

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

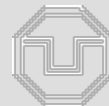
Invited Talk

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



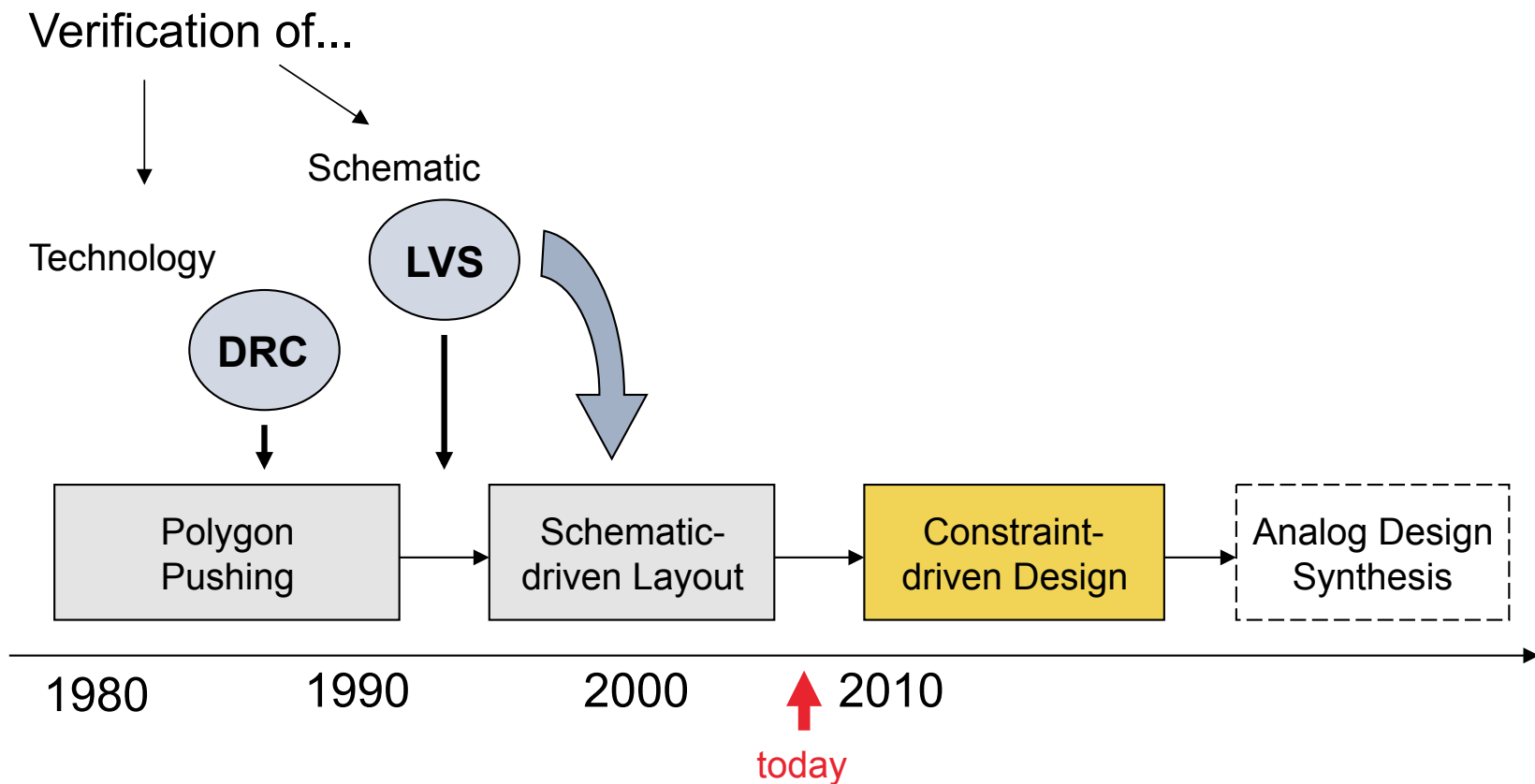
BOSCH



ifte

Motivation

Evolution of Analog IC Design



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

Technology Constraints (manufacturing)

→ min. wire width, spacing, overlap

Functional Constraints (circuit function)

→ max. IR-drop between two net terminals, device matching, ...

Primary 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

Secondary Constraints

The Verification Gap

Manufacturability

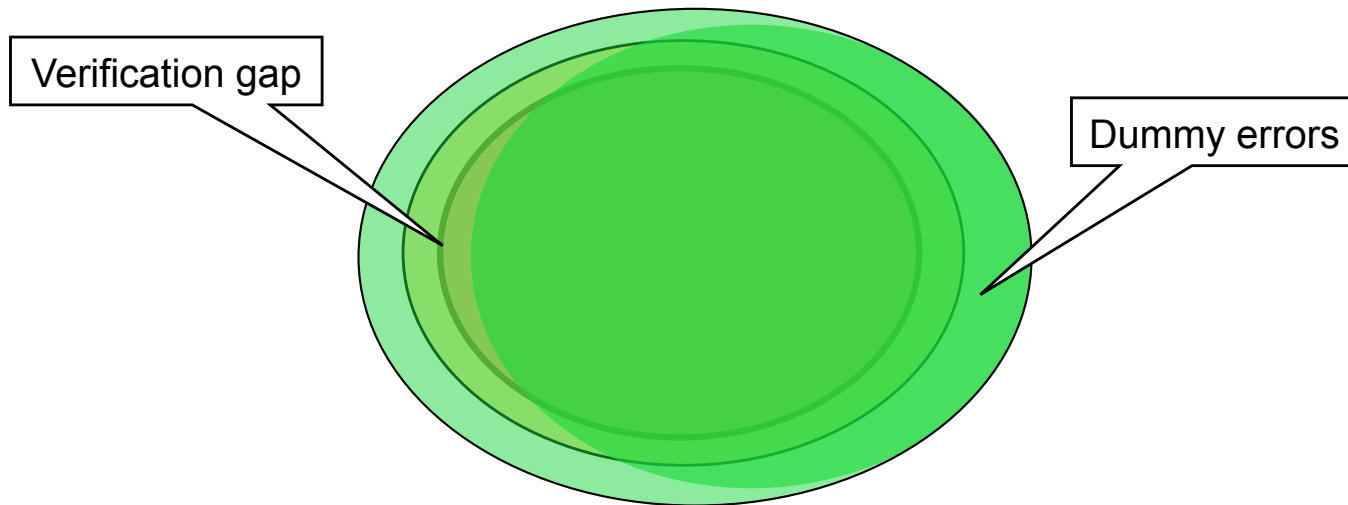
Technology Constraints



Layout Rules
(Meta layer)



DRC



EDA-tools guarantee manufacturability!

The Verification Gap

Functionality

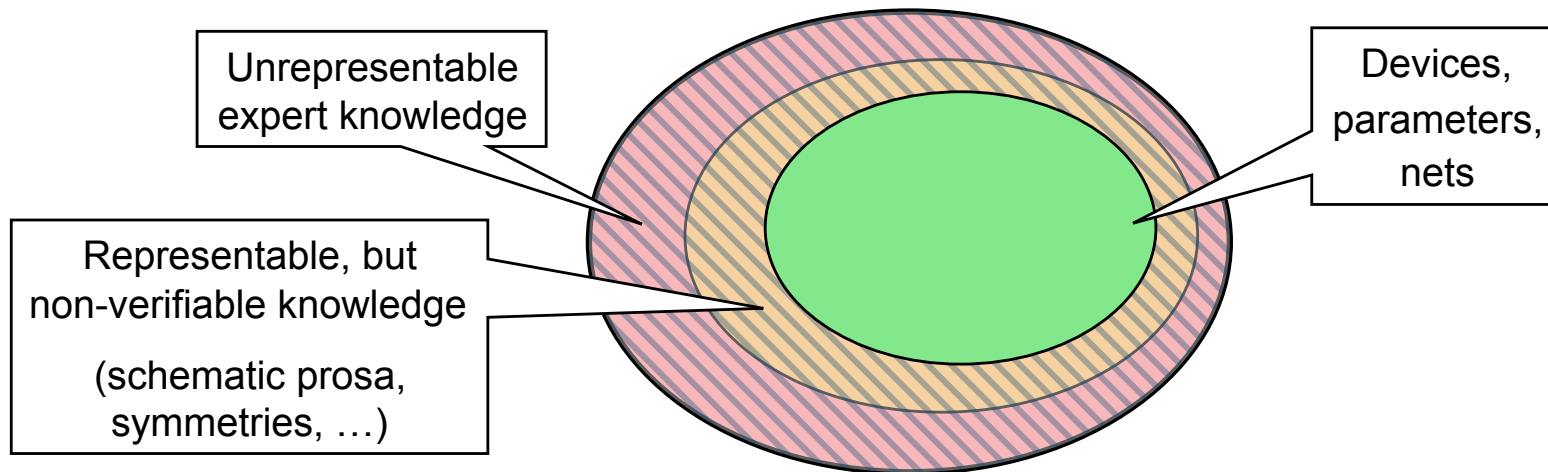
Functional Constraints
(Expert knowledge)



