On Optimal Irregular Switch Box Designs

Hongbing Fan, Wilfrid Laurier University, Canada

Yu-Liang Wu, The Chinese University of Hong Kong

Chak-Chung Cheung, Imperial College London

Jiping Liu, University of Lethbridge, Canada

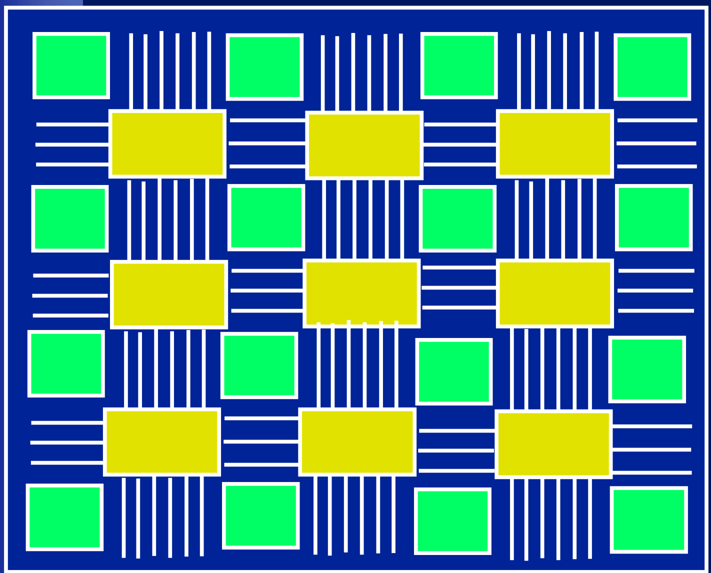
14th International Conference on Field Programmable Logic and Applications, Belgium

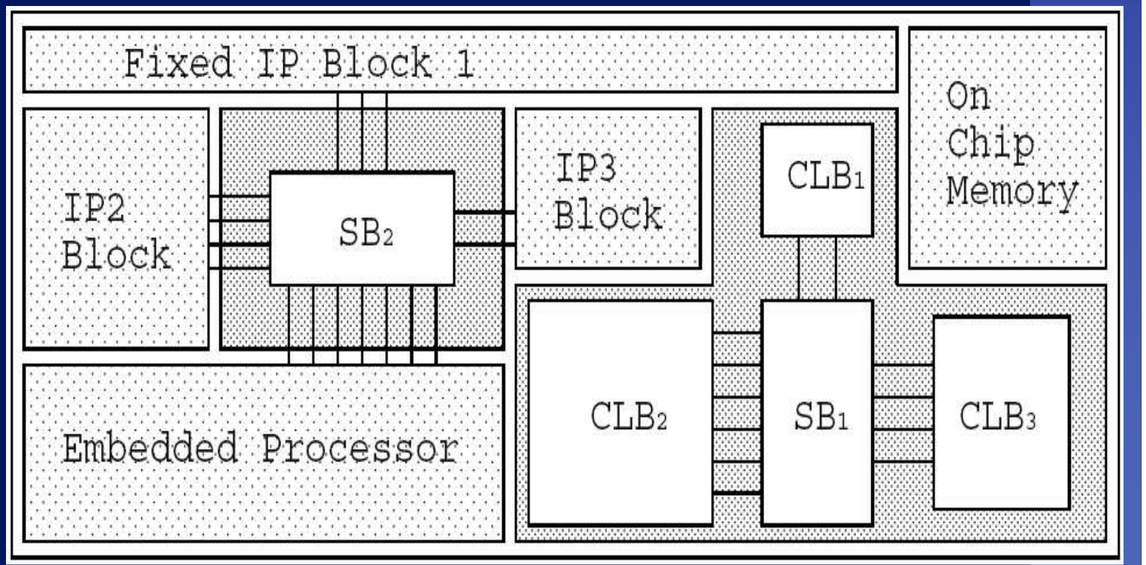
Outline

- Optimization metric & switch box model
- Previous work
- Our methodology
- Examples & results
- Conclusions
- Future work

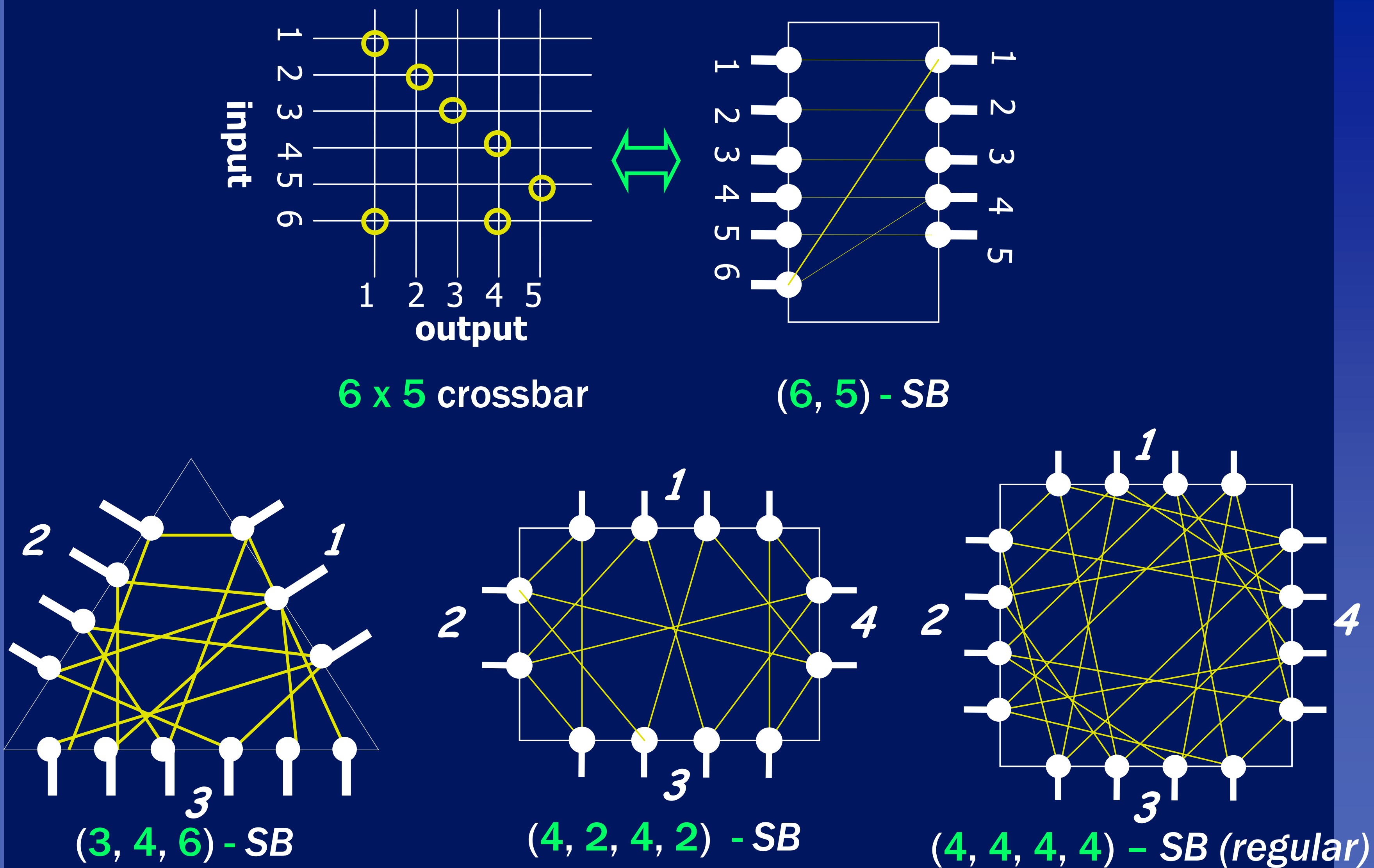
Optimization metric

- Optimzie the area of Switch box (not latency)
  - Fewer channel width, fewer internal switch
- Applications
  - Customised FPGA, SoC designs (various IP cores)
  - Reconfigurable interconnection networks
  - Communication, parallel computing, city planning



