

Comparator Circuits over Finite Bounded Posets [☆]

Balagopal Komarath ¹, Jayalal Sarma ^{*,1}, K.S. Sunil ¹

Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, 600036, India



ARTICLE INFO

Article history:

Received 31 May 2015

Available online 19 February 2018

Keywords:

Comparator circuits

Logspace computation

Polynomial size circuits

ABSTRACT

The comparator circuit model was originally introduced by Mayr *et al.* (1992) (and further studied by Cook *et al.* (2014)) to capture problems that are not known to be P-complete but still not known to admit efficient parallel algorithms. The class CC is the complexity class of problems many-one logspace reducible to the Comparator Circuit Value Problem and we know that $NLOG \subseteq CC \subseteq P$. Cook *et al.* (2014) showed that CC is also the class of languages decided by polynomial size comparator circuit families.

We study generalizations of the comparator circuit model that work over fixed finite bounded posets. We observe that there are universal comparator circuits even over arbitrary fixed finite bounded posets. Building on this, we show the following:

- Comparator circuits of polynomial size over fixed finite *distributive* lattices characterize the class CC. When the circuit is restricted to be skew, they characterize LOG. Noting that (uniform) polynomial sized Boolean circuits (resp. skew) characterize P (resp. NLOG), this indicates a comparison between P vs CC and NLOG vs LOG problems.
- Complementing this, we show that comparator circuits of polynomial size over arbitrary fixed finite lattices characterize the class P even when the comparator circuit is skew.
- In addition, we show a characterization of the class NP by a family of polynomial sized comparator circuits over fixed *finite bounded posets*. As an aside, we consider generalizations of Boolean formulae over arbitrary lattices. We show that Spira's theorem (Spira, 1971) can be extended to this setting as well and show that polynomial sized Boolean formulae over finite fixed lattices capture the class NC^1 .

These results generalize results in Cook *et al.* (2014) regarding the power of comparator circuits. Our techniques involve design of comparator circuits and finite posets. We then use known results from lattice theory to show that the posets that we obtain can be embedded into appropriate lattices. Our results give new methods to establish CC upper bounds for problems and also indicate potential new approaches towards the problems P vs CC and NLOG vs LOG using lattice theoretic methods.

© 2018 Published by Elsevier Inc.

[☆] A preliminary version of this paper was presented at The 42nd International Colloquium on Automata, Languages, and Programming (ICALP 2015), Kyoto, Japan, from where the paper was invited to this special issue.

* Corresponding author.

E-mail addresses: baluks@gmail.com (B. Komarath), jayalal@cse.iitm.ac.in (J. Sarma), sunsk@gmail.com (K.S. Sunil).

¹ Supported by TCS PhD Fellowship at Department of CSE, IIT Madras, India.

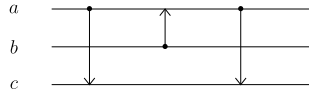


Fig. 1. A comparator circuit.

1. Introduction

Completeness for the class P for a problem is usually considered to be evidence that it is hard to design an efficient parallel algorithm for the problem. However, there are many computational problems in the class P, which are not known to be P-complete, yet designing efficient parallel algorithms for them has remained elusive. Some of the classical examples of such problems include lex-least maximal matching problem and stable marriage problem [5].

Attempting to capture the exact complexity of computation in these problems using a variant of Boolean circuit model, Mayr and Subramanian [5] (see also [2]) studied the comparator circuit model. A comparator circuit is a sorting network working over the values 0 and 1. A comparator gate has two inputs and two outputs. The first output is the AND of the two inputs and the second output is the OR of the two inputs. A comparator circuit is a circuit that has only comparator gates. In particular, fan-out gates are not allowed. Without loss of generality, we can assume that NOT gates are used only at the input level. A graphical representation of a comparator circuit is shown in Fig. 1. In this representation, we draw a set of parallel lines. Each line carries a logical value which is updated by gates incident on that line. Each gate is represented by a directed arrow from one line (say i) to another (say j) and the gate updates the values of lines as follows. The value of line i (j) is set to the AND (resp. OR) of values previously on lines i and j . The gates are evaluated from left to right. The output of the circuit is the final value of a line designated as the output line. We define the model formally in Section 2.

In order to study the complexity theoretic significance of comparator circuits, the corresponding circuit value problem was explored in [5]. That is, given a comparator circuit and an input, test if the output wire carries a 1 or not. The class CC is defined in [5] as the class of languages that are logspace many-one reducible to the comparator circuit value problem. They also observed that the class CC is contained in P. Feder's algorithm (described in [8]) for directed reachability proves that the class CC contains NLOG as a subclass. These are the best containments currently known about the complexity class CC.

There has been a recent spurt of activity in the characterization of CC. Cook et al. [2] showed that the class CC is robust even if the complexity of the many-one reduction to the comparator circuit value problem is varied from AC^0 to NLOG. They also gave a characterization of the class CC in terms of a computational model (comparator circuit families). Their main contribution in this regard is the introduction of a universal comparator circuit that can simulate the computation of a comparator circuit given as input (to the universal circuit). Comparison of CC with the class NC has interesting implications to the corresponding computational restrictions. For example, hardness for the class CC is conjectured to be evidence that the problem is not efficiently parallelizable. This intuition was further strengthened by Cook et al. [2] by showing that there are oracle sets relative to which CC and NC are incomparable (NC is the class of all languages efficiently solvable by parallel algorithms). In addition, it is conjectured in [2] that the classes NC, NC and CC are pairwise incomparable.

Our results & techniques In this paper, we study the computational power of comparator circuits working over arbitrary fixed finite bounded posets. Informally, instead of 0 and 1, the values used throughout the computation could be any element from the poset and the AND and OR gates compute maximal lower bounds and minimal upper bounds over the poset respectively. We define this model formally in section 3. We obtain the following results:

- There exist Universal Comparator Circuits for comparator circuits irrespective of the underlying bounded poset. (Proposition 3, Section 3.)
- Comparator circuits of polynomial size over fixed finite distributive lattices capture the class CC. (Theorem 4, Section 4). This leads to a new way to show that a problem is in the class CC. That is, by designing a comparator circuit over a fixed finite lattice and then showing that the lattice is distributive. (An application of this method to design CC algorithms for the stable matching problem can be found in [5]. See also Section 6.2 in [2]). Since there are lattice theoretic techniques known (cf. M_3 - N_5 Theorem [3]) for showing that a lattice is distributive, this alternate definition of the class CC using comparator circuits over distributive lattices might be of independent interest.
- Going beyond distributivity, we show that comparator circuits of polynomial size over fixed finite lattices characterize the class P. (Theorem 5, Section 4.) In particular, we design a fixed finite poset P over which, for any language $L \in P$, there is a polynomial size comparator circuit family over P computing L . During computation, we only use lubs and glbs that exist in the poset P . This enables the use of Dedekind-MacNeille completion (DM completion) to construct a fixed finite lattice completing the poset P while preserving existing lubs and glbs in the poset and that lattice can be used to perform all computations in P . A potential drawback of the lattice thus obtained is that the complexity class captured by comparator circuits over it may vary depending on the element in the lattice used as the accepting element. By using standard tools from lattice theory, we derive that there is a fixed constant $i \geq 3$, such that comparator circuits over Π_i (where Π_i is the i th partition lattice – see Section 2 for a definition) with polynomial size can compute all

Download English Version:

<https://daneshyari.com/en/article/6873770>

Download Persian Version:

<https://daneshyari.com/article/6873770>

[Daneshyari.com](https://daneshyari.com)