Schematic
(Meta layer)



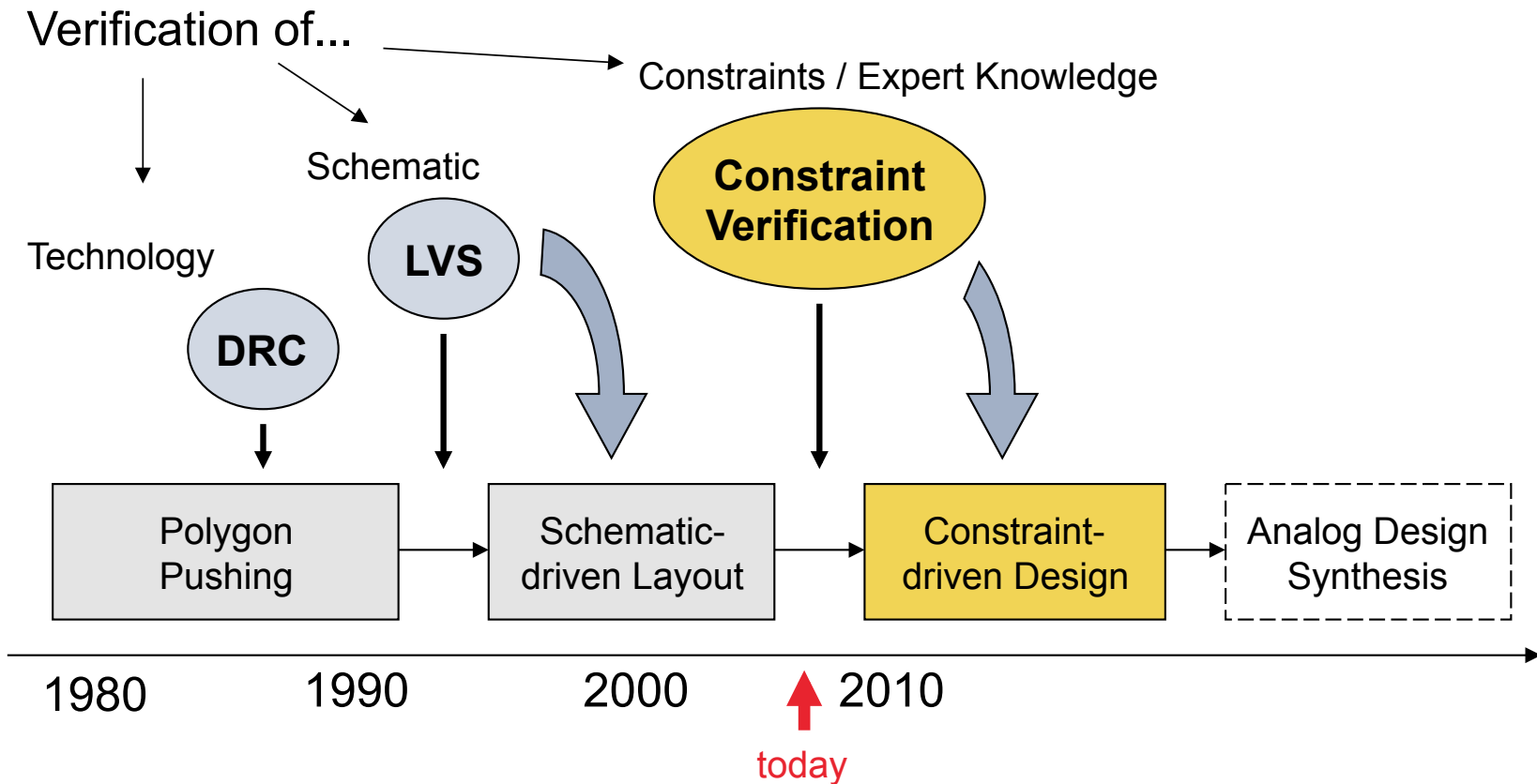
LVS



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

The Verification Gap

Evolution of Analog IC Design



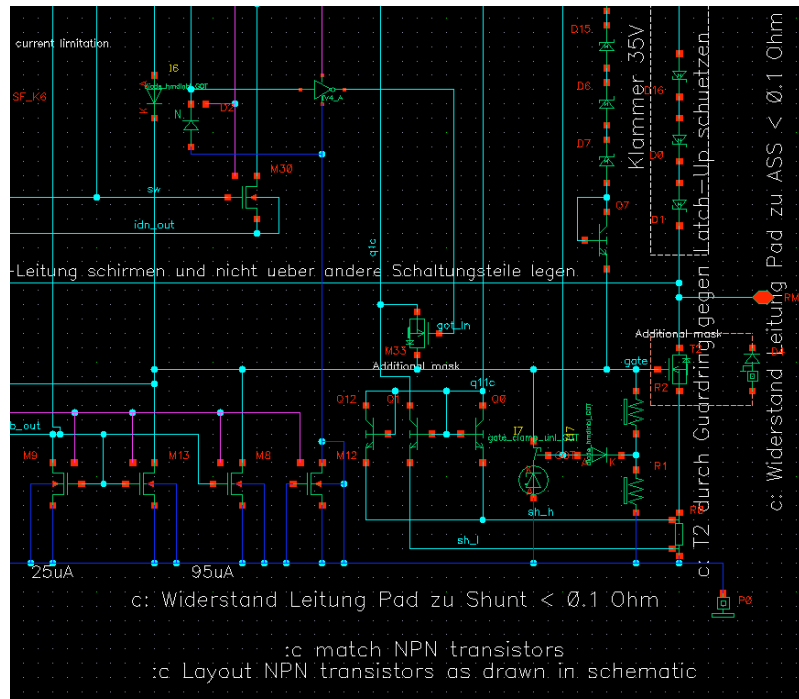
Contents

- 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“



2nd Gen. Constraint Management

Name	Constr_6
Owner	testlib_gjerke.test1...
Axis	horizontal (default)
Enabled	true
Status	enforced
Notes	
Side	bottom
PreserveCr...	false
pitch	(not set)
use	boundary
layerName	OVERLAP

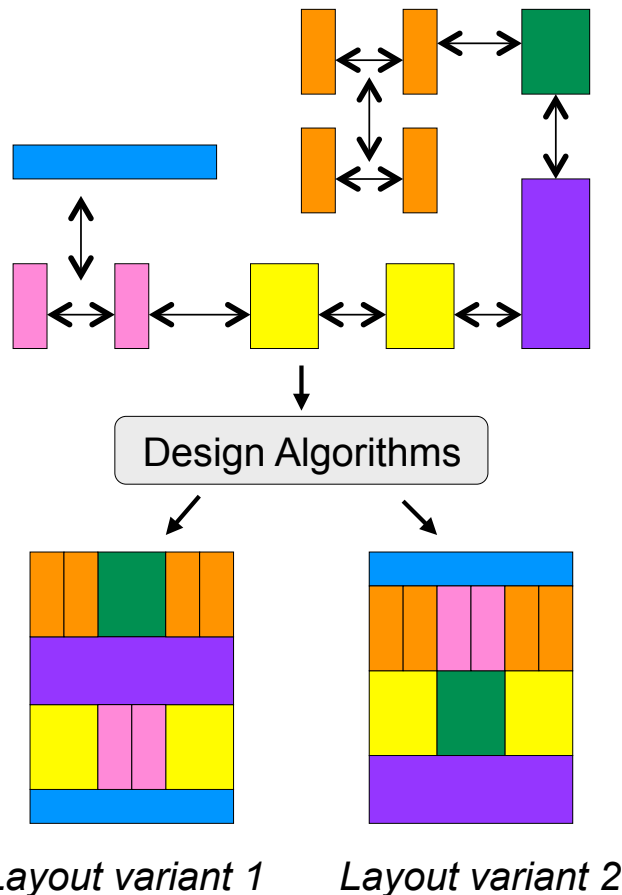
- + Man. consideration of “complex” constraints
- No autom. constraint verification possible

- + Constraints are part of the database
- No “complex” constraints (yet)