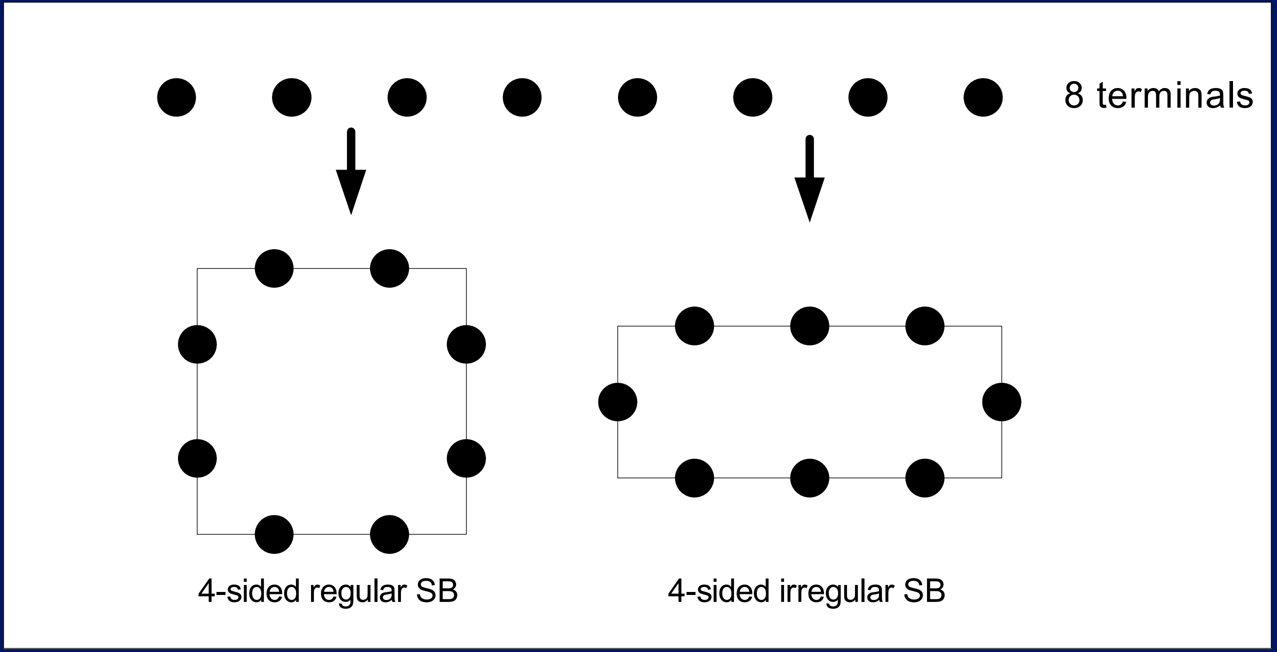


Physical implementation → Model



General switch box model

- A *k* - sided switch box means that terminals are partitioned into *k* sides
- A switch box is ***regular*** if all sides have the same number of switches; otherwise it is ***irregular***



Previous work

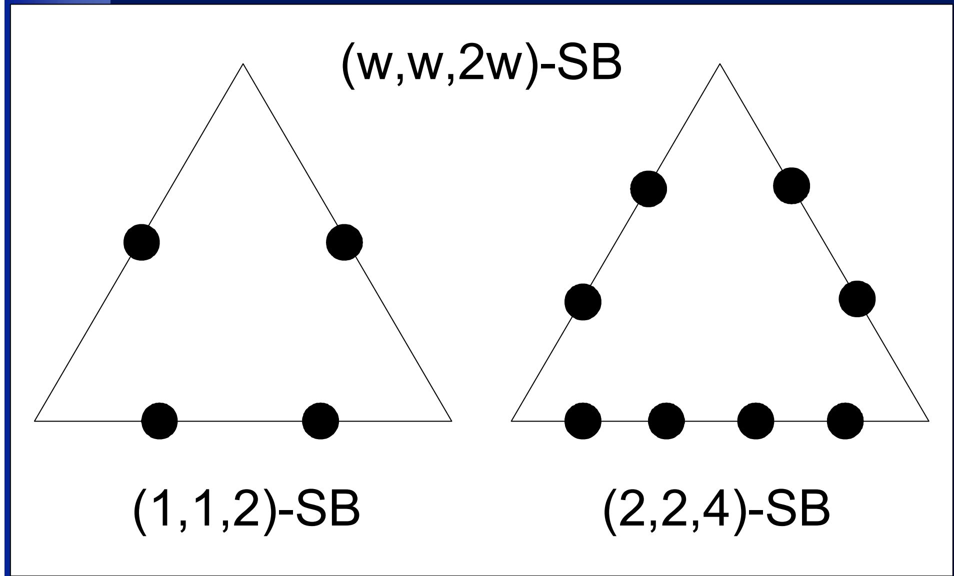
- Regular switch boxes
  - Flexibility, Routability model – *Rose et al, 1991*
  - Universal switch box – *Chang et al, 1996*
  - Hyper-universal switch box - *Fan et al, 2001*
- Irregular switch boxes
  - 3-sided switch boxes – *Dehon et al, 1999*
  - Rectangular switch boxes - *Wilton et al, 2001*
- This paper: general irregular switch box designs
  - For any given channel density, routing capacity

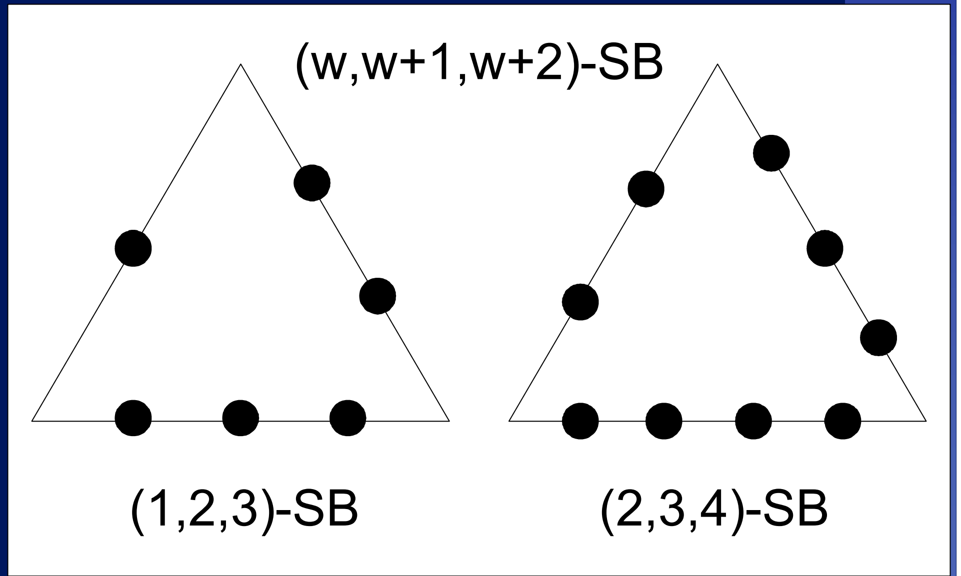
Outline

- Optimization metric & switch box model
- Previous work
- Our methodology
- Examples & results
- Conclusions
- Future work

