

doi: 10.3969/j.issn.1674-8425(z).2020.05.033

本文引用格式:杨晗,宋振明.基于子句长度的分支策略[J].重庆理工大学学报(自然科学),2020,34(5):252-255.

Citation format: YANG Han, SONG Zhenming. Branching Strategy Based On Clause Length [J]. Journal of Chongqing University of Technology (Natural Science), 2020, 34(5): 252-255.

基于子句长度的分支策略

杨 晗, 宋振明

(西南交通大学 数学学院, 成都 610000)

摘 要: 研究 SAT 问题不仅具有重要的理论意义, 而且具有实际应用价值。设计高效的 SAT 问题的算法是国内外研究的热点。而分支策略是影响求解算法关键因素之一, 目前大多数分支策略都是基于 VSIDS 策略提出的, 虽然存在很多分支策略, 但仍不能满足实际需要。为此, 针对求解 SAT 问题实例中出现子句长度不一致的情况, 提出一种新的分支策略, 基于子句长度的分支策略 (BSBCL), 并通过实验验证其有效性。

关 键 词: SAT 问题; VSIDS; 求解算法; 分支策略; 子句长度

中图分类号: TP181

文献标识码: A

文章编号: 1674-8425(2020)05-0252-04

Branching Strategy Based On Clause Length

YANG Han, SONG Zhenming

(School of Mathematics, Southwest Jiaotong University, Chengdu 610000, China)

Abstract: Studying the SAT problem not only has important theoretical significance but also has practical application value. The algorithm for designing efficient SAT problem is a hot topic at home and abroad. Branching strategy is one of the key factors affecting the solution algorithm. At present, most branching strategies are based on the VSIDS strategy. Although there are many branching strategies, they still cannot meet the actual needs. In this paper, a new branching strategy based on clause length branching strategy (BSBCL) is proposed to solve the problem of inconsistent clause length in the solution of SAT problem, and verify its effectiveness through experiments.

Key words: SAT problem; VSIDS; solving algorithm; branching strategy; clause length

从 1960 年开始, 可满足问题 (satisfiability problem, SAT) 逐渐出现在人们的视野中, 由于它的重要的理论价值和实际应用意义, 受到众多学者的关注, 研究 SAT 问题的算法逐渐成为了国内

外研究的热点^[1]。

求解 SAT 问题的算法是指在有限时间内, 判断以合取范式形式给出的命题逻辑公式的是否可满足, 在可满足的情况下, 算法会给出一个使得公

收稿日期: 2018-12-13

基金项目: 国家自然科学基金项目“基于矛盾体分离的动态自动演绎推理研究”(61673320)

作者简介: 杨晗, 女, 硕士研究生, 主要从事智能化信息处理研究, E-mail: 1659116121@qq.com; 宋振明, 男, 教授, 博

万方数据 主要从事智能化信息处理研究, E-mail: zhmsong@home.SwjtU.edu.cn。

式可满足的赋值。SAT 问题的算法可以分为完备算法和不完备算法两大类。完备算法不仅可以判断问题的可满足性,而且在问题不满足时可以给出证明。不完备算法在求解可满足性问题时往往比完备算法求解速度快,但它不能判断问题的不可满足性^[2],很多著名求解器都是基于完备算法提出,如: Minsat^[3]、glucose^[4]、Lingeling^[5]、glucose4.1^[6]、inIDGlucose^[7] 本文主要基于完备算法。

CDCL(conflict driven clause learning)^[8]是目前流行的完备算法,它是在经典的完备算法 DPLL(davis putnam longmann loveland)^[9]算法基础上发展而来,分支策略是 CDCL 算法中的重要过程,它也是影响算法求解效率的关键因素之一。VSIDS(variable state independent decaying sum)^[10] 分支策略的提出为分支策略的发展做出巨大的贡献。近几年的 SAT 竞赛数据显示 VSIDS 策略或者其变种是较为高效的分支策略^[11]如 CHB(conflict history-based branching heuristic)^[12],LRB(learning rate branching)^[13],WVSIDS(weig-hted VSIDS)^[14]。