Current Approaches for Constraint Consideration

Constraint-Consideration during Physical Design

„Atomic“ Module Approach



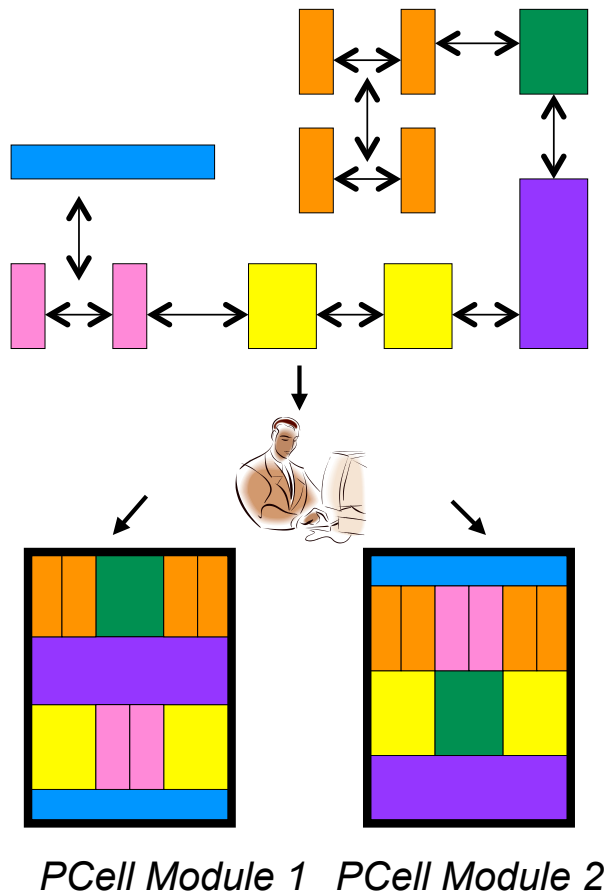
Characteristics:

- 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

Current Approaches for Constraint Consideration

Constraint-Consideration during Physical Design

„Molecular“ Module Approach



Characteristics:

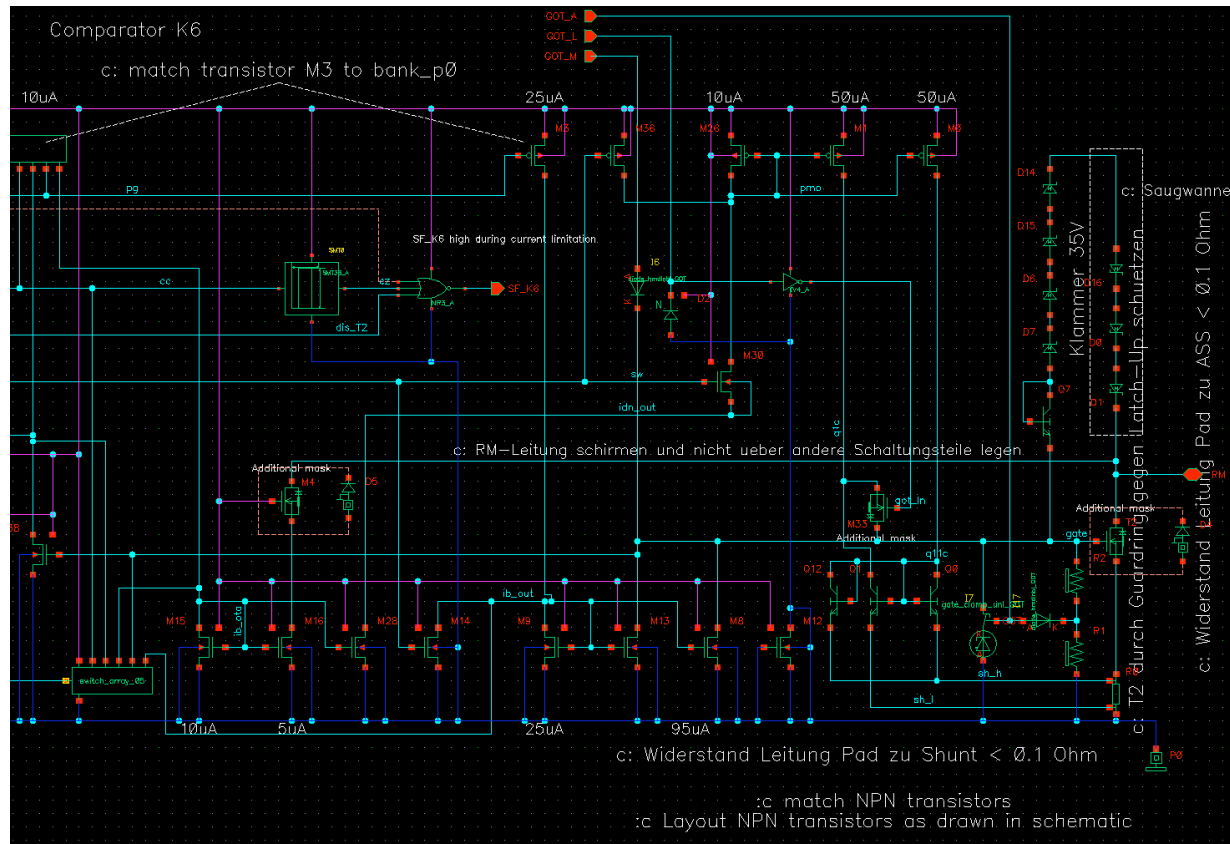
- Several design objects are combined to a hierarchical PCell module
 - Constraints will be fulfilled automatically by the PCell module
 - High-level re-use of design knowledge
-
- + Manual consideration of any constraint
 - + Very fast constraint-driven layout generation
 - Additional constraints require 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

