Constraint-driven Design -
The Next Step Towards Analog Design Automation

Invited Talk

Göran Jerke
Robert Bosch GmbH, AE/EIM
Reutlingen, Germany
Email: Goeran.Jerke@ieee.org

Jens Lienig
Dresden University of Technology, IFTE
Dresden, Germany
Email: jens@ieee.org
Motivation

Evolution of Analog IC Design

Verification of...

Schematic

Technology

DRC

LVS

Polygon Pushing

Schematic-driven Layout

Constraint-driven Design

Analog Design Synthesis

1980 1990 2000 2010 today
Contents

- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- Impact on design algorithms and design flow
- Open problems
- Summary and conclusion
Contents

- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- Impact on design algorithms and design flow
- Open problems
- Summary and conclusion
The Verification Gap

Constraint Classification

**Primary Constraints**

- Technology Constraints (manufacturing)
  - min. wire width, spacing, overlap

- Functional Constraints (circuit function)
  - max. IR-drop between two net terminals, device matching, ...

**Secondary Constraints**

- Design-Methodical Constraints (design complexity)
  - Design hierarchy, routing directions, standard cells

- Economic Constraints (cost, TTM)
  - Chip count, development costs and chip area determine IC technology
The Verification Gap

Manufacturability

Technology Constraints

1

Layout Rules

(Meta layer)

2

DRC

Verification gap

Dummy errors

EDA-tools guarantee manufacturability!
The Verification Gap

Functionality

Functional Constraints (Expert knowledge)

Unrepresentable expert knowledge

Representable, but non-verifiable knowledge (schematic prosa, symmetries, …)

Schematic (Meta layer)

LVS

Devices, parameters, nets

EDA-tools do not (yet) guarantee circuit functionality!
The Verification Gap

Evolution of Analog IC Design

Verification of...

Constraints / Expert Knowledge

Schematic

Technology

DRC

LVS

Constraint Verification

Polygon Pushing

Schematic-driven Layout

Constraint-driven Design

Analog Design Synthesis

Contents

- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- Impact on design algorithms and design flow
- Open problems
- Summary and conclusion
Current Approaches for Constraint Consideration

Constraint-Consideration during Schematic Design

Constraints as Schematic „Prosa“

+ Man. consideration of “complex” constraints
  - No autom. constraint verification possible

2nd Gen. Constraint Management

+ Constraints are part of the database
  - No “complex” constraints (yet)
Current Approaches for Constraint Consideration

Constraint-Consideration during Physical Design

„Atomic“ Module Approach

Characteristics:
- Individual design objects (→ transistors, resistors, capacitors, etc.) and constraints are considered (semi-) automatically
- Constraint assignment and management is required
- Design algorithms must “understand” all constraints

+ Full flexibility for layout optimization
- Missing constraints result in wrong layouts
- Long run-times of layout generation tools

- Layout variant 1
- Layout variant 2
Current Approaches for Constraint Consideration

Constraint-Consideration during Physical Design

„Molecular“ Module Approach

Characteristics:

- Several design objects are combined to a hierarchical PCell module
- Constraints will be fulfilled automatically by the PCell module
- High-level re-use of design knowledge

+ Manual consideration of any constraint
+ Very fast constraint-driven layout generation
- Additional constraints require new PCell module
- Limited freedom for design optimization
- Complexity of rel. PCell verification problem: $O(m^n)$

($m$ - number of parameters, $n$ - number of variants per parameter)
Contents

→ The verification gap
→ Current approaches for constraint consideration
→ The constraint-driven design flow
→ Impact on design algorithms and design flow
→ Open problems
→ Summary and conclusion
The Constraint-driven Design Flow

Constraint Representation

- Formalize constraints!
- Define all constraints explicitly!
- Account for design style!
The Constraint-driven Design Flow

Simple and Complex Constraints

Simple constraint examples:

\[ V_{IR}(\text{Pad} \rightarrow \text{T2}) < 0.1 \, \text{V} \]

