A New Algorithm to Implement Combinational Logic Cells with Reduced Number of Switches

Vinicius Possani, Vinicius Callegaro, Andre Reis,
Renato Ribas, Felipe Marques, Leomar da Rosa Jr.

vnpossani@inf.ufpel.edu.br

Outline

- Introduction
- Motivation
- Our Approach
  - SP Kernel Finder
  - Kernel Composition
- Experimental Results
- Conclusions

Introduction

- In current VLSI design, the total number of transistors necessary to implement a logic gate is strongly related to the **signal delay propagation**, **power consumption** and **area** of integrated circuits.

- **CAD tools** have held designers to manipulate more transistors allowing achieving **optimized cell libraries in a short time**.

Motivation

- Either in factorization methods as in the graph approaches, determine what is **the best point to start** the optimizations process is a not trivial task.

- Considering that, this point has a **strongly influence in the result's quality**, it is needed investigate different possibilities to determine it.

Motivation

- The existent methods try some alternatives to start the algorithm, but in general, this alternatives are related to the **greedy strategies**.

- This way, the **bad choices** in the begin of the optimization process carry the results to limited solutions.

- In this sense, we present a methods based on graph structures called **SP kernels**, which aims to provide a good starting point for the optimizations process.

Our Approach

- The proposed method starts from an irredundant sum-of-products (ISOP), and tries to combine cubes to build SP kernels.

- In a second step the algorithm tries to merge the found kernels in order to deliver an optimized switch network.

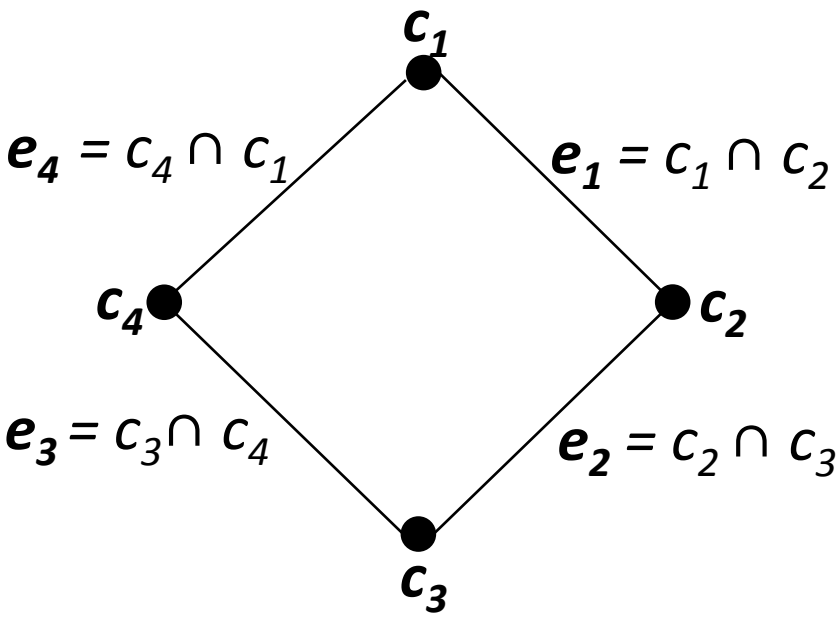
- These steps are run in the following sequence:

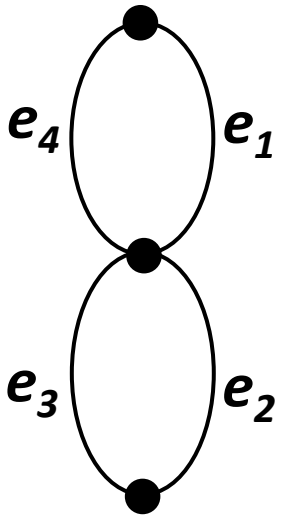
  – ***SP Kernel Finder***

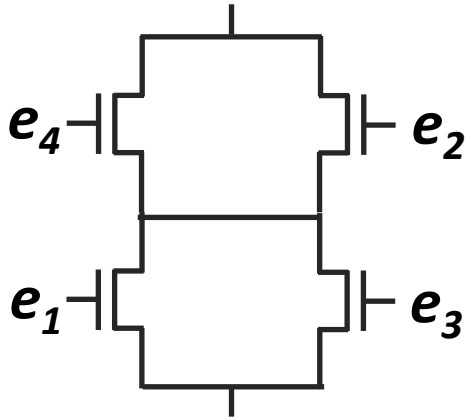
  – ***Kernel Composition***

SP Kernel Finder

- The main idea of this routine is select the cubes of the input ISOP in combinations C_n^4, and try to build **SP kernels**.