Defining channel density vector

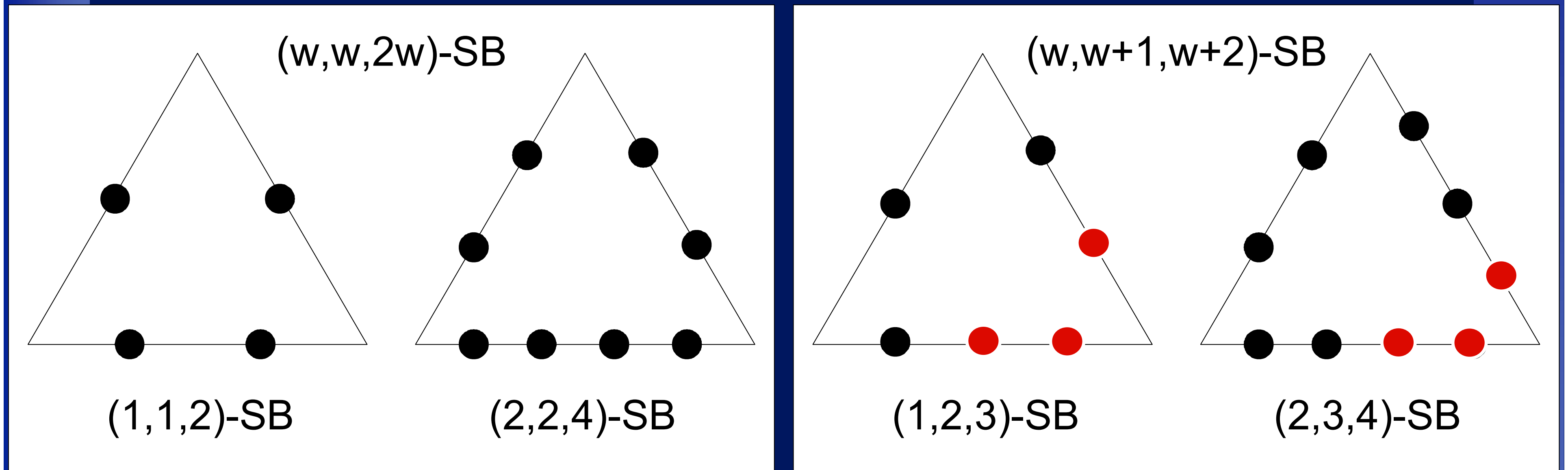
- $(r_1 , \ldots , r_k)$ – SB
  - k-sided switch box
  - r_i is the number of terminals on side i
  - $(r_1 , \ldots , r_k)$ is called the **channel density vector**





Decomposing channel density vector

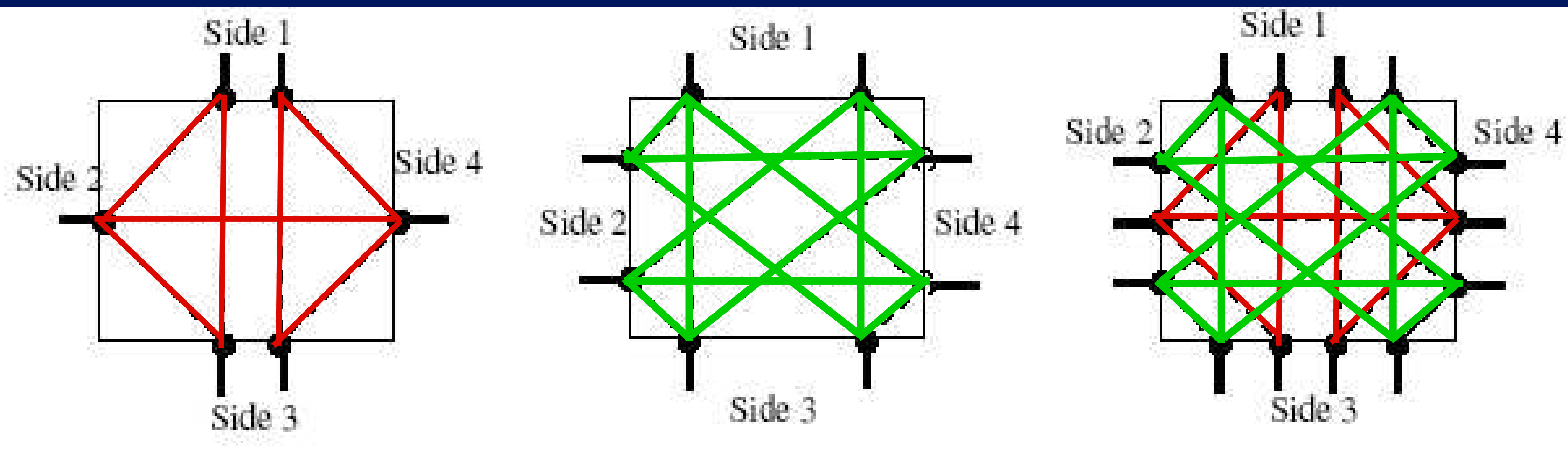
- $(w \times d + c)$ – SB
  - d - density vector, c - residual vector
  - w – integer scaler



d = (1,1,2)	d = (1,1,2)	d = (1,1,1)	d = (1,1,1)
c = (0,0,0)	c = (0,0,0)	c = (0,1,2)	c = (0,1,2)
w = 1	w = 2	w = 1	w = 2

Concept of combining switch boxes

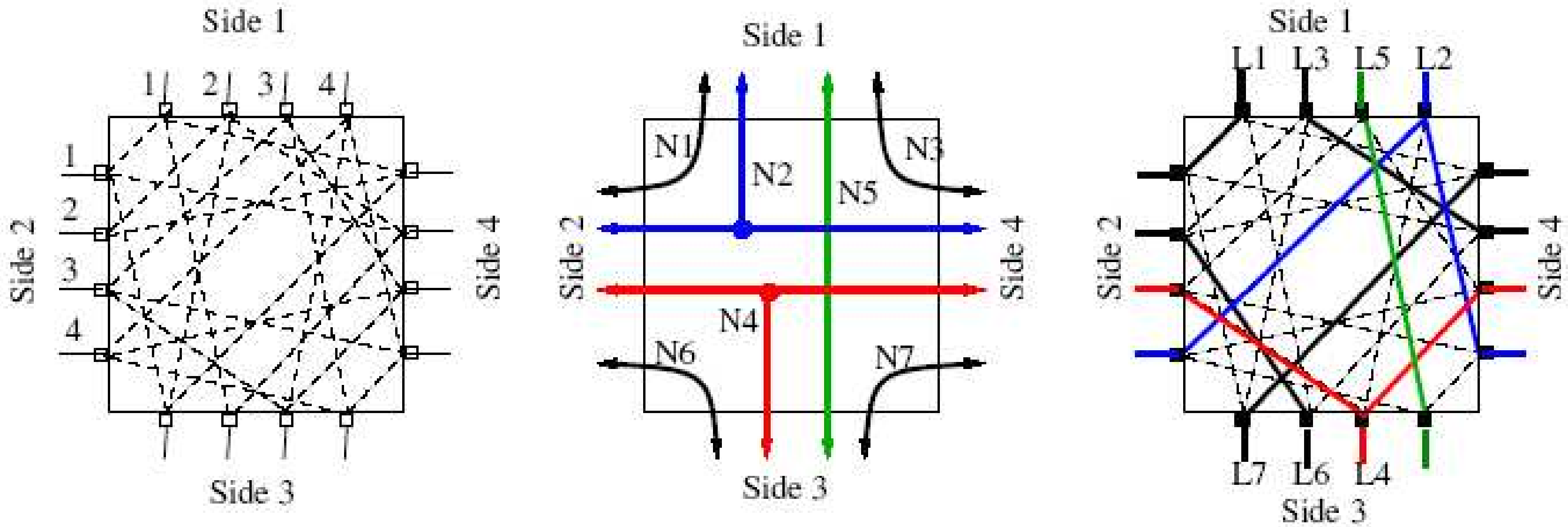
- disjoint union of two switch boxes
- adding two channel density vectors
  - (2, 1, 2, 1) + (2, 2, 2, 2) → (4, 3, 4, 3)
- each switch box
  - accommodates separate routing requirements