虽然存在很多分支策略,但仍不能满足实际的需要,本文基于子句长度提出一种新的分支策略。

1 基本知识

定义 1^[15-16] 命题逻辑中的命题称为命题变量,常用小写字母 x, y, z, \dots 表示。命题变量的否定形式用 $\neg x$ 表示。命题变量集 X 是命题变量的集合。

定义 2^[15] 变量 x 或其否定 $\neg x$ 称为文字,用 l 表示文字, x 称为正文字, $\neg x$ 称为负文字。

定义 3^[15] 真值赋值是一个从命题变量集到真假值集 $P\{1, 0\}$ 的函数。

定义 4^[15] 合取范式(conjunctive normal form, CNF)是若干个子句的合取,其一般形式为

$$\varphi = C_1 \wedge C_2 \wedge \dots \wedge C_n$$

其中 $C_i (i = 1, 2, \dots, n)$ 表示子句是一些文字的析取,形如:

$$l_1 \vee l_2 \vee \dots \vee l_m$$

定义 5^[15-17] SAT 问题是指判断在命题集 X 上以合取范式形式给出的命题逻辑公式 $\varphi(X)$, 是否存在一组使得 $\varphi(X)$ 为真的真值赋值,若存在,则称 $\varphi(X)$ 是可满足的,否则称为不可满足的。

任意一个命题逻辑公式都可在多项式时间内转化为合取范式,对于 SAT 问题的求解算法中通常所说的逻辑公式是合取范式的形式。

例 1 设有公式

$$\varphi(X) = C_1 \wedge C_2 \wedge C_3$$

$$C_1 = x_1 \vee x_2 \vee \dots \vee \neg x_3$$

$$C_2 = \neg x_2 \vee x_4$$

$$C_3 = \neg x_1 \vee x_3 \vee \neg x_4$$

存在一组真值赋值 $\{x_1 = 1, x_2 = 0, x_3 = 1, x_4 = 0\}$ 使得 $\varphi(X) = 1$, 故命题逻辑公式 $\varphi(X)$ 是可满足的。

2 基于子句长度的分支策略

2.1 BSBCL 策略

在冲突(即在一组赋值之下公式中所有子句的赋值为 0)产生之前的分支变量,称之为初始变量,初始变量的选择,直接影响整个搜索空间的大小,选择不好的策略,在求解过程中会产生大量冲突,反复进行冲突分析,产生学习子句,减慢布尔约束传播过程(BCP),进而影响求解效率,增大求解时间。故初始变量的选择策略,直接影响问题的求解效率。

对于在 VSIDS 策略基础上发展而来的分支策略无论是 ESIDS(Exponential VSIDS)^[18] 策略、ACIDS(average conflict index decision score)^[19] 策略,在初始变量选择时采用 VSIDS 策略的方式,即为变量设置一个计数器记录变量在子句中出现的次数,选择最大的变量作为分支变量。忽略子句的长短对于初始变量选择的影响,故本文提出一种新的分支策略,基于子句长度的分支策略(branching strategy based on clause length, BSBCL)。

1) 在求解过程中会出现较长子句(子句中的文字数较多)和较短子句(子句中文字数较少),选择较短子句的变量作为分支变量,则可得到更多

变量信息,进而加速求解过程。

2) 子句中的文字越多,越有可能被满足,子句中文字越少,越不可能被满足,如单子句(只含有一个文字的子句) x_1 ,只有在 x_1 被赋值为1时才能被满足。对于子句 $x_1 \vee x_2$,在 x_1 赋值为1时可满足,并且在 x_1 赋值为1时也可满足,故子句中文字越多,越有可能被满足,越少越不可能被满足,

通过以上2点,故优先选择较短子句中的作为分支变量。

若给出一个子句集含有 n 个子句,具体策略如下:

1) 计算子句的长度,记为: $\text{leng}(C_i), i=1,2,\cdots,n$ 。

2) 计算子句集中子句的平均长度,记为: $\text{aver}(C)$ 。

3) 子句 C_i 中的每个变量得分为 $\frac{\text{aver}(C)}{\text{leng}(C_i)}$,记为: $t_i, i=1,2,\cdots,n$ 。