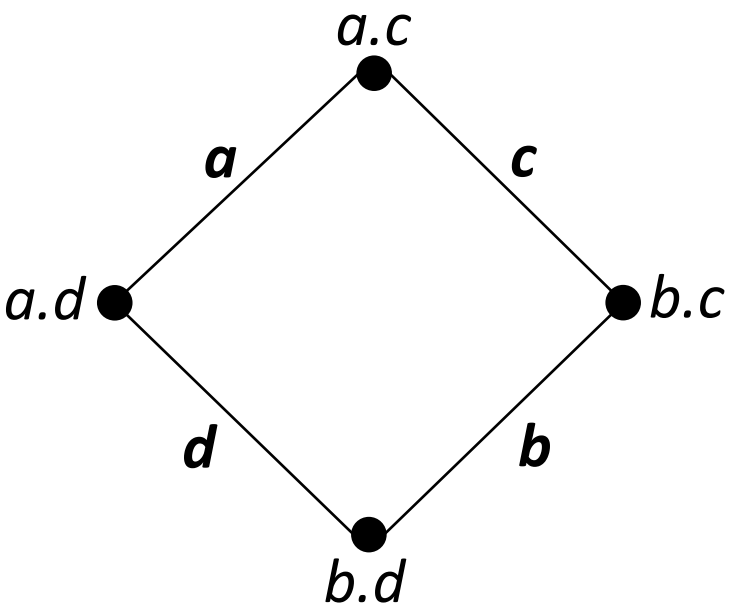


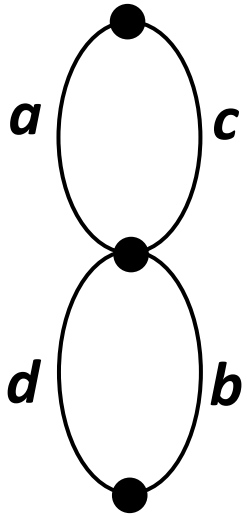


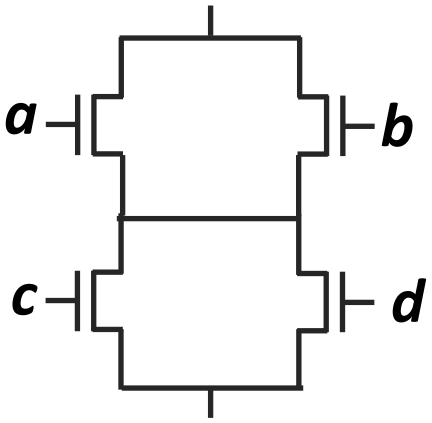
SP Kernel Finder

- For instance, consider as input the follow ISOP:

$$f = a.c + a.d + b.c + b.d$$







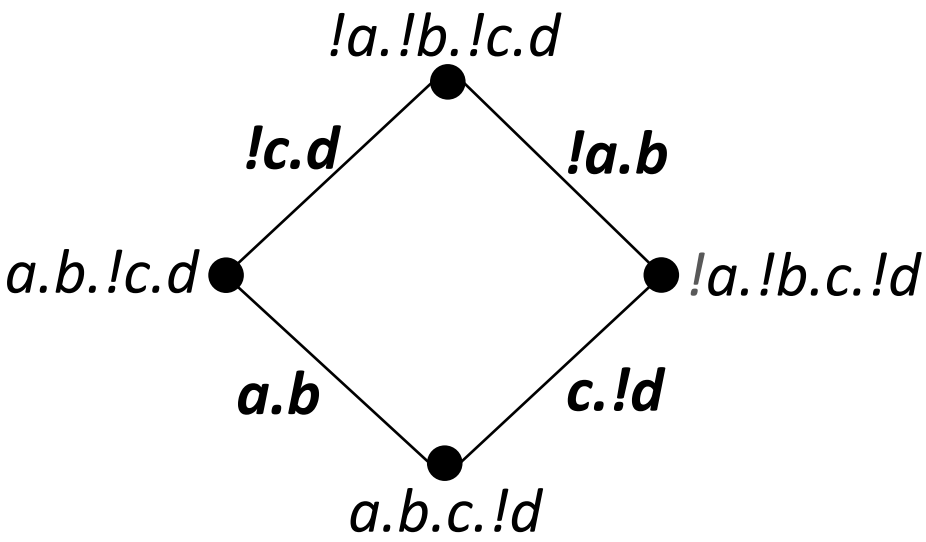
Kernel Composition

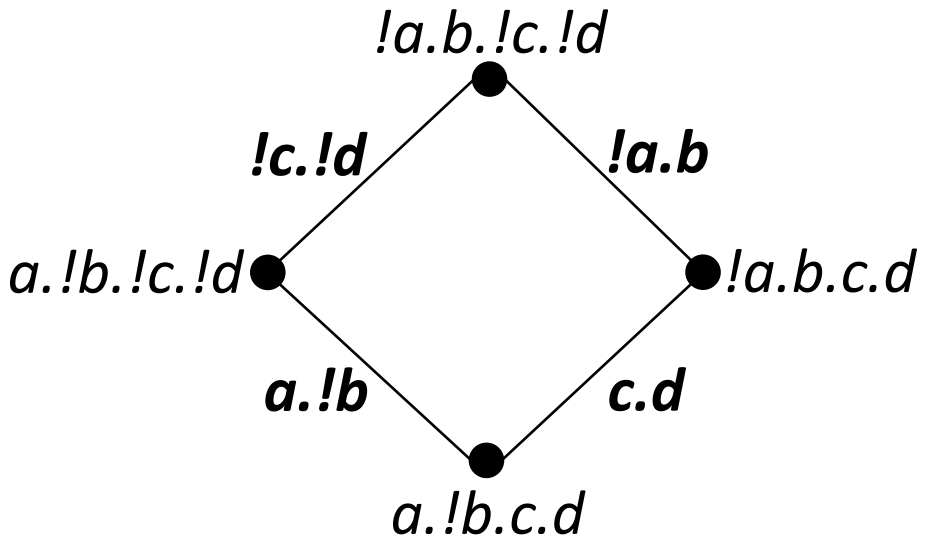
- Depending on the input ISOP multiple kernels may be found.
- To build the transistor network, the found kernels are gradually merged through parallel associations.
- For each kernel added in a parallel arrangement, an edges sharing procedure is applied to remove some redundancies of the network.
- Besides that, some cubes from the input ISOP may not compose any kernel.

Kernel Composition

- For instance, consider as input the follow ISOP:

$$f = !a.!b.!c.d + !a.!b.c.!d + !a.b.!c.!d + !a.b.c.d + a.!b.!c.!d + a.!b.c.d + a.b.!c.d + a.b.c.!d$$

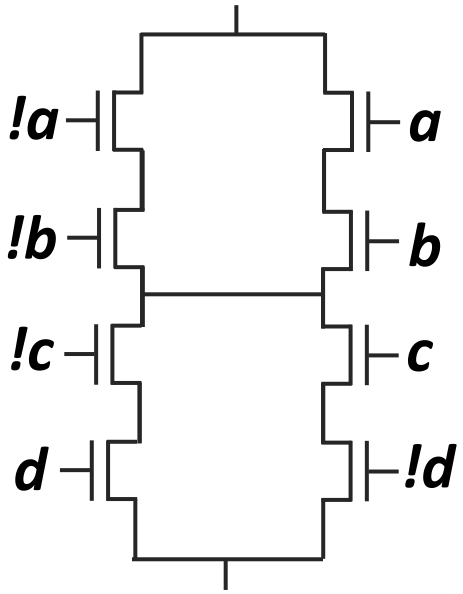


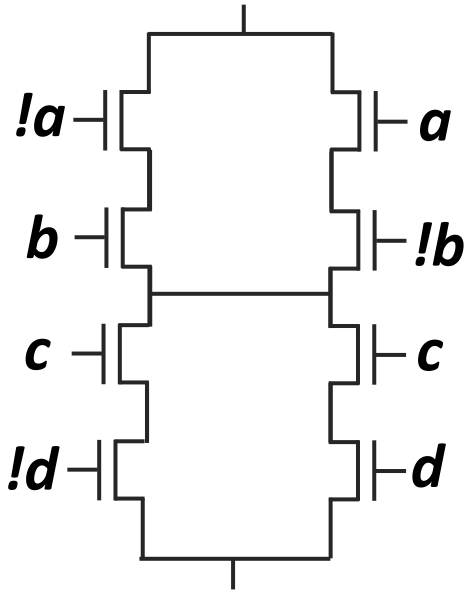


Kernel Composition

- For instance, consider as input the follow ISOP:

$$f = !a.!b.!c.d + !a.!b.c.!d + !a.b.!c.!d + !a.b.c.d + a.!b.!c.!d + a.!b.c.d + a.b.!c.d + a.b.c.!d$$

