Introduction to Real-Time Systems

ECE 397-1

Northwestern University

Department of Computer Science

Department of Electrical and Computer Engineering

Teachers:	Robert Dick	Peter Dinda
Office:	L477 Tech	338, 1890 Maple Ave.
Email:	dickrp@ece.northwestern.edu	pdinda@cs.northwestern.edu
Phone:	467–2298	467-7859
Webpage:	http://www.ece.northwestern.edu/EXTERNAL/realtime	

Homework index

Goals for lecture

- Explain details of a real-time design problem
- Give some background on development of area
- Synthesis solution
- Current commercial status

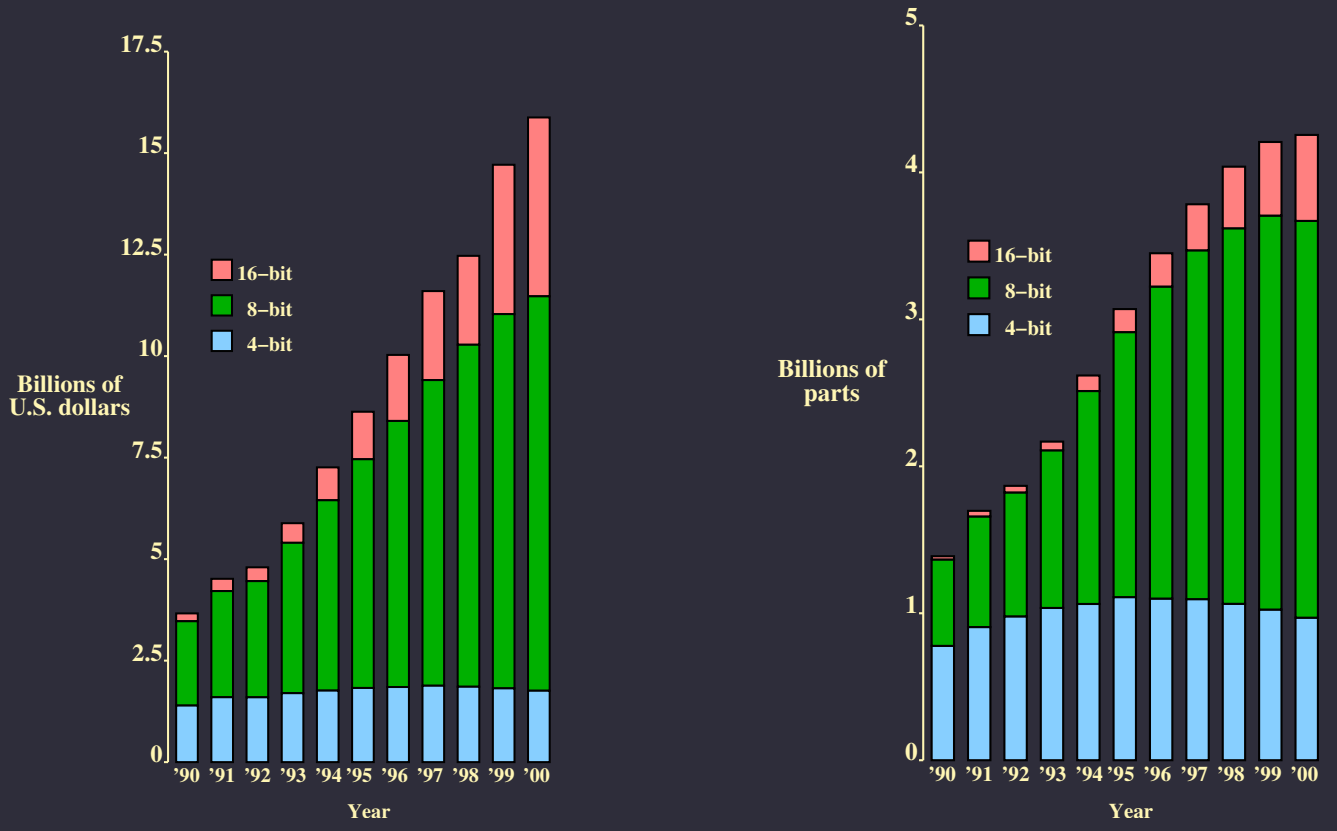
Distributed real-time: Part one

- Distributed needn’t mean among cities or offices – Same IC?
- Process scaling trends
- Cross-layer design now necessary

Embedded system / SOC synthesis motivation

- Wireless: effects of the communication medium important
- Hard real-time: deadlines must not be violated
- Reliable: anti-lock brake controllers shouldn't crash
- Rapidly implemented: IP use, simultaneous HW-SW development
- High-performance: massively parallel, using ASICs
- SOC market from $1.1 billion in 1996 to $14 billion in 2000 (Dataquest), to $43 billion in 2009 (Global Information, Inc.)