4) 若含有变量 x_j 的子句为 C_1, C_2, \cdots, C_n ,则变量 x_j 的得分为 t_i 之和,记为

$$s(x_j) = \sum_{i=1}^n \frac{\text{aver}(C)}{\text{leng}(C_i)}$$

5) 选择得分最高的变量作为分支变量,并默认赋值为0(本文采取Minsat默认赋值为0的方式)。

2.2 BSBCL 策略分析

若 $\text{avg}(C) < \text{leng}(C_i)$ 子句 C_i 为较长子句,则 $\frac{\text{avg}(C)}{\text{leng}(C_i)} < 1$ 。

若 $\text{avg}(C) > \text{leng}(C_j)$ 子句 C_j 为较短子句,则 $\frac{\text{avg}(C)}{\text{leng}(C_j)} > 1$ 。

则较长子句中变量得分小于较短子句变量得分,即优先选择较短子句中的变量。

下面给出一个具体的例子来说明BSBCL策略比VSIDS策略的不同之处。

例2 设有公式
 $\varphi(X) = C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \wedge C_7 \wedge C_8$
 $C_1 = x_1 \vee x_2, C_2 = \neg x_1 \vee x_2 \vee x_7$

$$C_3 = \neg x_2 \vee x_6 \vee \neg x_8, C_4 = \neg x_1 \vee \neg x_4$$
$$C_5 = x_3 \vee x_5 \vee \neg x_6, C_6 = \neg x_2 \vee x_4 \vee x_7$$
$$C_7 = x_7 \vee x_9, C_8 = x_6 \vee x_8 \vee \neg x_9$$

子句的平均长度为 $\text{avg}(C) = 2.63$ (保留两位小数),含有变量 x_1 的子句有 C_1, C_2, C_4 ,则 x_1 得分通过以下方式计算 $S(x_1) = t_1 + t_2 + t_4 = 1.3 + 0.87 + 1.3 = 3.47$,通过各个变量得分通过以上方式计算,得到表1。其中 S_i :表示通过VSIDS策略计算方式的变量得分; S 为通过上面介绍的计算方式的变量得分。

表1 各变量得分

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
s_1	3	3	1	2	2	3	3	2	2
s	3.47	2.61	0.87	2.17	2.17	2.61	3.04	1.74	2.17

根据VSIDS策略可知:当得分相同时,随机选择变量作为分支变量,在此例中首先选择 x_1 ,决策层为1,并赋值为0,记为 $x_1 = 0@1$,子句 C_1 布尔传播可得 $x_5 = 1@1$,按照这种方式可得变量赋值及决策层,即 $x_2 = 0@2, x_6 = 0@3, x_7 = 0@4$,由子句 C_7 可得 $x_9 = 1@4$,由子句 C_8 可得 $x_8 = 1@4$ 。

根据本文提出的新的初始变量选择策略,选择得分最高的变量进行赋值,选择 x_1 作为分支变量,决策层为1,并赋值为0,记为 $x_1 = 0@1$,子句 C_1 布尔传播可得 $x_5 = 1@1, x_7 = 0@2$,由子句 C_7 可得 $x_9 = 1@2, x_6 = 0@3$,由子句 C_8 可得 $x_8 = 1@3$,由子句 C_3 可得 $x_2 = 0@3$ 。

由此例可以看出选择VSIDS策略决策层数为4,而本文提出的初始变量选择策略决策层数为3,明显新的分支策略比VSIDS策略有明显的优势,决策层数比VSIDS少,决策层数少,避免搜索时因决策树过深,得不到解的情况,即可以起到加速求解的过程。

3 实验结果

将BSBCL策略形成新的算法嵌入到求解器glucose4.1中形成新的求解器glucose4.1+BSBCL。

采用SATLIBBenchmark中例子300个使用求

解器 glucose4.1 和 glucose4.1 + BSBCL 不同的求解器进行测试,测试结果如表 2 所示。

表 2 求解器测试结果

系列	Glucose4.1	Glucose4.1 + BSBCL
Flat200 - 547(100)	0.033 27(s)	0.032 48(s)
BMS_k3_n100_m429(100)	0.015 75	0.014 01
RTI_k3_n100_m429(100)	0.009 87	0.008 17