(a) (2, 1, 2, 1) - SB (b) (2, 2, 2, 2) - SB (c) (4, 3, 4, 3) - SB

Mapping routing requirements to switch box

- A feasible routing for routing requirements $R = [N_1, \ldots, N_m]$ in an $(r_1, \ldots, r_k)$ – SB
  - below shows when $k = 4$, $m = 7$
  - N_2, N_4 are 3-pin nets, others 2-pin nets



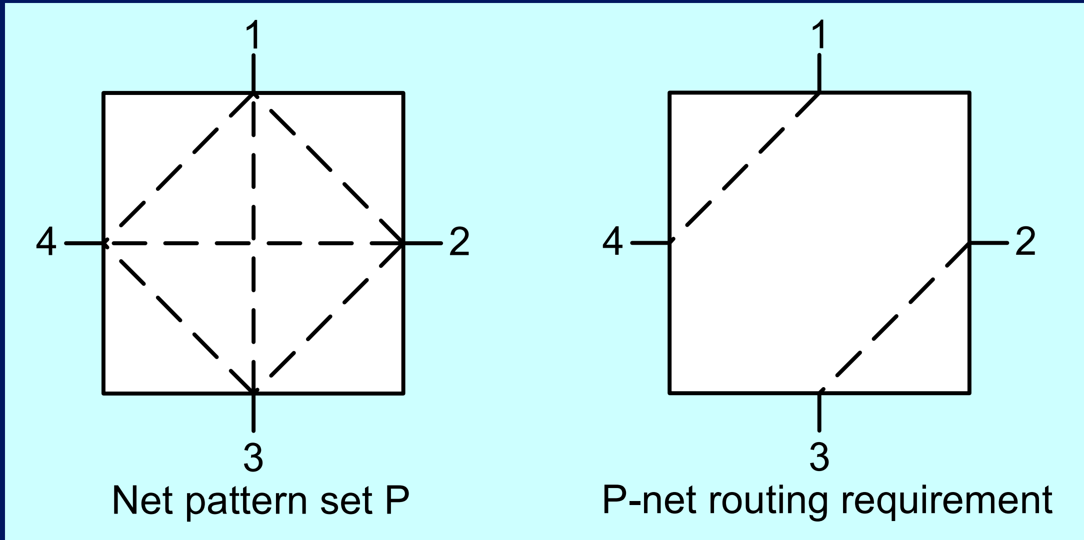
(a) a (4, 4, 4, 4) – SB

(b) a (4, 4, 4, 4) – RR

(c) a feasible routing

Modeling routing requirement vector

- A **net** specifies a subset of connected terminals {*1, … , k*}
- A **net pattern set *P*** consists of the all possible types of nets
- A ***P*-net routing requirement** is a collection of nets in set P



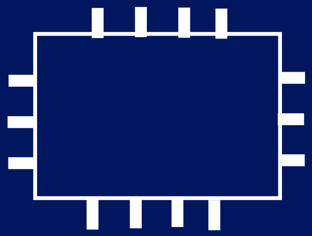
A ***P*-net routing requirement** can be expressed as a vector X satisfying a system of linear Diophantine equations

$$A\,X = (r_1 , \ldots , r_k)^T$$

where A is the connectivity matrix of P

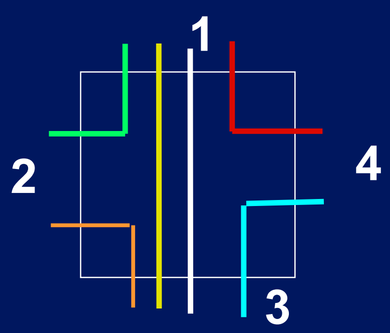
Example of routing requirement vector

Consider a (4, 3, 4, 3)-SB with net pattern set (2-pin nets)
P = [{1,2}, {1,3},{1,4}, {2,3}, {2,4}, {3,4}, {1}, {2}, {3}, {4}]



The connectivity matrix of P is

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$



A routing requirement (RR)
R = {{1, 2}, {3, 4}, {2, 3}, {1, 3}, {1,3}, {1, 4}}

RR vector: X = (1, 2, 1, 1, 0, 1, 0, 0, 0, 0)

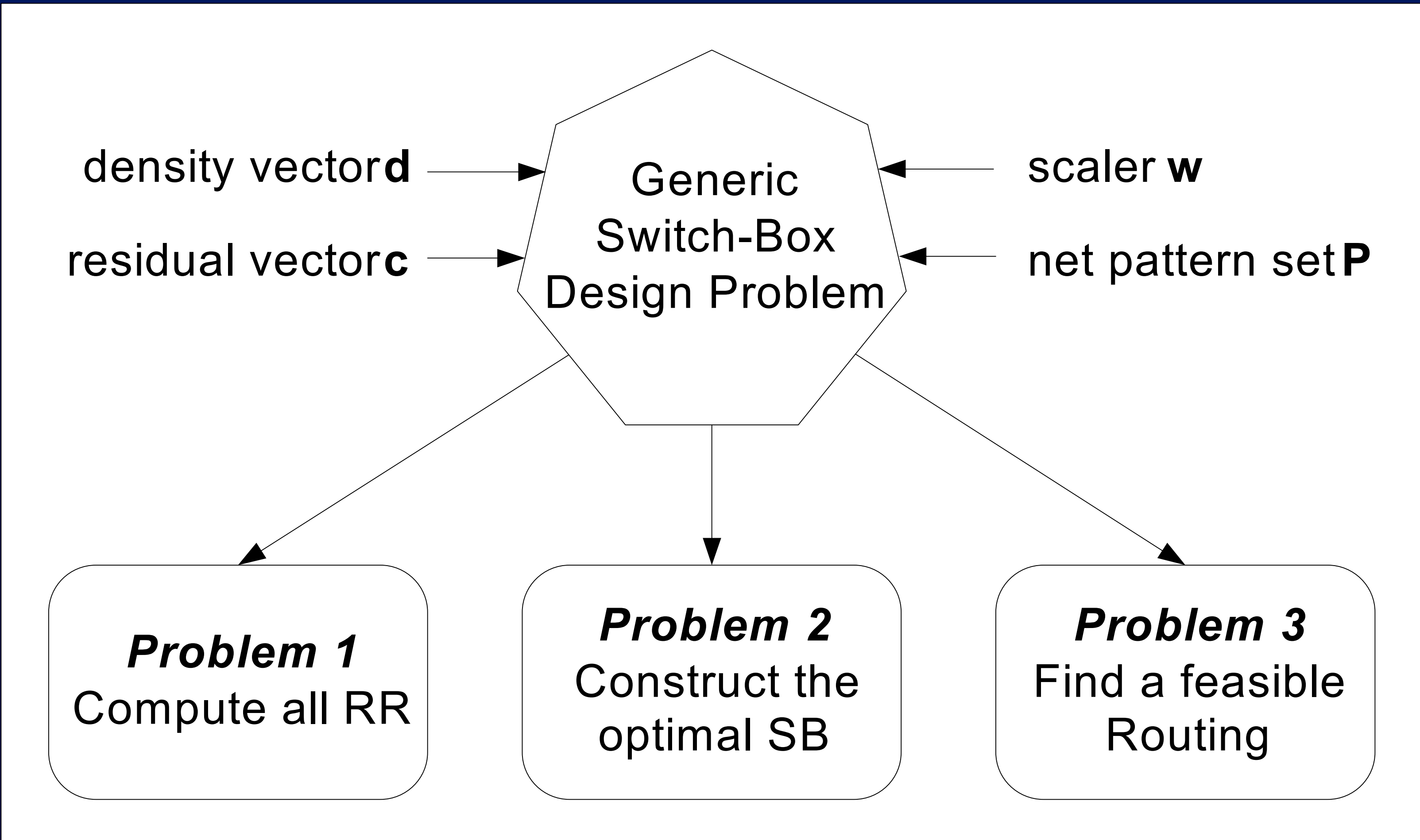
$A\,X^T = (4, 3, 4, 3)^T$

Switch box design problems

- A switch box is *P*-universal if it is routable for every *P*-net routing requirements
  - Universal if *P* consists of all 2-pin nets
  - Hyper-universal if *P* contains all possible types
- Switch box design problem
  - given channel density vector and net pattern set *P*
  - design a *P*-universal $(r_1, \ldots, r_k)$-SB with the minimum number of switches
- Generic switch box design problem
  - given vectors *d* and *c*, and net pattern set *P*
  - design optimal $(w \times d + c)$-SB for every integer $w > 0$
- Solve the SB design problem by selecting proper *density vector*, *residual vector*, and *scaler*