Voltage class (Pad) = \{50V, 80V\}
Simple and Complex Constraints

Complex constraint example (*independent constraints*):

```
if ( net type == P&G ) then
    [[Pad->T1], [Pad->T2], [Pad->T3]] must have star-shaped net topology
    &&
    R(Pad->T1) < 1Ω && R(Pad->T2) < 1Ω && R(Pad->T3) < 1Ω !
```
The Constraint-driven Design Flow

- Simulation
- Circuit Design
- Floorplanning
- Device Generation
- Placement
- Routing
- Compaction
- Verification
- Manufacturing
- Test

Constraint Management

- Constraint Data
- Constraint Derivation
- Constraint Templates
- Constraint Transformation
- Constraint Sensitivity Analysis (CSA)
- Constraint Verification
- Verification Rules

Start
The Constraint-driven Design Flow

Constraint Management (Data Consistency)

Today: Separate design and constraint databases

- Difficult design and constraint data management (data consistency, data versioning)

Design Data

Constraint Data

DO1

C1

DO2

Future

DO1

C1

DO2

V1.0

DO3

V2.0

Design and Constraint Database

DOx - Design Object

Cy - Constraint
The Constraint-driven Design Flow

Constraint Management (Propagation)

Top-Down Propagation

Examples:
- Floorplanning constraints
- IR-drop constraints

Bottom-Up Propagation

Examples:
- Placement constraints
- Routing blockages

Top-Down and Bottom-Up Propagation

Examples:
- ESD path definition
- Net shielding
Constraint Derivation Methods

- Expert knowledge

- Direct derivation rules and templates
  - Example:
    ```
    if ( differential pair ) then
    Assign matching constraint to transistor pair
    ```

- Deduction processes
  - Example:
    ```
    Net N1 is connected to 40V IO pad &&
    I1 is connected to net N1
    ⇒ I1 is connected to 40V IO pad
    → Assign 40V design constraints to I1
    ```

- Indirect method (transformation)
The Constraint-driven Design Flow

Constraint Transformation

Definition: Consistent and unambiguous transformation of high-level constraints into low-level constraints

1. Transformation of electrical constraints into circuit-specific constraints
   - Max. IR-Drop [V]

2. Transformation of circuit-specific constraints into layout-specific constraints
   - Max. Resistance [Ohm]

3. Assignment of layout-specific constraints to (geometrical) design parameters
   - Wire length, -width layer …
The Constraint-driven Design Flow

Constraint Sensitivity Analysis (CSA)

Definition: Context-dependent sensitivity and gap determination of design parameters under consideration of one or more constraints

Coupled constraints: \( V_{IR} < V_{IR\text{-max}}; \quad RC < (RC)_{\text{max}}; \quad MTTF > MTTF_{\text{max}} \)

\[
R = \rho_{\text{Conductor}} \cdot \frac{l}{w \cdot h} \cdot (1 + TK_1 \cdot (T_0 - \Delta T))
\]

\[
C = \epsilon_0 \cdot \epsilon_r \cdot l \left[ 1.15 \cdot \frac{w}{d_{\text{ox}}} + 2.80 \cdot \left( \frac{h}{d_{\text{ox}}} \right)^{0.222} \right]
\]

\[
MTTF = A \cdot \left( \frac{w \cdot h}{i} \right)^n \cdot \exp \left( \frac{E_a}{k \cdot (T_0 - \Delta T)} \right)
\]

\[
V_{IR} = i \cdot R
\]
Constraint Sensitivity Analysis (CSA)

Possible constraint violation!

Sensitivity of $w$ with respect to $V_{IR}$

Sensitivity of $w$ and $\Delta T$ with respect to $V_{IR}$

$\Delta T > T_1 = \text{const.}$

$\Delta T = T_1 = \text{const.}$

$\Delta T < T_1 = \text{const.}$
1. Constraints are not formalized

The voltage class of each well connected to a supply line should match the voltage range occurring at the pad during operation!