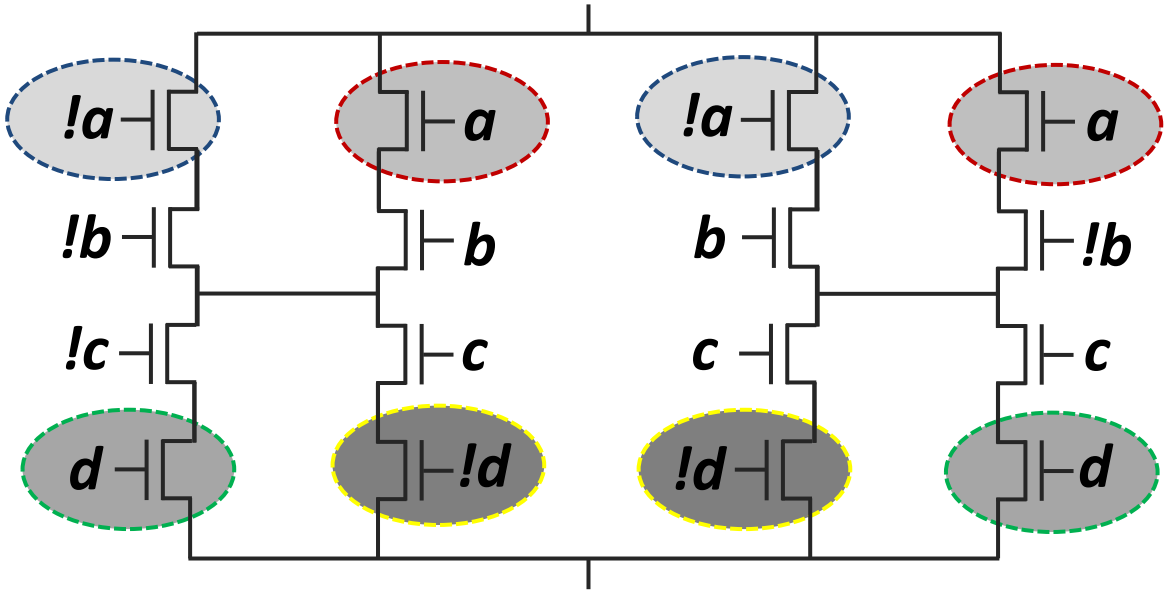




Kernel Composition

- For instance, consider as input the follow ISOP:

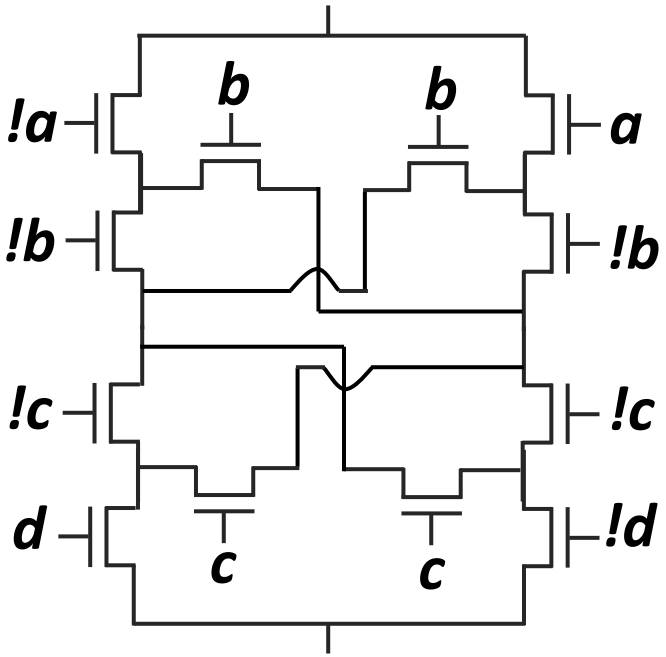
$$f = !a.!b.!c.d + !a.!b.c.!d + !a.b.!c.!d + !a.b.c.d + a.!b.!c.!d + a.!b.c.d + a.b.!c.d + a.b.c.!d$$



Kernel Composition

- For instance, consider as input the follow ISOP:

$$f = !a.!b.!c.d + !a.!b.c.!d + !a.b.!c.!d + !a.b.c.d + a.!b.!c.!d + a.!b.c.d + a.b.!c.d + a.b.c.!d$$



Experimental Results

- The experiments were performed over the set of 4-input P-class logic functions, that is composed by 3982 functions.