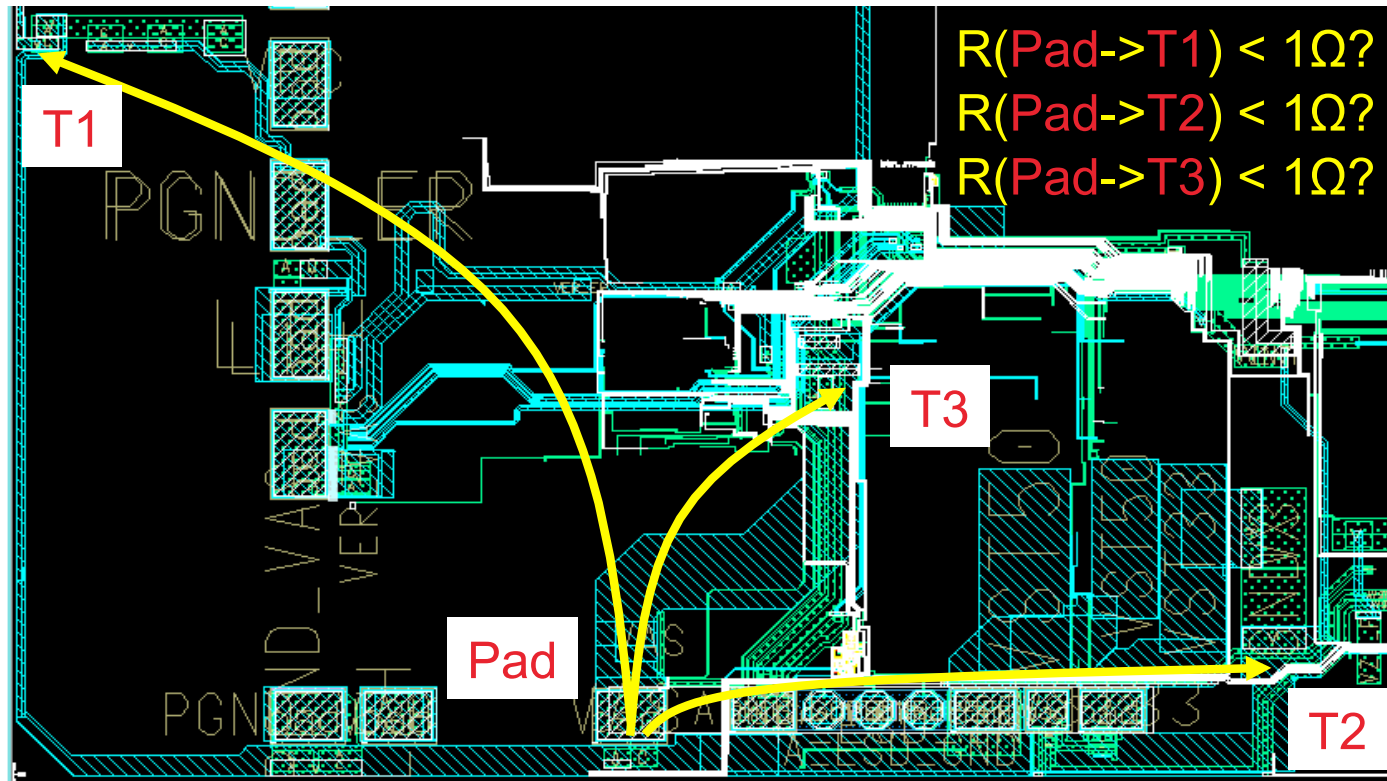
The Constraint-driven Design Flow

Constraint Representation



The Constraint-driven Design Flow

Simple and Complex Constraints



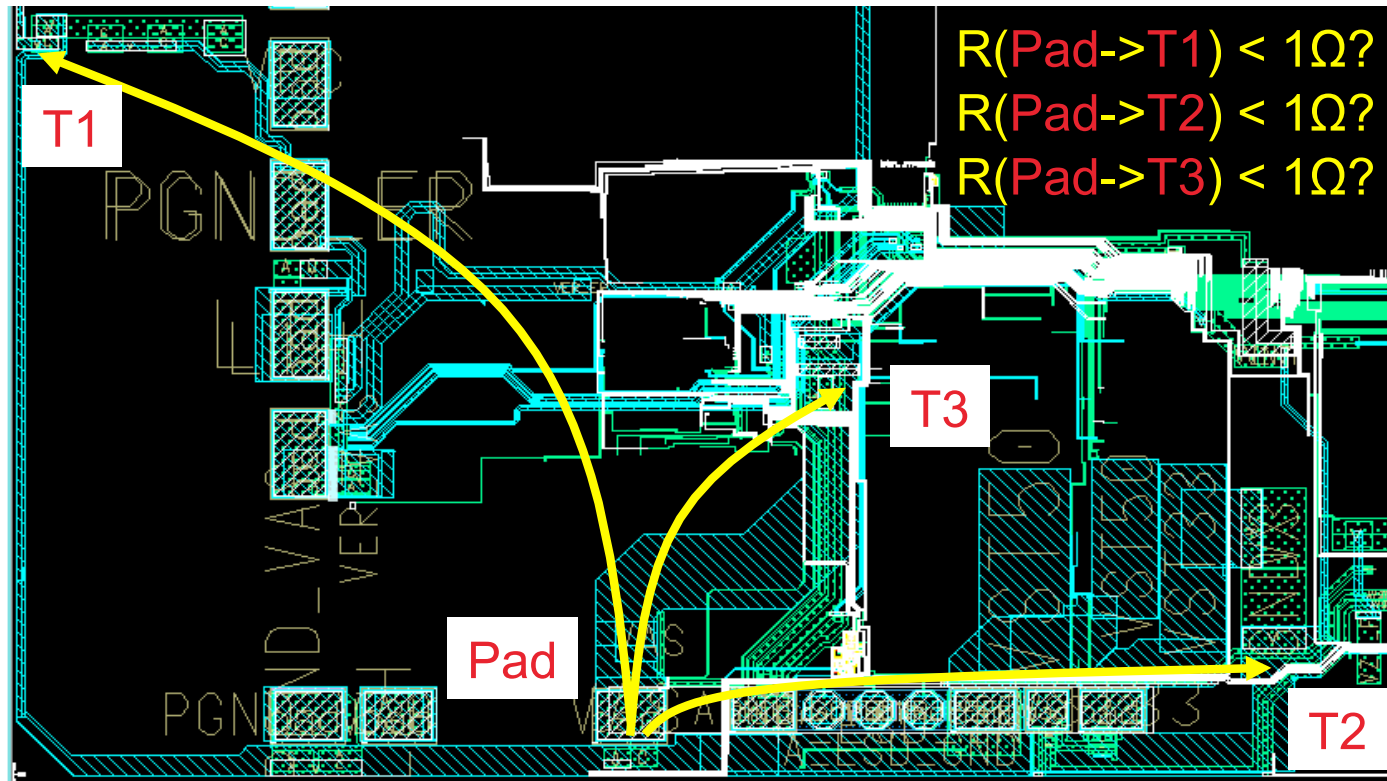
Simple constraint examples:

$$V_{IR}(\text{Pad} \rightarrow T2) < 0.1 \text{ V}$$

Voltage class (Pad) = {50V, 80V}

The Constraint-driven Design Flow

Simple and Complex Constraints



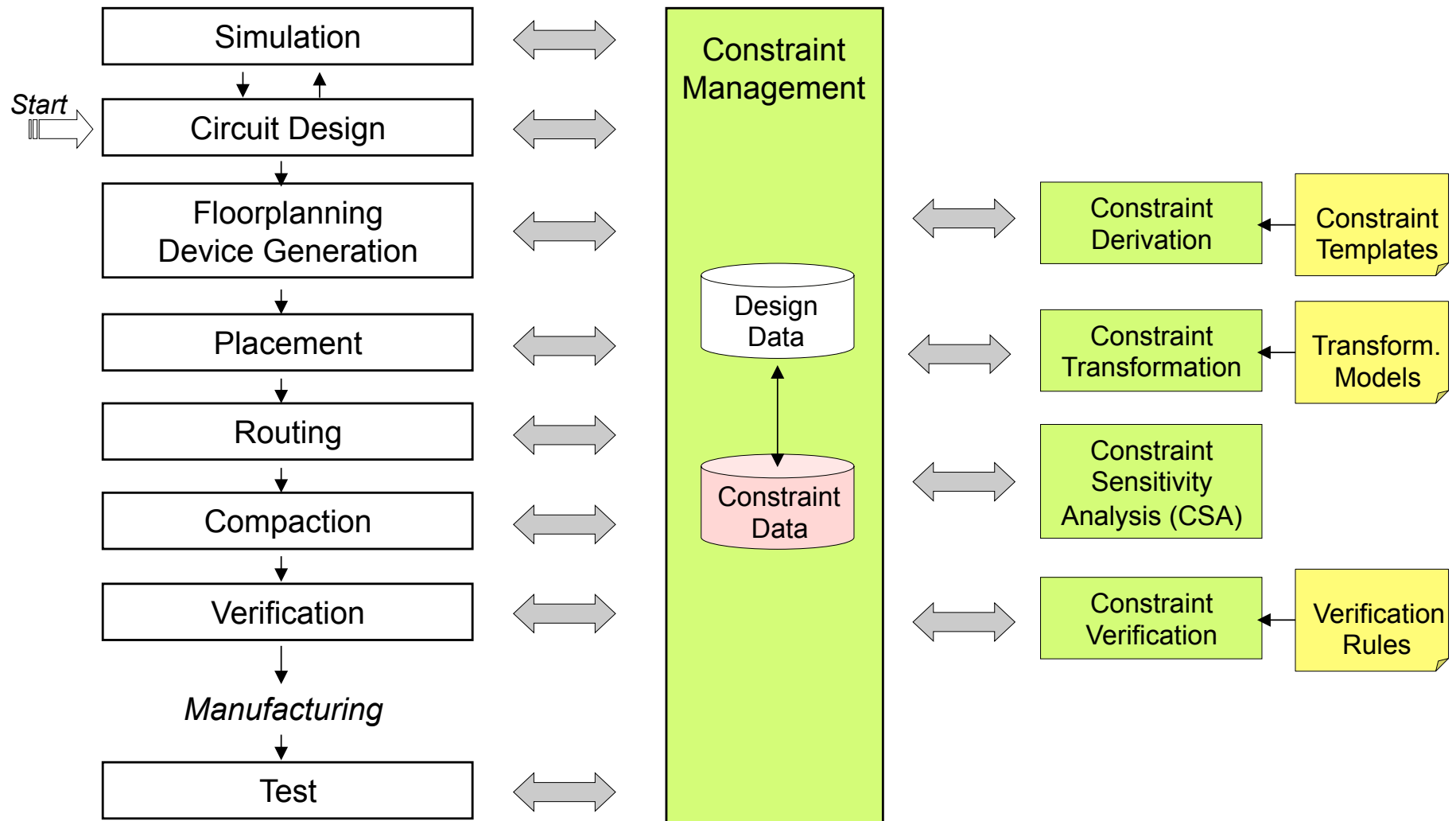
Complex constraint example (*independent constraints*):

```
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 !
```