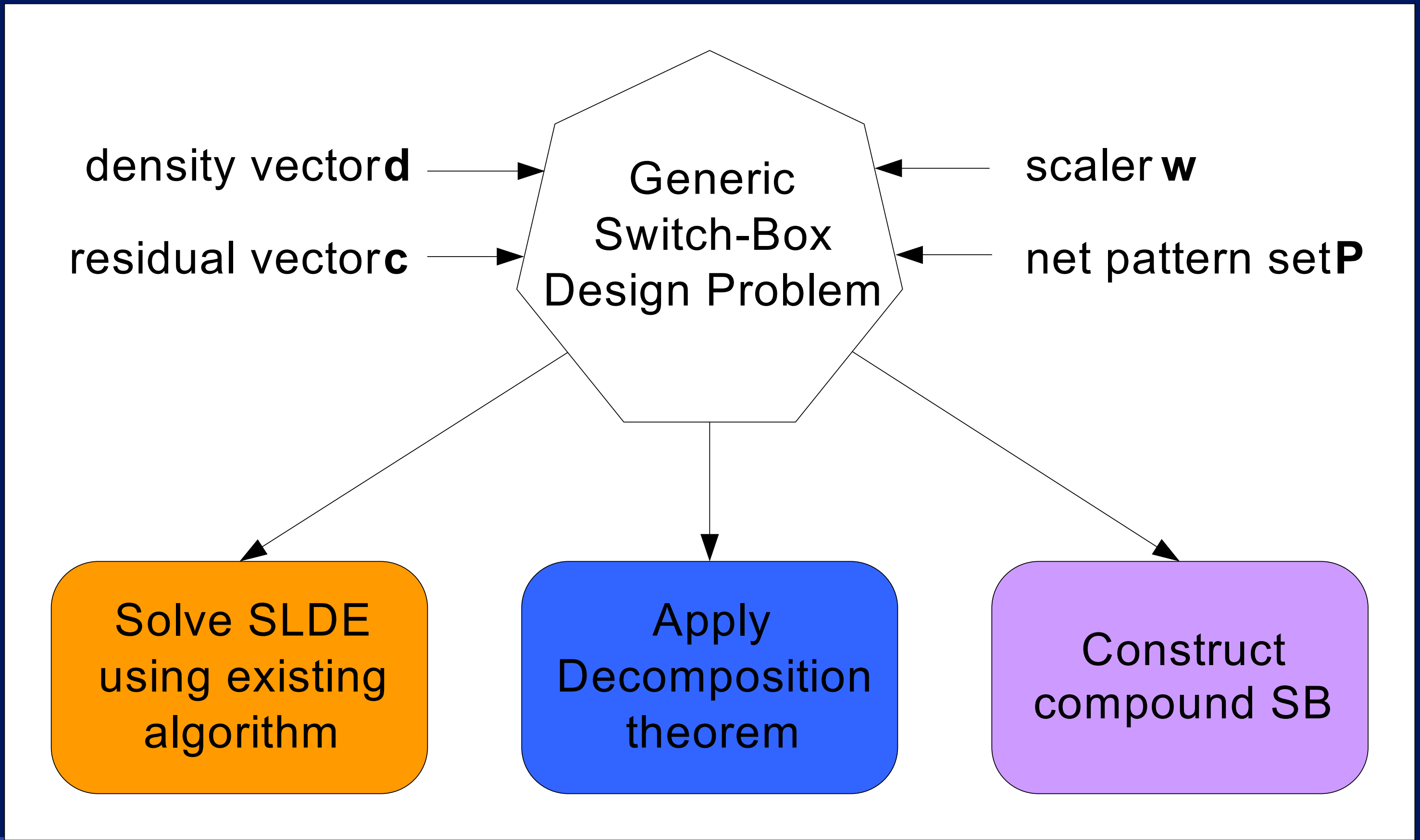
Challenges

- There are three major problems in SB designs



Solution: 3-step algorithm

- 1) Decompose routing requirement vectors
- 2) route in different switch boxes
- 3) combine disjoint switch boxes



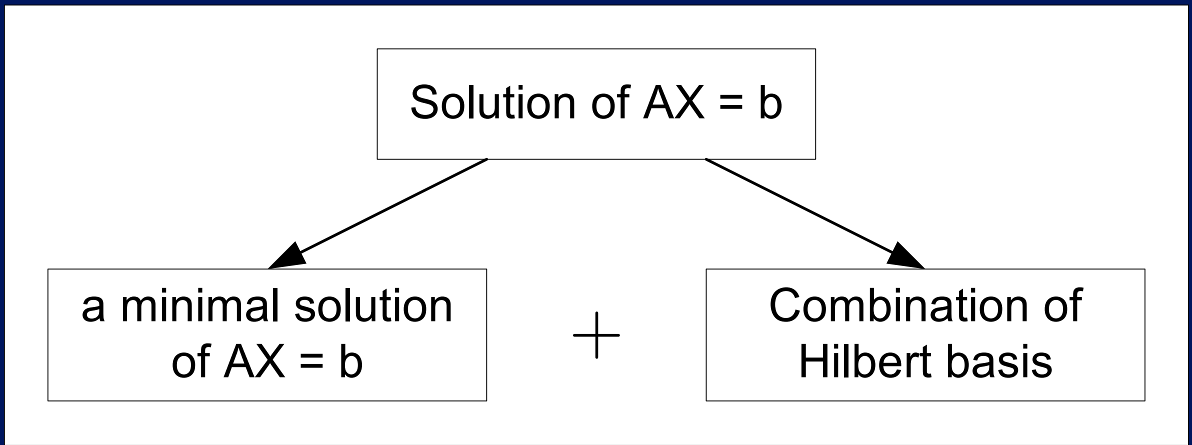
System of Linear Diophantine Equation

- A Diophantine equation is an equation in which only nonnegative integer solutions are considered
  - Example: $1027x_1 - 712x_2 = 1$
- A system of linear Diophantine equation ***AX*** = ***b*** has a finite number of minimal solutions

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & -1 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ w \end{pmatrix}}_{X} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}}_{b}$$

Main contribution – Divide and Conquer theorem

- The set of minimal solutions of $AX = 0$ (Hilbert basis) is computed by using existing algorithms
- Theorem: any solution of $AX = b$ can be expressed as the sum of a minimal solution of $AX = b$ and linear combination of Hilbert basis of $AX = 0$
- The set of routing requirements can be generated efficiently by applying Hilbert basis algorithm



Implication:

Decompose RRV

Proof of theorem

Available upon request

Compute the Hilbert basis

B_0 $AX^T - \mathrm{d}^T w = 0$

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & -1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ w \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Input: $AX = 0$, a system of linear Diophantine equations.