4 结论

本文分析子句长度对分支策略的影响,提出一种新的分支策略,即基于子句长度的分支策略(BSBLC),优先满足短子句中的变量,有利于加速求解,通过实验结果可知在 300 个问题实例中新的求解器所需的时间要比原来的短,说明 BSBCL 策略的有效性,即起到加速求解的作用。

参考文献:

[1] 赵玉娟. CDCL + SAT 求解器中的分支变量启发式算法研究[D]. 成都:西南交通大学,2017.

[2] 杜仲和. 基于 CDCL 结构的 SAT 问题优化策略的研究[D]. 成都:西南交通大学,2017.

[3] EEN N, SORENSSON N. An Extensible SAT-solver [C]//Theory and Applications of Satisfiability Testing, 2003:502 - 518.

[4] AUDEMARD G, SIMON L. GLUCOSE;a solver that predicts learnt clauses quality[J]. SAT Competition,2009:7 - 8.

[5] BIERE A. Lingeling and friends entering SAT Challenge 2012[J]. Proceedings of SAT Challenge,2012:33 - 34.

[6] OH C, MINISAT HACK 999ED, MiniSat HACK 1430ED and SWDiA5BY [C]//Proceedings of SAT Competition 2014: Solver and Benchmark Descriptions, Volume B-2014-2 of Department of Computer Science Series of Publications B, University of Helsinki, 2014.

[7] DEVRIENDT J, Inidglucose [J]. Proceedings of SAT Competition 2018: Solver and Benchmark Descriptions, volume B-2018-1 of Department of Computer Science Series of Publications B, University of Helsinki 2018.

[8] MARQUES-SILVA J P, LYNCE I, MALIK S. Conflict-driven clause learning SAT solver [M]. In Handbook of Satisfiability, Amsterdam:IOS Press,2009:127 - 149.

[9] DAVIS M, LOGEMANN G, LOVELAND D. A machine program for theorem proving [J]. Communication of ACM,1962,5(7):394 - 397.

[10] MALIK S, ZHAO Y, MADIGAN C F, et al. Chaff: Engineering an efficient SAT solver[J]. Proceedings of DAC-01,2001,17(5):530 - 535.

[11] 刘燕丽,徐振兴,熊丹. 基于动态奖惩的分支策略的 SAT 完备算法[J]. 计算机应用,2017(12):149 - 154.

[12] LIANG J H, GANESH V, POUPART P, et al. Exponential recency weighted average branching heuristic for SAT solvers[C]//Thirtieth AAAI Conference on Artificial Intelligence. AAAI Press,2016:3434 - 3440.

[13] LIANG J H, GANESH V, POUPART P, et al. Learning rate based branching heuristic for SAT solvers [C]//International Conference on Theory and Applications of Satisfiability Testing. Springer, Cham,2016:123 - 140.

[14] CHEN J. GLU_vc:Hacking glucose by weighted variable state independent decay sum branching policy [J]. Proceedings of SAT Competition 2017: Solver and Benchmark Descriptions, volume B-2017-1 of Department of Computer Science Series of Publications. B, University of Helsinki, 2017.

[15] 郭莹,张长胜,张斌. 求解 SAT 问题的算法的研究进展 [J]. 计算机科学,2016,43(3):8 - 17.

[16] 陈青山. 基于矛盾体分离的命题逻辑动态自动演绎推理求解系统研究[D]. 成都:西南交通大学, 2018.

[17] 陈青山,徐扬,何星星. 利用逻辑演绎求解 SAT 问题的启发式完全算法[J]. 西南交通大学学报,2017,52(6):1224 - 1232.

[18] EÉN N, SÖRENSSON N. An Extensible SAT-solver [C]//International Conference on Theory and Applications of Satisfiability Testing,2003:502 - 518.

[19] BIERE A, FROHLICH A. Evaluating CDCL Variable Scoring Schemes [C]//Theory and Applications of Satisfiability Testing,2015:405 - 422.

(责任编辑 何杰玲)