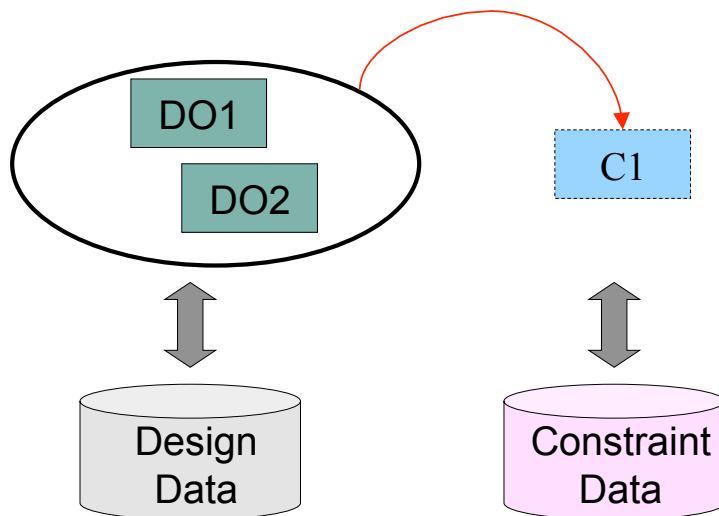
The Constraint-driven Design Flow



The Constraint-driven Design Flow

Constraint Management (Data Consistency)

Today: Separate design and constraint databases

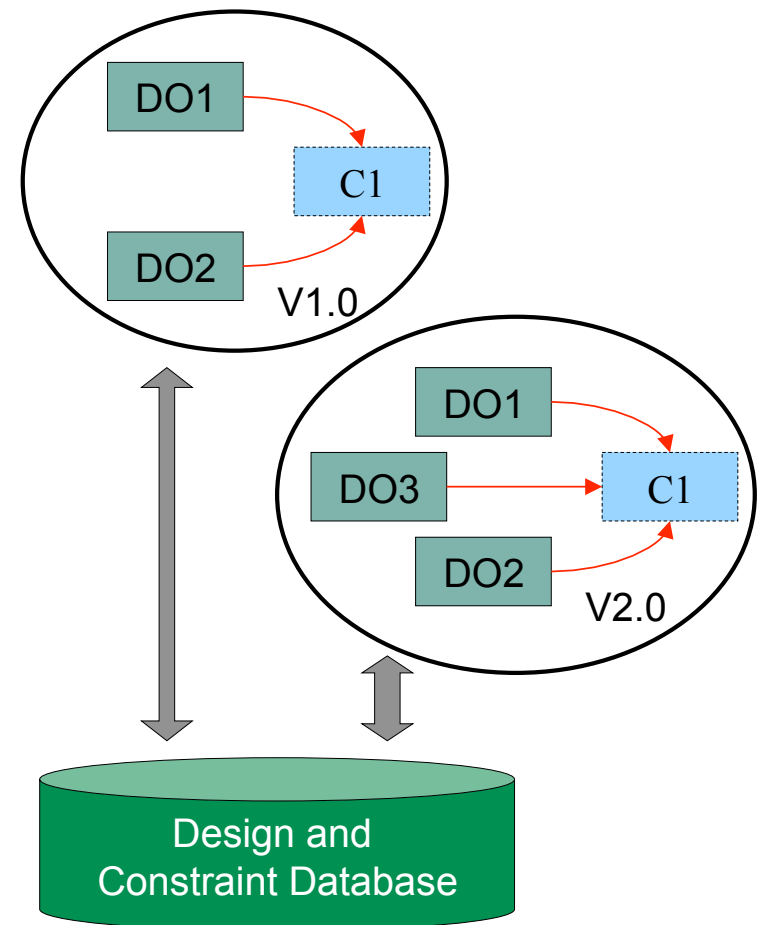


- Difficult design and constraint data management (data consistency, data versioning)

C_y - Constraint

DO_x - Design Object

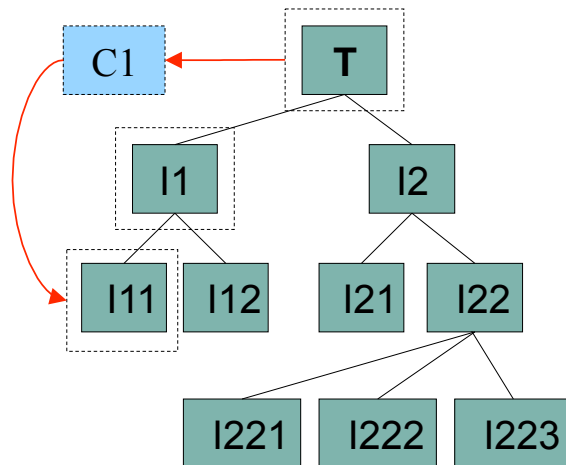
Future



The Constraint-driven Design Flow

Constraint Management (Propagation)

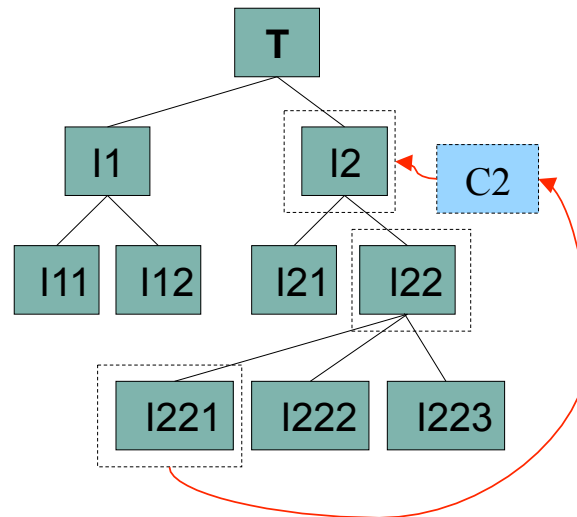
Top-Down Propagation



Examples:

- Floorplanning constraints
- IR-drop constraints

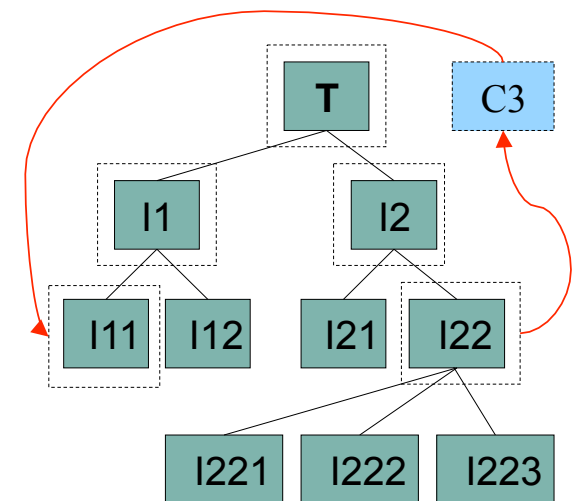
Bottom-Up Propagation



Examples:

- Placement constraints
- Routing blockages

Top-Down and Bottom-Up Propagation

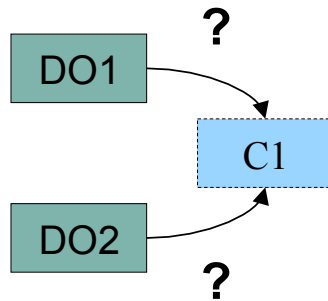


Examples:

- ESD path definition
- Net shielding

The Constraint-driven Design Flow

Constraint Derivation Methods



C_y - Constraint

DO_x - Design Object

→ 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
2. Transformation of circuit-specific constraints into layout-specific constraints
3. Assignment of layout-specific constraints to (geometrical) design parameters



Example: IR-Drop

Max. IR-Drop [V]

Max. Resistance [Ohm]

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

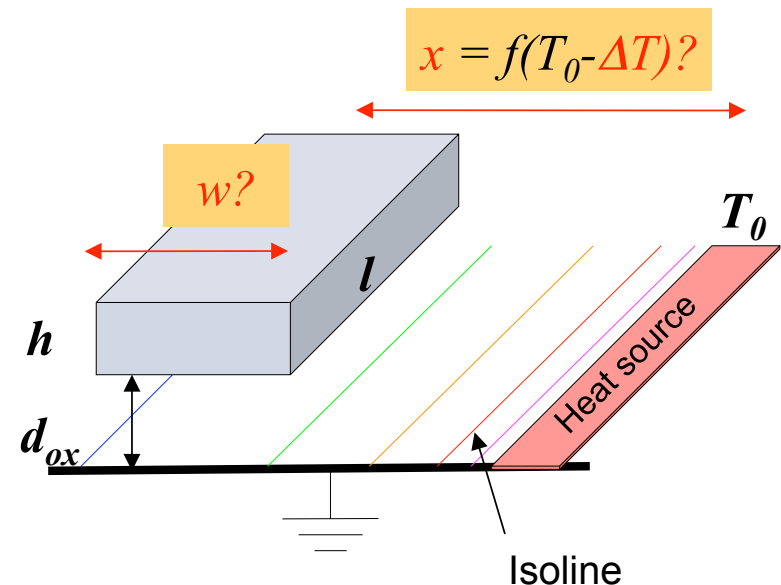
Coupled constraints: $V_{IR} < V_{IR-max}$; $RC < (RC)_{max}$; $MTTF > MTTF_{max}$