$\mathcal{E} := \{e_1, \ldots, e_n\};$

$\mathcal{B} := \emptyset;$

while $\mathcal{E} \neq \emptyset$ **do**;

$\mathcal{B} := \mathcal{B} \cup \{X \in \mathcal{E} \mid AX = 0\};$

$\mathcal{C} := \{X \in \mathcal{E} \setminus \mathcal{B} \mid \forall S \in \mathcal{B}, X \not\geq S\};$

$\mathcal{E} := \{X + e_i \mid X \in \mathcal{C}, (AX)^T(Ae_i) < 0\};$

end while

Output $\mathcal{B}$

$(1,1,1,0,0,0,0,1), (0,0,0,0,0,0,1,1),$
$(1,0,0,0,0,1,0,1), (0,1,0,0,1,0,0,1),$
$(0,0,0,1,1,1,0,2), (0,0,1,1,0,0,0,1).$

Compute minimal solutions

B $AX^T - \mathrm{d}^T w = \mathrm{c}^T$

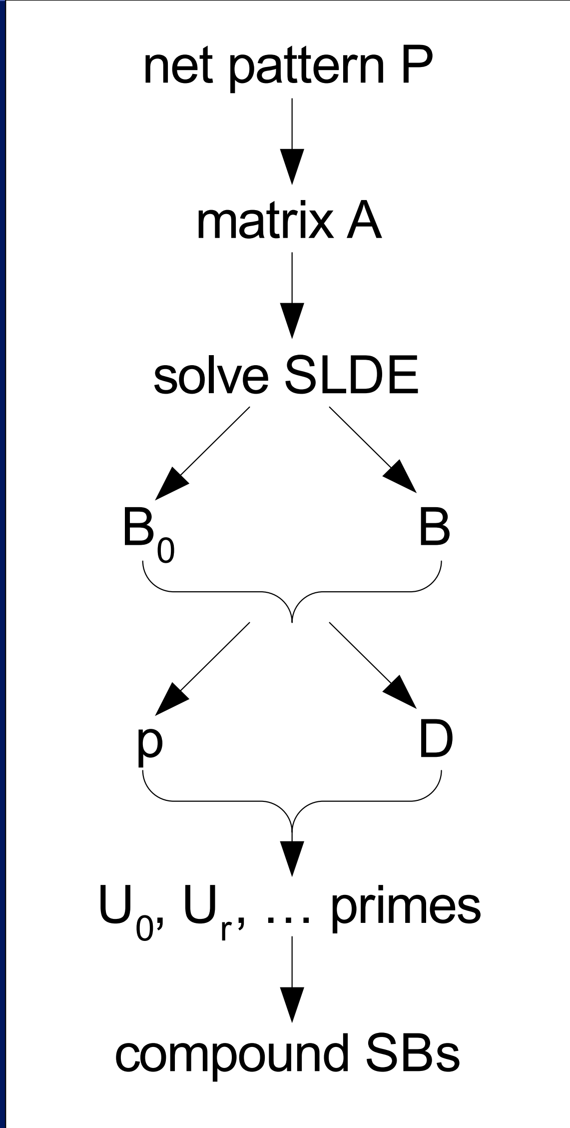
$(0,1,2,0,0,0,0,0), (0,0,1,0,0,1,0,0), (0,0,0,0,1,2,0,1)$

General decomposition theorem

- Given: density vector *d*, residual vector *c* and net pattern set *P*
- Output: an integer *p* and a set of integers *D*
  - any width $w > 1$, there is an integer q_w in *D*
  - every RR for (w d + c)-SB $\Rightarrow$
    - one $(q_w\ d + c)$ – RR
    - $(w - q_w)/p$ copies of $(p\ d)$-RRs
    - E.g., $w = 10$, $q_w = 2$, $p = 2 \Rightarrow 4$ copies
- *P*-universal $(w\ d + c)$-SB
  - disjoint union of *P*-universal $(q_w\ d + c)$-SB
  - $(w - q_w)/p$ copies of a *P*-universal $(p\ d)$-SB

Design scheme for generic switch boxes

1. Input: connectivity matrix *A* of net pattern set *P*
   - find the Hilbert basis B_0 of
     - $AX - d^T w = 0$
   - B of all minimal solutions of
     - $AX - d^T w = c^T$
2. Determine *p* and *D* by the B_0 and B
3. Design a *P*-universal *(p d)*-SB U_0 and a *P*-universal *(r d + c)*-SB U_r for every *r* in *D*, called them prime SBs
4. For every $w > 1$, let *n* be the minimum n such that $w-np$ is in *D* Then $U_{w-np} + nU_0$ is a *P*-universal *(w d + c)*-SB

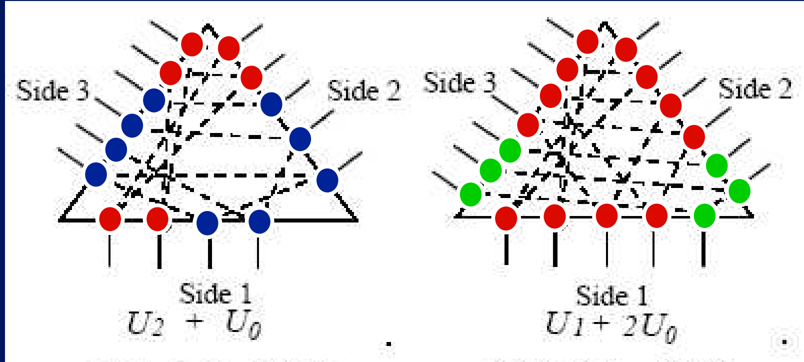


Outline

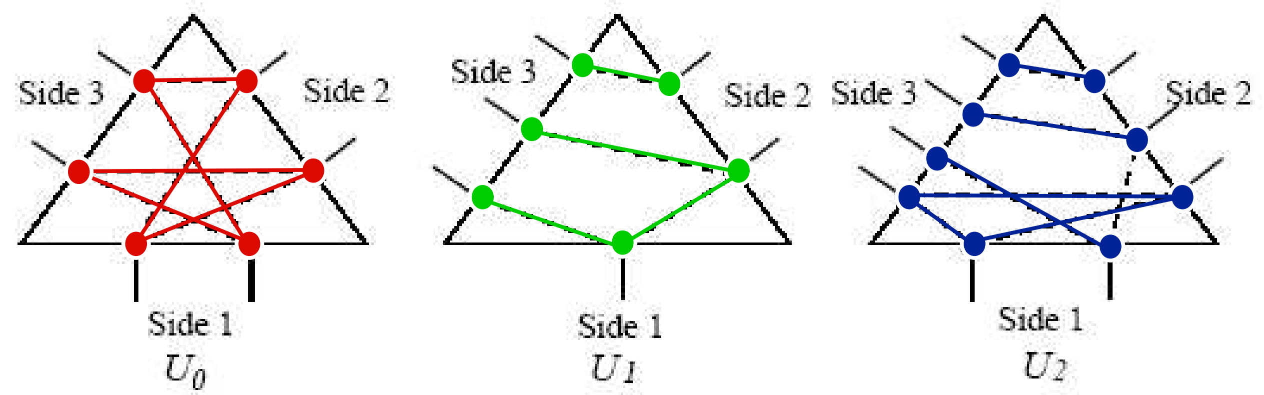
- Optimization metric & switch box model
- Previous work
- Our methodology
- Examples & results
- Conclusions
- Future work

Design example 1: (4, 5, 6) & (5, 6, 7)-HUSB

- Given: Terminal (v) (4,5,6), (5,6,7)
- Compute (w, w+1, w+2)-HUSB problem
  - d = (1, 1, 1), c = (0, 1, 2)
  - Net pattern P = {{1}, {2}, {3}, {1,2}, {1,3}, {2,3}, {1,2,3}}
  - Solved *p* = 2, *D* = {1, 2}
- Design *(p d)*-SB U_0
  - (2, 2, 2)-SB
- *(r d + c)*-SB U_r for every *r* in *D*
  - (1, 2, 3)-SB, (2, 3, 4)-SB



