co-LODEX: A Cooperative Logic Design Expert System

ABSTRACT

CAD systems that can quickly produce quality designs are needed for the expanding VLSI market. co-LODEX accepts constraints on area and speed, and outputs a CMOS standard cell netlist that satisfies the constraints. It can even obtain an exact optimal circuit for area or speed. Short turnaround is attained through the combination of parallel processing by several processors and their cooperation.

KEY FEATURES

Global optimization

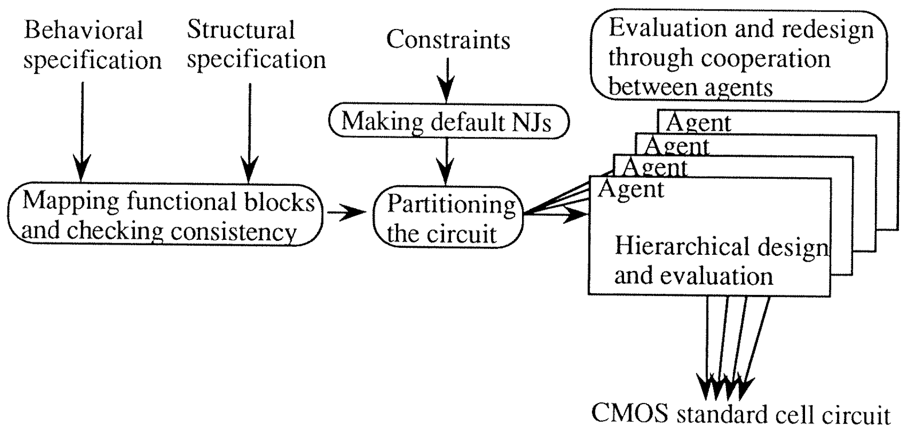
An exact optimal circuit for area or speed can be obtained by iteratively strengthening the corresponding constraint.

Evaluation-redesign mechanism

Sufficient conditions for constraint violation, nogood justifications (NJs), are used.

Cooperative design mechanism

Design agents exchange design results (in case of success) or NJs (in case of failure).



Overview of co-LODEX

OVERVIEW

The user specifies a behavioral specification, a structural specification, and constraints on area and speed. Figure 1 shows an example. A behavioral specification (upper right) is specified in a hardware description language. A structural specification (left) is specified in terms of a block diagram of the datapath. Constraints are expressed as inequalities in the gate count or propagation delay and transformed into opposite inequalities, or default NJs (lower right). One constraint on speed is highlighted in each window, the corresponding path (left), the corresponding operation (upper right), and the corresponding default NJ (lower right).

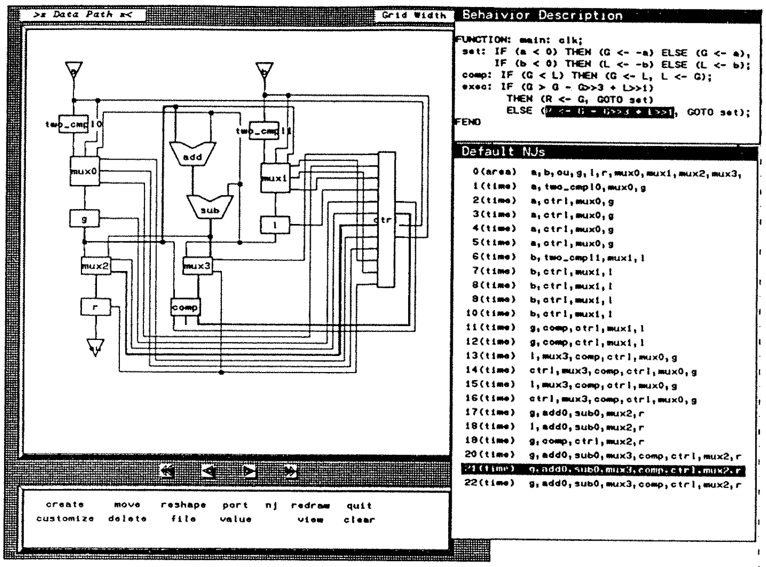


Figure 1. Inputs to co-LODEX

co-LODEX divides the whole circuit to be designed into subcircuits. Each subcircuit is designed by a design agent. co-LODEX divides the circuit so that the blocks along critical path candidates are distributed to as few agents as possible. It is likely that agents along a critical path candidate need a considerable amount of mutual communication since agents sharing a constraint must communicate with each other.

