異なる方向の、気軽に使用できるシステムという位置づけであるものの、速度についても非常に劣っているというほどではない。一方理論面においては、関係データベースの基礎理論となるデータ代数を集合上の代数に自然な形で拡張し、それが実用になることを示した点においてインパクトがある。

## (3) 今後の展望 :

組合せ問題が容易に記述にできるようになり SQL 言語さえ知っていれば利用可能という状況にはなったものの、組合せ問題の記述中において SQL 問い合わせの誤りの原因の特定に困難を感じることもあり、使いやすい利用環境の構築が望ましい。特に、通常の SQL 問い合わせを構成するには特定のデータに対して実行することによるデバッグが標準的に行われるが、組合せ問題ではユーザが与える表と記述中の SQL 問い合わせ対象の探索空間の要素の表に乖離がある。この乖離を埋めることは理論的には単純で、探索空間であるテーブルの集合の要素のうちの一つをその代表として実行させることにより、利用者が記述している SQL 問い合わせが所望の動作か否かの確認が出来る。このような機能を有する使いやすいユーザインタフェースの作成は手間のかかる作業であるが、その構築が望まれる。

現状のシステムにおいては、SQL の構文のうちで WITH の利用による不具合が確認されている。理論面での誤りがないことは確認済みであるが、システム改修についてはそれほど容易ではなく、核となる SQL 変換の部分の再構築が近道と考えられる。

さらなる高速化については、擬ブール制約の導入と同様に、All different 制約などの汎用ソルバの高速化に役立つ機能を利用できるようにすることが効果的であろう。

(4) 新たな知見 : 研究計画当初は、SQL 言語に Find 構文と呼ぶ組合せ問題記述に特化した構文を導入することにより SQL 言語を拡張することにより組合せ最適化問題を記述する方針であった。しかしながらこの方法でシステムを設計しようとしても、もともとの SQL 言語自体が複雑であることもあり、システム全体が複雑になりすぎて実装が困難になるばかりか、正しさの保証も出来そうになかった。その言語が表現

する意味の形式化を諦めかけていた頃に、突然、意味の領域として表の集合を扱うというアイデアが浮かんだ。その結果、通常の表を表すデータ代数を、表の集合を扱うという単純かつ自然な拡張を行っただけで、システムを複雑にしていた問題点の殆どを一気に解決することに成功した。それによりシステムの設計の正しさの形式的な証明も可能になり、実現も容易になった。

5. 主な発表論文等

〔雑誌論文〕 計2件（うち査読付論文 1件/うち国際共著 0件/うちオープンアクセス 1件）

1. 著者名 Sakanashi Genki, Sakai Masahiko	4. 巻 PPDP2018
2. 論文標題 Transformation of Combinatorial Optimization Problems Written in Extended SQL into Constraint Problems	5. 発行年 2018年
3. 雑誌名 Proceedings of the 20th International Symposium on Principles and Practice of Declarative Programming (PPDP 2018), ACM, isbn:978-1-4503-6441-6	6. 最初と最後の頁 19:1-19:13
掲載論文のDOI (デジタルオブジェクト識別子) 10.1145/3236950.3236963	査読の有無 有
オープンアクセス オープンアクセスとしている (また、その予定である)	国際共著 -

1. 著者名 Genki Sakanashi, Masahiko Sakai	4. 巻 B903-03
2. 論文標題 Transformation of SQL-based Combinatorial Optimization Problems into Constraint Problems	5. 発行年 2020年
3. 雑誌名 The Japanese Society for Artificial Intelligence, SIG-FPAI	6. 最初と最後の頁 12-17
掲載論文のDOI (デジタルオブジェクト識別子) なし	査読の有無 無
オープンアクセス オープンアクセスではない、又はオープンアクセスが困難	国際共著 -

〔学会発表〕 計8件（うち招待講演 1件/うち国際学会 2件）

1. 発表者名 Masahiko Sakai
2. 発表標題 An extended SQL for combinatorial optimization problems and transformation into constraint problems
3. 学会等名 the 49th TRS meeting (国際学会)
4. 発表年 2018年

1. 発表者名 Genki Sakanashi and Masahiko Sakai
2. 発表標題 Transformation of combinatorial optimization problems written in extended SQL into constraint problems
3. 学会等名 IPSJ-SIGPRO
4. 発表年 2018年

1. 発表者名 坂梨元軌, 酒井正彦, 西田直樹, 橋本健二
2. 発表標題 組合せ最適化問題の記述からSMTソルバの入力式を生成するSQL問合せ
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2019年

1. 発表者名 坂梨元軌, 酒井正彦, 西田直樹, 橋本健二
2. 発表標題 組合せ最適化問題を記述するための関係代数の集合上への拡張
3. 学会等名 第20回プログラミングおよびプログラミング言語ワークショップ PPL2018
4. 発表年 2018年

1. 発表者名 Masahiko Sakai
2. 発表標題 SQL for combinatorial optimization problems and SMT-based solving by SQL transformation
3. 学会等名 6th International Workshop on Rewriting Techniques for Program Transformations and Evaluation (WPTe 2019) (招待講演) (国際学会)
4. 発表年 2019年

1. 発表者名 Genki Sakanashi, Masahiko Sakai
2. 発表標題 Transformation of SQL-based Combinatorial Optimization Problems into Constraint Problems
3. 学会等名 The 113th meeting of Special Interest Group on Fundamental Problems in Artificial Intelligence (SIG-FPAI)
4. 発表年 2020年

1. 発表者名 磯貝孝明, 橋本健二, 酒井正彦
2. 発表標題 命題論理式の全ての投射モデルを表現するBDDの構成法
3. 学会等名 人工知能学会 第116回人工知能基本問題研究会
4. 発表年 2021年

1. 発表者名 岸潤一郎, 酒井正彦, 西田直樹, 橋本健二
2. 発表標題 擬ブール制約の導入による組合せ最適化ソルバCombSQL+の高速化
3. 学会等名 電子情報通信学会ソフトウェアサイエンス研究会
4. 発表年 2021年

〔図書〕 計0件

〔産業財産権〕

〔その他〕

システムの公開： <a href="https://www.trs.css.i.nagoya-u.ac.jp/projects/CombSQLplus/">https://www.trs.css.i.nagoya-u.ac.jp/projects/CombSQLplus/</a>
---

6. 研究組織		
氏名 (ローマ字氏名) (研究者番号)	所属研究機関・部局・職 (機関番号)	備考

7. 科研費を使用して開催した国際研究集会

〔国際研究集会〕 計0件



8 . 本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関
---------	---------