(d) (4, 5, 6) – HUSB
(e) (5, 6, 7) – HUSB



(a) (2, 2, 2) – HUSB
(b) (1, 2, 3) – HUSB
(c) (2, 3, 4) – HUSB

Design example 2: (w, 2w, w, 2w)-USB

Design optimal (w, 2w, w, 2w)-USBs for every $w > 1$

1. Given: density vector $d = (1,2,1,2)$,

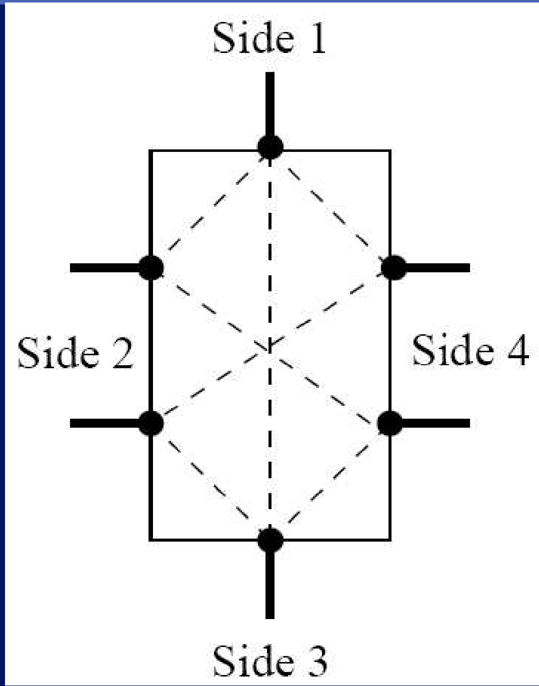
   residual vector $c = (0,0,0,0)$ and net pattern set P =

   [{1,2}, {1,3},{1,4}, {2,3}, {2,4}, {3,4}, {1}, {2}, {3}, {4}]

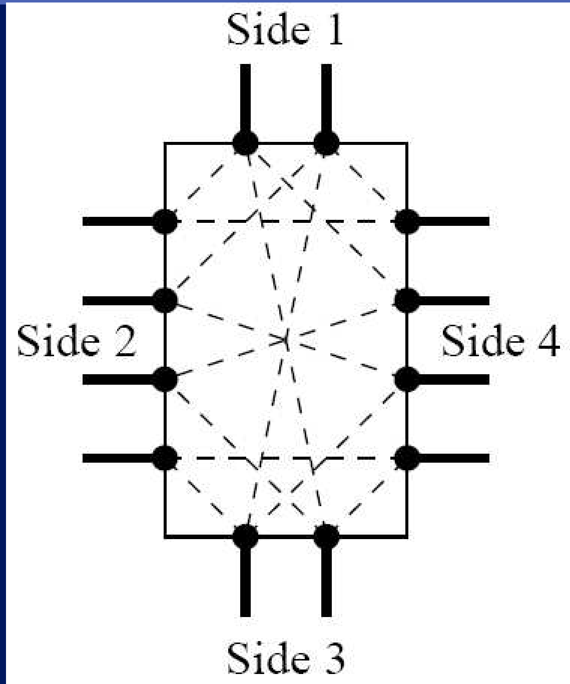
$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

2. Compute: p and D, we obtain $p = 2$ and $D = \{1, 2\}$
3. Design optimal (2,4,2,4)-USB and (1,2,1,2)-USB

Design example 2 – Prime SBs



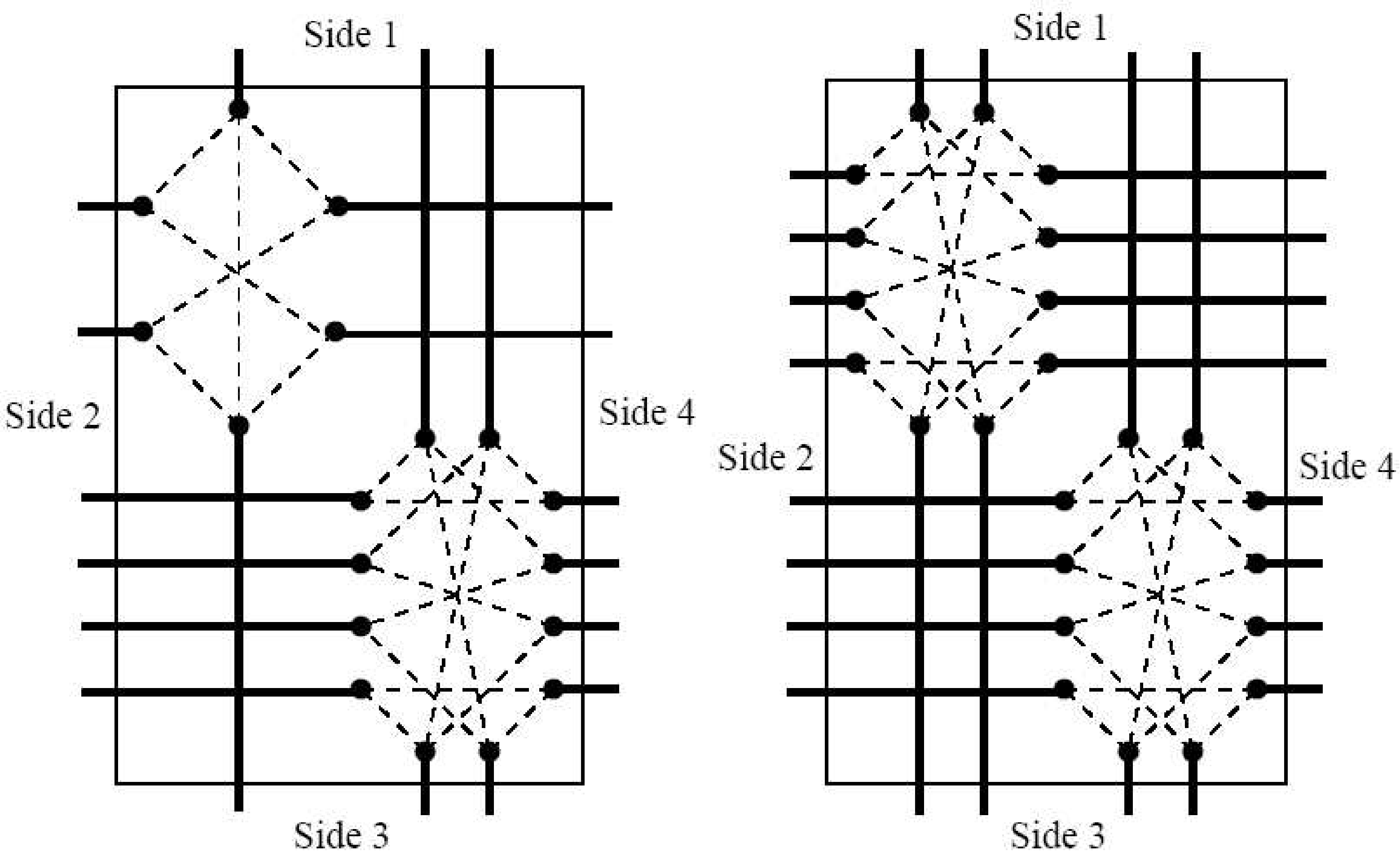
Optimal (1,2,1,2)-USB U1



Optimal (2,4,2,4)-USB U0

4. **Output: optimal (w, 2w, w, 2w)-USBs**
   - **Even w, disjoint union of w/2 copies of U0**
   - **Odd w, disjoint union of one U1 and (w-1)/2 U0**