Global μ-controller sales



Source: Embedded Processor and Microcontroller Primer and FAQ by Russ Hersch

Low-power motivation

- Embedded systems frequently battery-powered, portable
- High heat dissipation results in
  - Expensive, bulky packaging
  - Limited performance
- High-level trade-offs between
  - Power
  - Speed
  - Price
  - Area

Past embedded system synthesis work

- **Early 1990s**: Optimal MILP co-synthesis of small systems
  [Prakash & Parker], [Bender], [Schwiegershausen & Pirsch]
- **Mid 1990s**: One CPU-One ASIC
  [Ernst, Henkel & Benner], [Gupta & De Micheli]
  [Barros, Rosenstiel, & Xiong], [D'Ambrosio & Hu]
- **Late 1990s – present**: Co-synthesis of heterogeneous distributed embedded systems [Kuchcinski],
  [Quan, Hu, & Greenwood], [Wolf]

Past low-power work

- **Mid 1990s**: VLSI power minimization design surveys
  [Pedram], [Devadas & Malik]
- **Mid – late 1990s**: High-level power analysis and optimization
  [Raghunathan, Jha, & Dey], [Chandrakasan & Brodersen]
- **Late 1990s**: Embedded processor energy estimation
  [Li & Henkel], [Sinha & Chandrakasan]
- **Late 1990s – present**: Low-power hardware-software co-synthesis
  [Dave, Lakshminarayana, & Jha], [Kirrovski & Potkonjak]

Overview of system synthesis projects

- **TGFF**: Generates parametric task graphs and resource databases
- **MOGAC**: Multi-chip distributed systems
- **CORDS**: Dynamically reconfigurable
- **COWLS**: Multi-chip distributed, wireless, client-server
- **MOCSYN**: System-on-a-chip composed of hard cores, area optimized

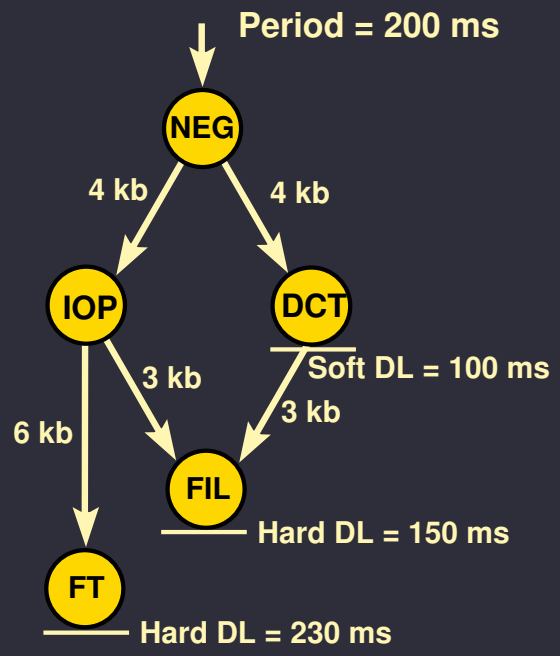
Overview of system synthesis projects

- Synthesize embedded systems
  - heterogeneous processors and communication resources
  - multi-rate
  - hard real-time
- Optimize
  - price
  - power consumption
  - response time

Overview of system synthesis projects

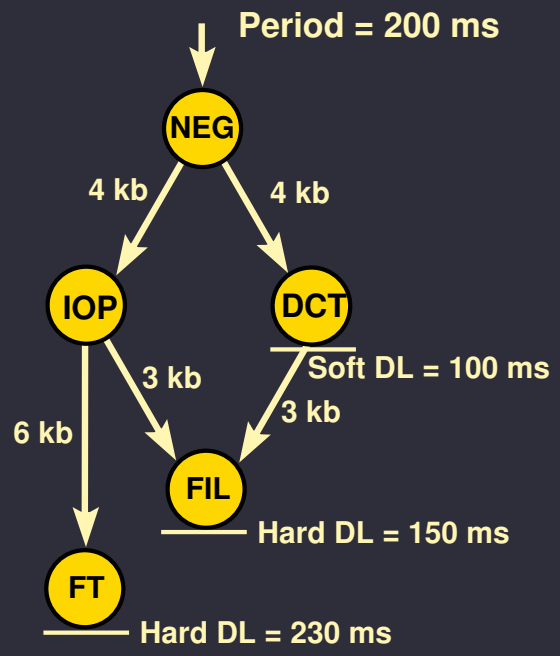
- **TGFF**: Generates parametric task graphs and resource databases
- **MOGAC**: Multi-chip distributed systems
- **CORDS**: Dynamically reconfigurable
- **COWLS**: Multi-chip distributed, wireless, client-server
- **MOCSYN**: System-on-a-chip composed of hard cores, area optimized