co-LODEX outputs a CMOS standard cell netlist that satisfies the constraints. The resulting netlist can be input to an automatic place-and-route system for CMOS standard cells.

COOPERATIVE DESIGN MECHANISM

A subcircuit is designed hierarchically within an agent. Figure 2 shows a snapshot. An adder of a subtracter (upper right) is about to be implemented with standard cells. There are three alternatives, from upper left to lower right.

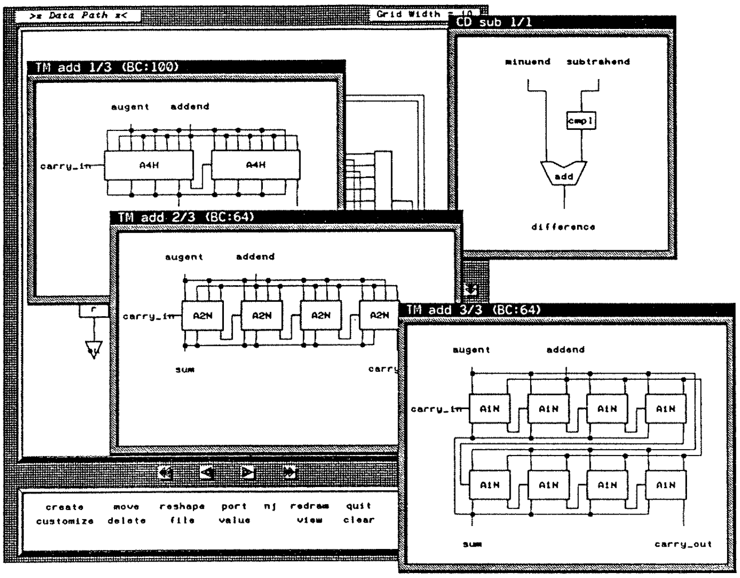


Figure 2. Hierarchical design

The evaluation-redesign mechanism within each agent uses sufficient conditions for constraint violation, nogood justifications (NJs). Satisfying an NJ means a constraint violation and invokes the mechanism. See [Minoda 1992] for further details.

Global evaluation-redesign takes place by agents exchanging design results (in case of success) or NJs (in case of failure). In our cooperative design mechanism, NJs received from other agents help narrow down the search space for an agent in the sense that NJs made out of the received ones enable the agent to prune the search space. That is the reason why we claim co-LODEX as "cooperative". The APPENDIX shows the flowchart of the cooperation algorithm.

EXPERIMENTAL RESULTS

We implemented co-LODEX on Multi-PSI in KL1. Experimental results show that (1) co-LODEX can efficiently carry out global optimization. (2) The best linear speedup has been observed by increasing the number of agents to 15.

OUTLINE OF DEMONSTRATION

We show how co-LODEX works taking as an example the circuit shown in Figure 1 and Figure 2. We have prepared a special interface based on a log of an actual run for illustrating how agents cooperate. Please refer to the APPENDIX for the details of the cooperation algorithm.

Then we demonstrate an actual run of co-LODEX for a large circuit example.

REFERENCE

[Minoda 1992] Y. Minoda, et al. "A Cooperative Logic Design Expert System on a Multiprocessor," Proc. of FGCS'92, pp.1181-1189 (1992).

APPENDIX (Cooperation algorithm)