Design example 2 – Compound SBs

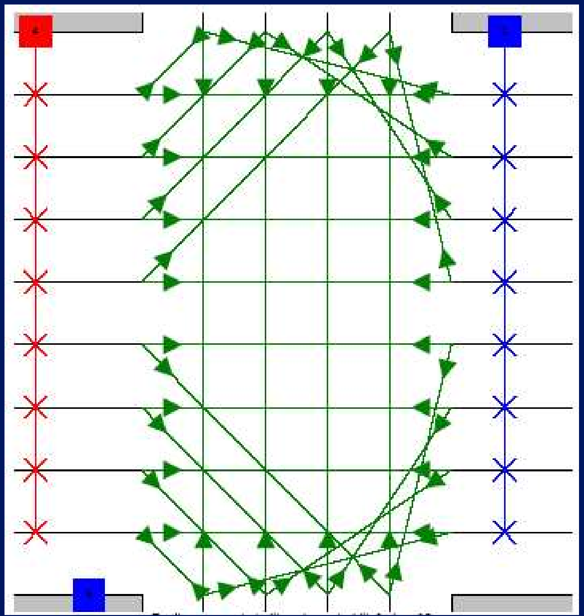


w = 3 ⇒ U1 + U0

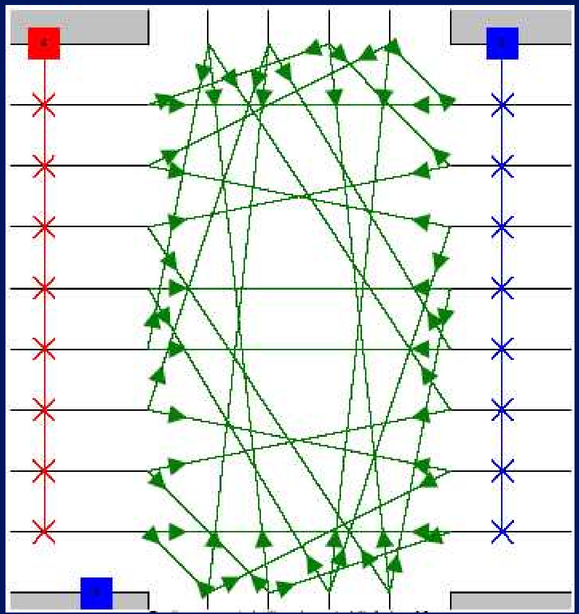
w = 4 ⇒ 2 U0

FPGA Experiments

- **Given: design rectangular (w, 2w, w, 2w)-USBs**
- **Compute: optimal switch boxes with any w**
- **Output: use “VPR” on 21 MCNC benchmark circuits**
- **Compare the channel width to route the circuits**

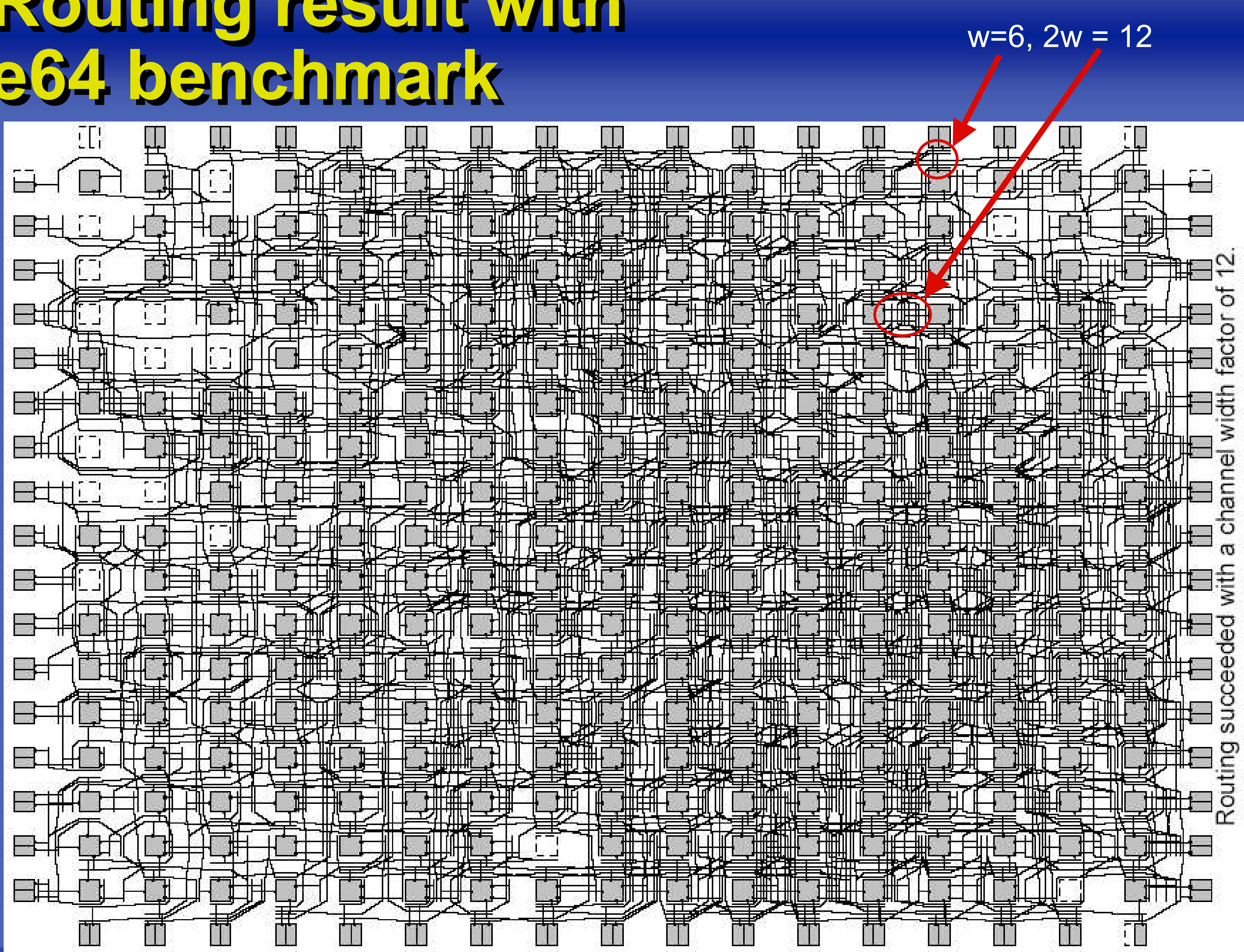


Disjoint-like rectangular switch box



Optimal rectangular switch box

Routing result with e64 benchmark



Experimental results

Channel width reduction → smaller switch box & smaller FPGA area

	Disjoint-like	Optimal Design		Disjoint-like	Optimal Design
alu4	7	7	ex5p	11	10
apex2	8	8	frisc	10	9
apex4	10	9	misex3	9	8
bigkey	5	5	s298	6	6
clma	9	9	s38417	6	5
des	6	5	s38584.1	6	6
diffeq	6	6	seq	9	8
dsip	5	5	spla	10	10
elliptic	10	9	tseng	5	5
ex1010	8	7	e64	6	6
Total				152	143 (-6.3%)

Channel densities required for different benchmark circuits

Fc = W, Fs = 3

Conclusions

- A general divide and conquer design theory and technique for a wide range of switch boxes
- Optimal design: construct by disjoint union of smaller prime switch boxes
- SB designs have linear number of switches and a linear time feasible routing algorithm
- Experiments show the optimal rectangular switch boxes improve of global FPGA routability
- Future work
  - Unidirectional switch boxes
  - Multi-sided switch boxes justification
  - Network-on-a-chip

The End

Thank you for your questions!