2. Formalize constraints and verification task

For all power and ground pads:
- Get voltage class $V_{PAD}$ of pad
For all net terminals of the active net:
- Get voltage class $V_{Inst}$ of owning instance
If $V_{Inst} \neq V_{PAD}$ then Return ERROR
Return SUCCESS

3. Define verification requirements

3.1 Specify and implement verification routine(s)

4. Verify constraint fulfillment

- Manual (4+n eyes verification)
- Automatical verification
The Constraint-driven Design Flow

Constraint Verification (Example)

ESD-Verification

Netlist

Layout

DRC / LVS

Symmetry Verification

IR-drop Verification

Reliability Verification

Circuit Verification

Property Verification

Sub-circuit recognition

Circuit simulation

Instance property retrieval

Terminal current retrieval

Layout polygon extraction

EOS reliability calculation

Layout topology recognition

Resistance calculation

Combine capabilities of several tools to define and perform verification tasks!
The Constraint-driven Design Flow

Tool A
Functionality:
A1, A2, A3, ..., An

Tool B
Functionality:
B1, B2, B3, ..., Bm

Constraint data
Design data
Tool functionality

Constraint-Engineering System (CES)

CLP-based Logic Core & Constraint Solver

TIK_A, TIK_B, TIK_X, TIK_Y

Constraint Solver
Constraint Derivation
Constraint Transformation
Constraint Sensitivity Analysis (CSA)
Constraint Verification

CLP – Constraint Logic Programming
TIK – Tool Integration Kit

Constraint Templates
Transform. Models
Verification Rules
The Constraint-driven Design Flow

Constraint Verification

Example:
Check all pin-to-pin resistances $R_{C,Pn}$ in star-shaped nets: $R_{C,Pn} \leq R_{\text{max}}$.

CES query for $R_{\text{max}} = 5 \, \Omega$: $\text{valStarRes}(N, P, 5)$.

Result: List of all violating combinations of nets and terminals
Contents

- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- Impact on design algorithms and design flow
- Open problems
- Summary and conclusion
Impact on Design Algorithms and Design Flow

Sequential Design Flow

Continuous Design Flow [5]

Design Steps

Degree of Design Freedom

Time

Future

Physical Realization

Degree of Design Freedom

Time

Physical Realization

Floorplanning

Device Generation

Placement

Routing

Compaction

Iterations
Impact on Design Algorithms and Design Flow

High-Level Design Algorithms

Design Problem

**Algorithm 1**
- IFC

**Algorithm 2**
- IFC

**Algorithm 3**
- IFC

... to **Algorithm n**
- IFC

Design and Constraint Database

**Strategy A:**
1. Placement \(\rightarrow\) Alg. 1
2. Glob. Routing \(\rightarrow\) Alg. 2
3. Det. Routing \(\rightarrow\) Alg. 6

**Strategy B:**
1. Route Planning \(\rightarrow\) Alg. 4
2. Placement \(\rightarrow\) Alg. 9
3. Det. Routing \(\rightarrow\) Alg. 6
Contents

- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- Impact on design algorithms and design flow
- Open problems
- Summary and conclusion
Open Problems (Long Term)

- Constraint solver:
  - Consideration of constraints with statistical boundaries is required

- Constraint methods:
  - Scalability of constraint sensitivity analysis (CSA) must be improved
  - Approaches for automatic constraint rule optimization should be developed

- High-level design algorithms:
  - Improvement of concepts for abstraction of design and verification algorithms
  - Development of strategies for high-level design task partitioning (with CSA)
Summary and Conclusion

- Presentation covered (1) today’s verification gap, (2) current and future approaches for constraint-driven design and (3) open problems.

- Constraint-driven design is a major and a necessary step towards a fully-automated analog design synthesis.

- Constraint verification reduces the existing verification gap in A/MS designs.

- The comprehensive and automatic constraint consideration is a potentially disruptive technology with a very strong impact on the design process!

- Constraint-driven X-design → interdisciplinary field with a tremendous potential and many challenging problems.
Thank You!
Bibliography


