

SAT ソルバーを用いた命題論理問題の説明と具体的問題に対する計算機実験

文理学部情報科学科
5419045 高林 秀

2021 年 10 月 30 日

概要

本稿は、今年度論理と計算 2 における課題学習として「命題論理の説明」及び「SAT ソルバーを使用した、その具体的問題の解決を行う計算機実験」を行うものである。本稿の冒頭～中盤では関係理論の説明を行い、終盤ではその理論を利用して、実際に具体的な問題を SAT ソルバーを使用して解答する。なお、本演習には SAT ソルバーとして clasp を使用した。

1 目的

本稿は、今年度論理と計算 2 における課題学習として、SAT ソルバーを用いた命題論理による宣言的問題解決を通じ、命題論理に関する学修内容を振り返ることを目的とする。

本稿は大まかに次のように構成される。

1. 計算理論説明
 - (a) 命題論理における解釈とモデル、その他関連する事項について
 - (b) SAT 問題とはなにか
 - (c) DPLL アルゴリズムの解説
2. 計算機実験
 - (a) N 人の女王
 - (b) グラフ頂点の彩色
3. 各問に関する考察
4. まとめ
5. 巻末資料

2 計算理論説明

この章では、今回の計算機実験に使用した各計算理論の解説を行う。

2.1 命題論理とは

まず命題論理とはなにか説明する。命題論理という言葉の意味はデジタル大辞泉に次のように書かれている。

記号論理学の基礎的部門。個々の命題を結合する「かつ」「または」「ならば」「でない」などの関係を、論理記号を用いて論理積 (\wedge)・論理和 (\vee)・含意 (\Rightarrow)・否定 (\neg) などにより記号化して演算形式に表し、複合された命題を研究する学問。命題計算。

そもそも命題は、数学では「真偽の判断の対象となる文章または式」であり、論理学においては「判断を言葉で示したもので真または偽」という性質を持つもの、という意味である。したがって、命題論理とは、命題同士の関係性を論理記号を使用して記号化し、演算できるようにしたものということだ。

■結合子 今説明したように、命題論理では命題同士の性質、関係性を扱う。それを説明する上で「結合子 (論理記号)」と呼ばれるものが定義されている。

記号	意味
\wedge	連言 (and, &&)
\vee	選言 (or,)
\neg	否定 (not)
\Rightarrow	伴意
\Leftrightarrow	同値

■命題文～原子文, 複合文 命題論理は次の要素から構成される。

- 文、命題文 (sentence) ※これは命題論理式とも言う。
 - 原子文 (atomic formula) : これ以上分解することができない命題。最も単純な文。いわゆる 1 つの命題であり、以下の例のようにそれぞれ固有の記号で示すことができる。
(例) p : 「動物はいつか死ぬ」, q : 「人間は動物である」, z : 「人間はいつか死ぬ」
 - 複合文 (complex sentence) : 文同士を「結合子」で連結した文。命題同士の関係性を結合子を利用して連結し、新たな文を作ることができる。
(例) $p \wedge q \Rightarrow z$: 「動物はいつか死ぬ」かつ「人間は動物である」ならば「人間はいつか死ぬ」。

まとめると、命題文には原子文や複合文と呼ばれる区分けが存在し、原子文は「真 (true)、偽 (false)、それ以上分解できない命題 (記号)」であり、複合文は「命題文同士を結合子で連結した新たな命題文」ということになる。

なお、true, false と呼ばれる命題の真偽を示すこれらの記号は論理定数と呼ばれる。原子文の例で示した命題文を各記号に置き換えたもの p, q, z は命題記号ないしは命題変数と呼ばれる。

■命題文の構文 これらの要素を組み合わせで作られる命題文は、使用する結合子によって以下に区分けされる。

- 否定文 : 結合子 \neg で連結されている複合文。

- 連言文：結合子 \wedge で連結されている複合文。
- 選言文：結合子 \vee で連結されている複合文。
- 含意文：結合子 \Rightarrow で連結されている複合文。
- 同値文：結合子 \Leftrightarrow で連結されている複合文。

とくに含意文 $\alpha \Rightarrow \beta$ については α を前提、 β を帰結と呼ぶ。また、原子文とその否定文 $\neg p, \neg q, \neg z$ をひとくくりにしてリテラルと呼ぶ。

■知識ベース また、命題文の集合において、各文を結合子 \wedge で連結したものを知識ベースと呼ぶことがある。(例： $KB[\text{KnowledgeBase}] = \{S1, S2, S3\} \Rightarrow S1 \wedge S2 \wedge S3$)

2.1.1 真理値表 (truth table)

命題文の真偽は先に上げた論理定数 true, false で示す。複合文の真偽を表にまとめて示したものを真理値表と呼ぶ。先に上げた各構文の真理値は、真理値表を用いて次のように定義されている。

2.2 命題論理における解釈

2.3 モデルについて

2.4 SAT 問題とは

2.5 DPLL アルゴリズム

3 計算機実験

3.1 実験準備

3.1.1 実験環境

今回の実験は仮想マシン上で clasp のバイナリをダウンロードして行った。下記に実験時の環境を示す。

- ホスト OS : Window10 Home 20H2
- 仮想 OS : Ubuntu 20.04.2 LTS
- CPU : Intel(R)Core(TM)i7-9700K @ 3.6GHz
- GPU : Nvidia Geforce RTX2070 OC @ 8GB
- ホスト RAM : 16GB
- 仮想 RAM : 4GB

3.1.2 問題 1: N 人の女王

配布資料中に、Processing のプログラムが「nQueen.pde」として以下の関数が用意されている。

- バックトラック法を用いて nQueen を解く関数
- clasp への入力ファイルを作成する関数

1. この問題に対する SAT 符号化を詳細に説明せよ。

2. N の大きさを様々に変えながら、バックトラック法で解いた場合と SAT ソルバーで解いた場合とでの実行時間を比較・考察しなさい。

3.1.3 問題 2: グラフ頂点の彩色問題

配布資料中の「GraphColoring」フォルダに、「都道府県の隣接関係」を表すグラフの頂点彩色問題の CNF ファイルが用意されている。

1. この問題に対する SAT 符号化を詳細に説明せよ。
2. 関東地方を対象に、いくつの塗分け方法があるか調べなさい。
3. 47 都道府県を対象とした色塗りの例を一つ示しなさい。
(a) (例) 長野県：青色, 神奈川県：赤色、のように、どの都道府県をどの色で塗るのかを具体的に示すこと。

3.2 各問に対する解答・考察

3.2.1 問題 1: N 人の女王

3.2.2 問題 2: グラフ頂点の彩色問題

4 まとめ

5 巻末資料

本稿で使用した画像、プログラムコード等はすべて以下のリンク先に掲載している。必要に応じてご覧頂きたい。

- GoogleDrive: https://drive.google.com/drive/folders/1k0W_1KPUw_kBznaMWjge7HaBI7FoRAoq?usp=sharing
- GitHub: https://github.com/tsyu12345/logical_and_calculating_LectureCode/tree/master/No5