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

$$C = \varepsilon_0 \cdot \varepsilon_r \cdot l \cdot \left[1.15 \cdot \frac{w}{d_{ox}} + 2.80 \cdot \left(\frac{h}{d_{ox}} \right)^{0.222} \right]$$

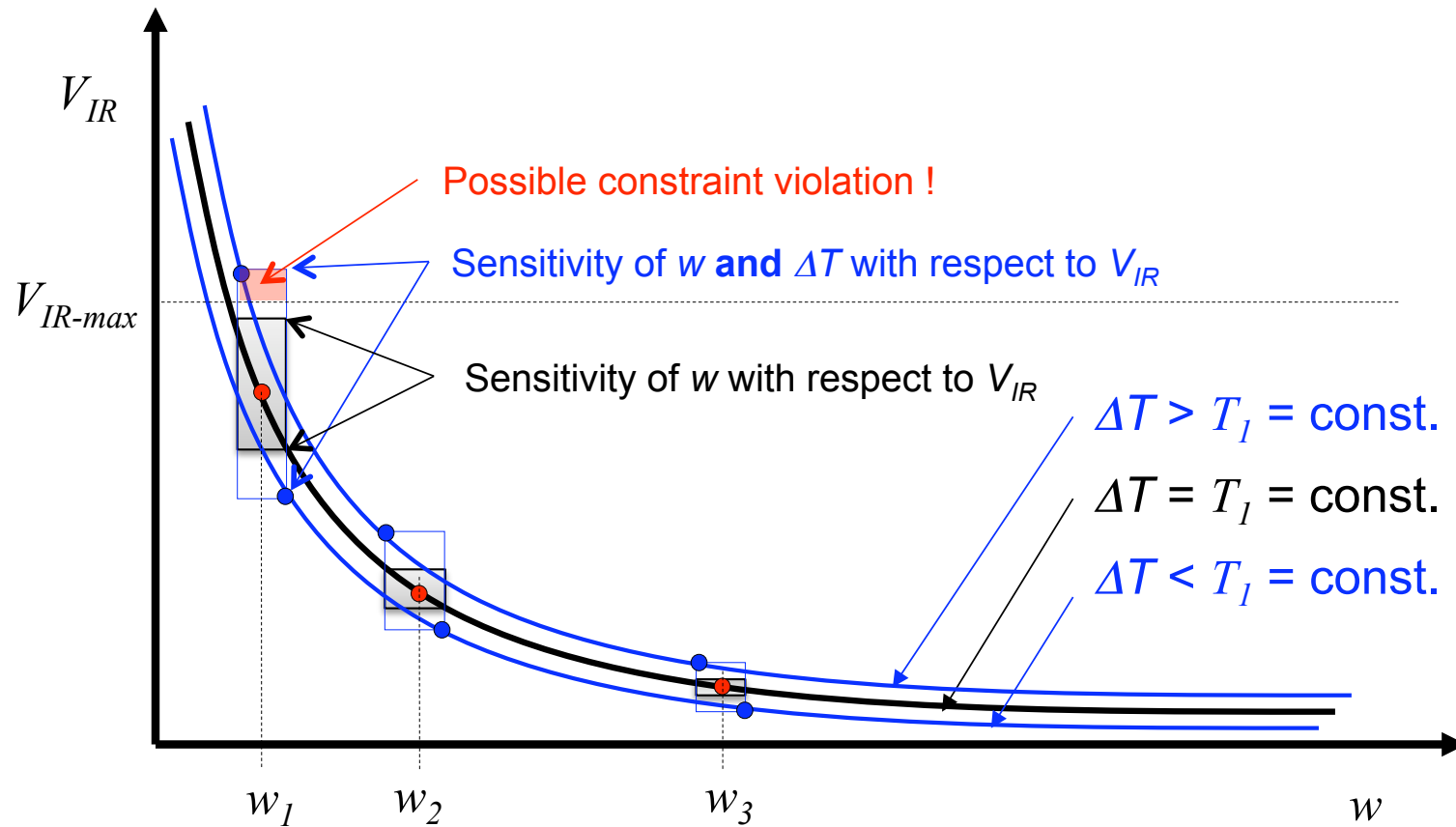
$$MTTF = A \cdot \left(\frac{w \cdot h}{i} \right)^n \cdot \exp\left(\frac{E_a}{k \cdot (T_0 - \Delta T)} \right)$$

$$V_{IR} = i \cdot R$$



The Constraint-driven Design Flow

Constraint Sensitivity Analysis (CSA)