- We have generated CMOS logic gates for each functions of this set.

- In this experiment, we also take in to account the number of switches needed to implement the inverters for each variable that appear in both polarity in the functions.

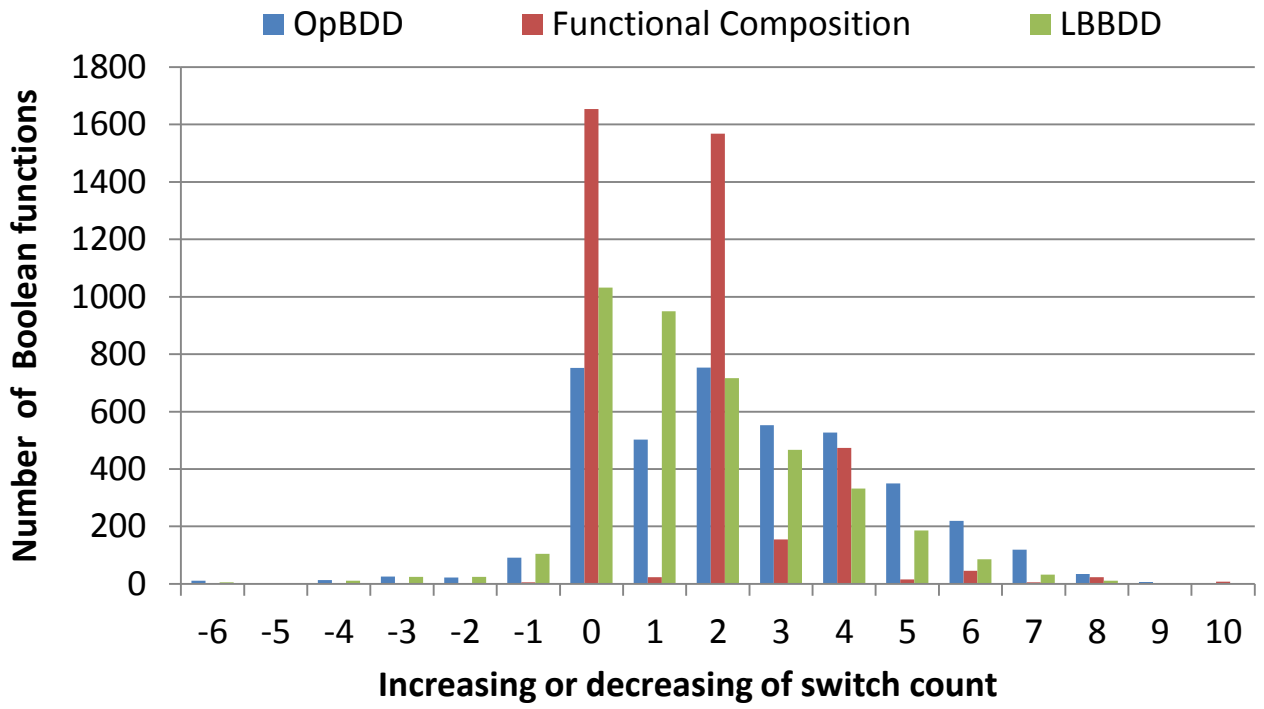
Experimental Results

- The total number of switches needed to implement all functions of P-class was compared with the solution provided by other methods available in the literature.

	FC Matins, 2010	**OpBBD** Da Rosa Junior, 2006	**LBBD** Da Rosa Junior, 2007	**Kagaris** Kagaris, 2007	**Proposed Method**
Total number of switches	106,162	102,668	103,049	97,174	**96,484**

Experimental Results

- Distribution of switch count when comparing the proposed approach to the other methods, considering the set of 4-input P-class logic functions.



Conclusions

- This paper proposed a new graph-based method to generate optimized transistor (switch) networks.
- Through this feature, the method can **avoid the greedy** choices in some steps of the optimizations process.
- The proposed method results in a **reduction of transistor count** when compared to previous approaches.

A New Algorithm to Implement Combinational Logic Cells with Reduced Number of Switches

Vinicius Possani, Vinicius Callegaro, Andre Reis,
Renato Ribas, Felipe Marques, Leomar da Rosa Jr.

Thank you!
Questions?

vnpossani@inf.ufpel.edu.br