The Constraint-driven Design Flow

Constraint Verification



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
```

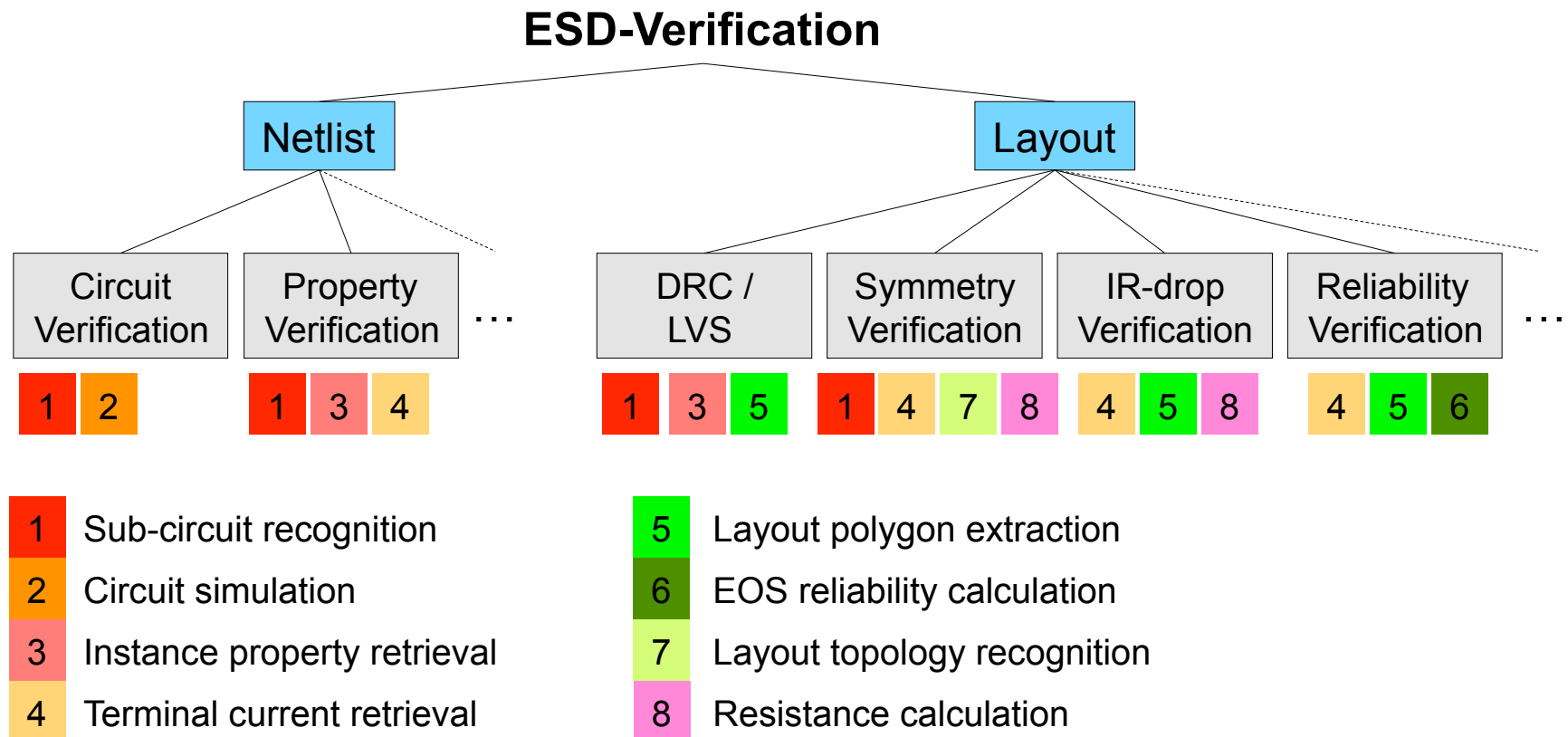
- 3.1 Define verification requirements
- 3.2 Specify and implement verification routine(s)



4. Verify constraint fulfillment
 - Manual (4+n eyes verification)
 - Automatic verification

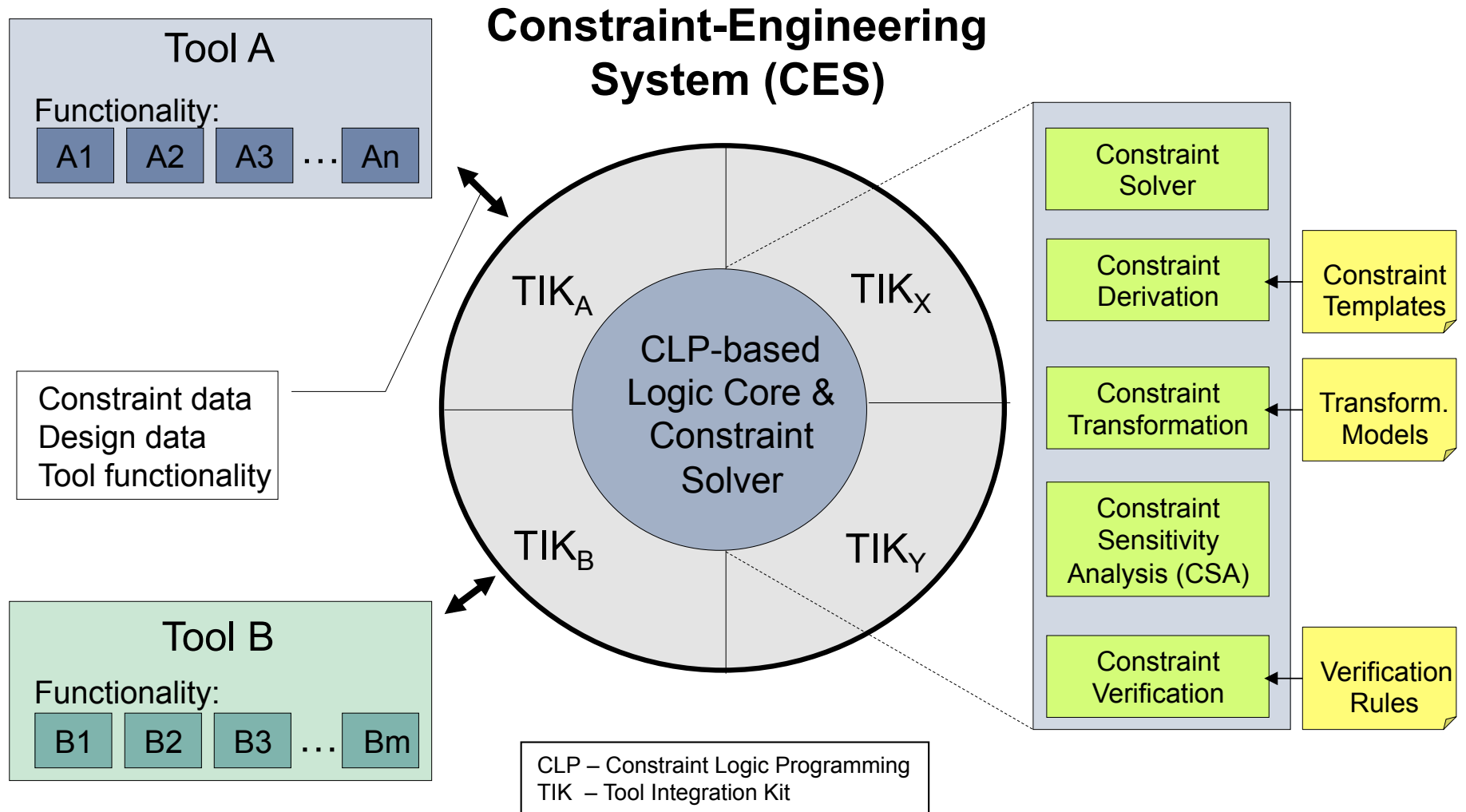
The Constraint-driven Design Flow

Constraint Verification (Example)



Combine capabilities of several tools to define and perform verification tasks !

The Constraint-driven Design Flow

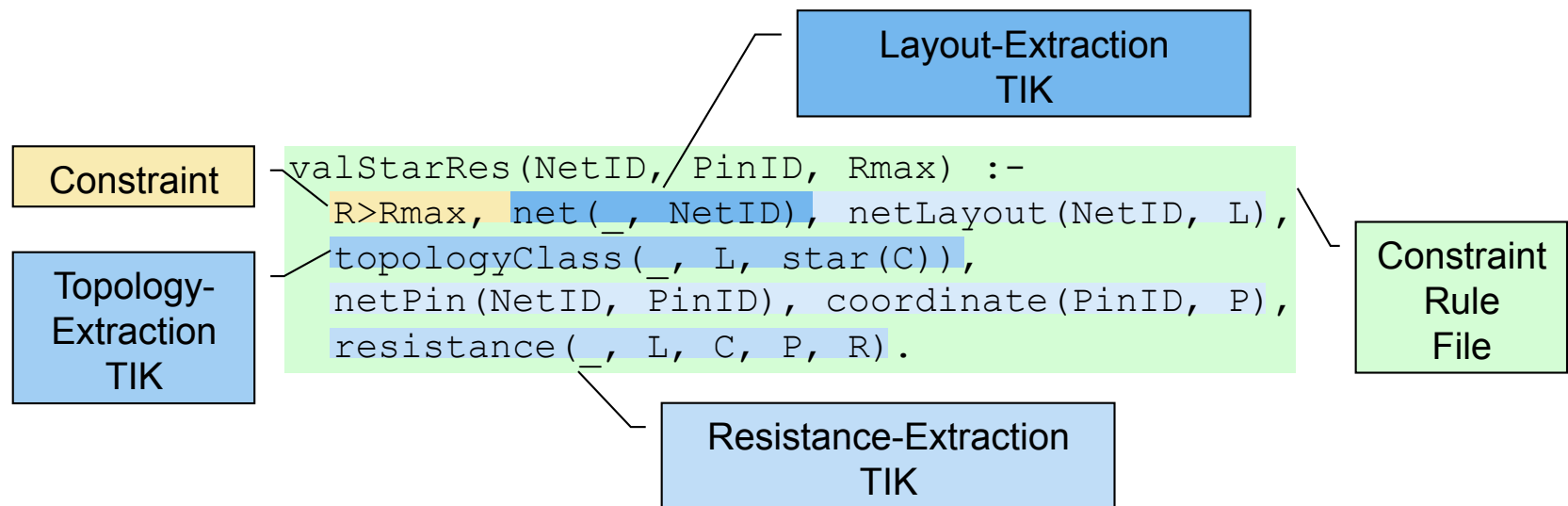


The Constraint-driven Design Flow

Constraint Verification

→ Example:

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



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

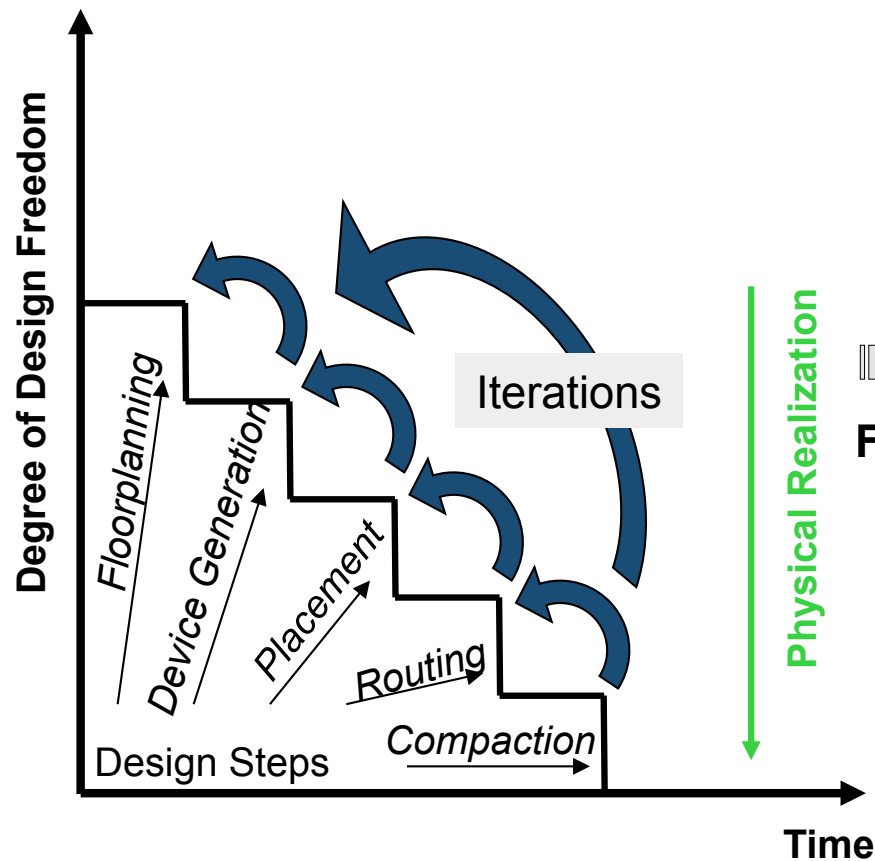
Result: List of all violating combinations of nets and terminals

Contents

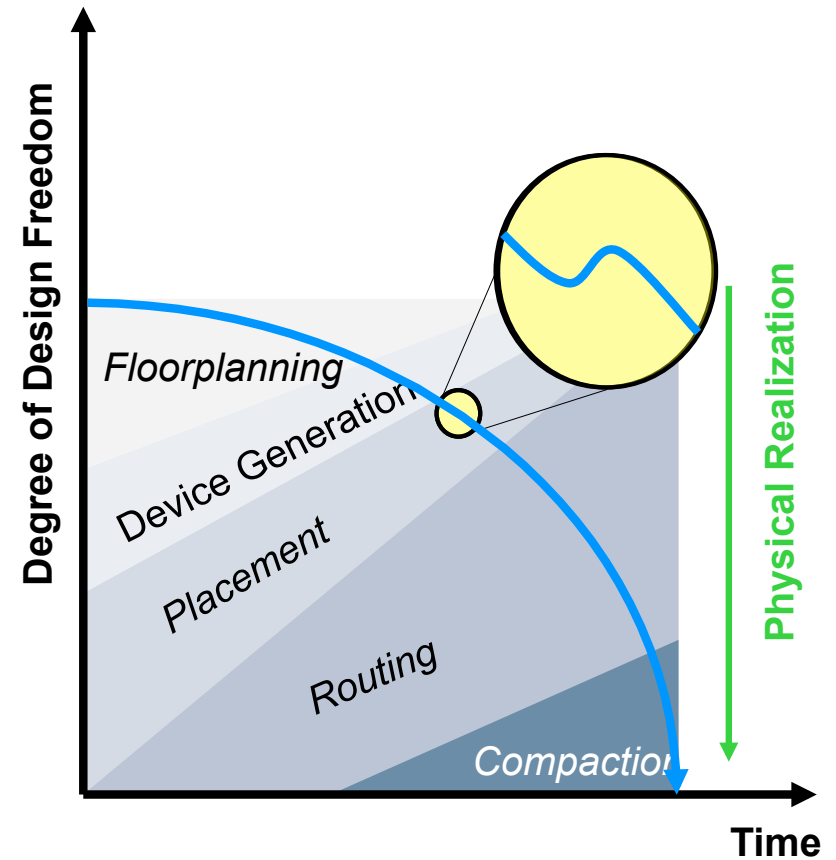
- The verification gap
- Current approaches for constraint consideration
- The constraint-driven design flow
- **Impact on design algorithms and design flow**
- Open problems
- Summary and conclusion

Impact on Design Algorithms and Design Flow

Sequential Design Flow

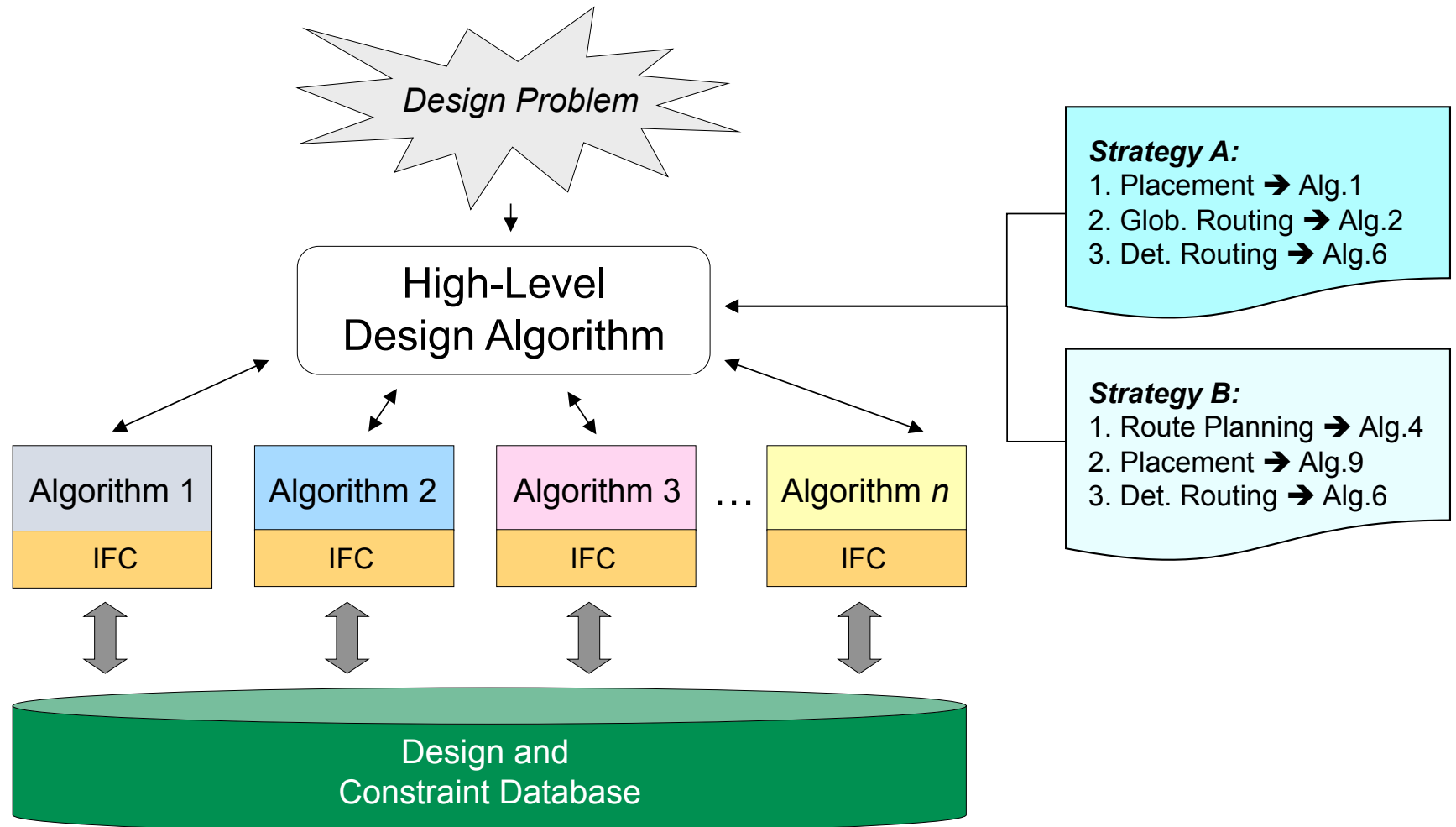


Continuous Design Flow [5]



Impact on Design Algorithms and Design Flow

High-Level Design Algorithms



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

- [1] J. Jaffar et al.: “The CLP(R) language and system: an overview” in *Proc. Compton Spring 1991, Digest of Papers*, pp. 376-381, 1991.
- [2] E. Malavasi, E. Charbon et al.: “Automation of IC Layout with Analog Constraints”, in *IEEE Trans. on CAD of Integrated Circuits*, Vol. 15, No. 8, 1996.
- [3] E. Malavasi, E. Charbon: “Constraint Transformation for IC Physical Design”, in *IEEE Trans. on Semiconductor Manufacturing*, Vol. 12, No. 4, 1999.
- [4] J. Freuer, G. Jerke, J. Gerlach, W. Nebel: “On the Verification of High-Order Constraint Compliance in IC Design”, in *Proc. Design, Automation and Test in Europe, DATE '08*, pp. 26 – 31, 2008.
- [5] Scheible, Jürgen: “Constraint-driven Design – Eine Wegskizze zum Designflow der nächsten Generation”, in *Proc. Beiträge der 10. GMM/ITG-Fachtagung, Analog'08*, Siegen, Germany, 2008.
- [6] G. Jerke, J. Lienig: “Constraint-Driven Design – The Next Step Towards Analog Design Automation”, in *Proc. International Symposium on Physical Design, ISPD'09*, 2009.
- [7] Sakurai T., Tamaru, K.: “Simple Formulas for 2D and 3D capacitances”, in *IEEE Trans. Electron. Dev.*, Vol. ED30, No.2, pp. 183-185, 1983.