Definitions



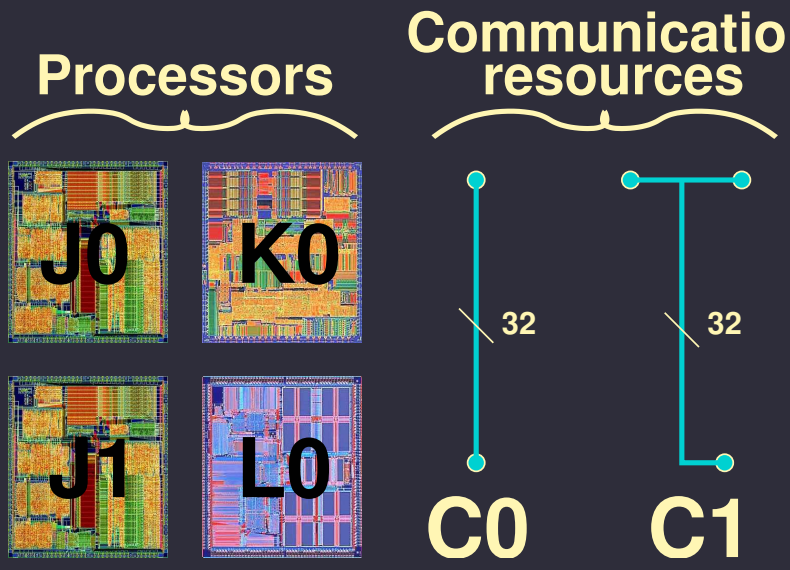
- Specify
  - task types
  - data dependencies
  - hard and soft task deadlines
  - periods
- Analyze performance of each task on each resource
- Allocate resources
- Assign each task to a resource
- Schedule the tasks on each resource

Definitions



- Specify
  - task types
  - data dependencies
  - hard and soft task deadlines
  - periods
- Analyze performance of each task on each resource
- Allocate resources
- Assign each task to a resource
- Schedule the tasks on each resource

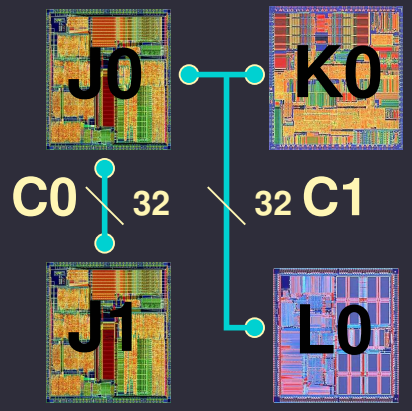
Allocation

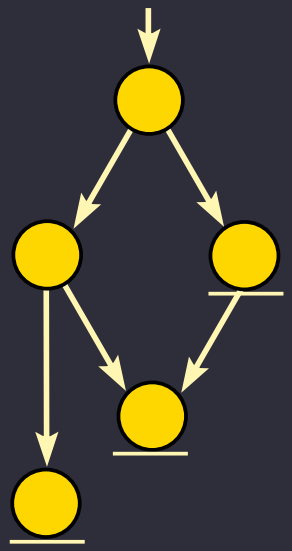


Number and types of:

- PEs or cores
- Commun. resources

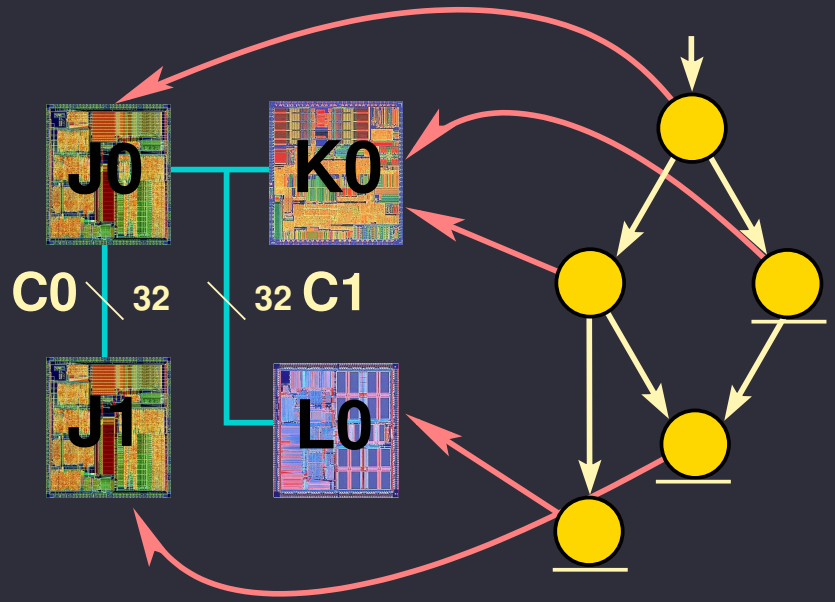
Assignment





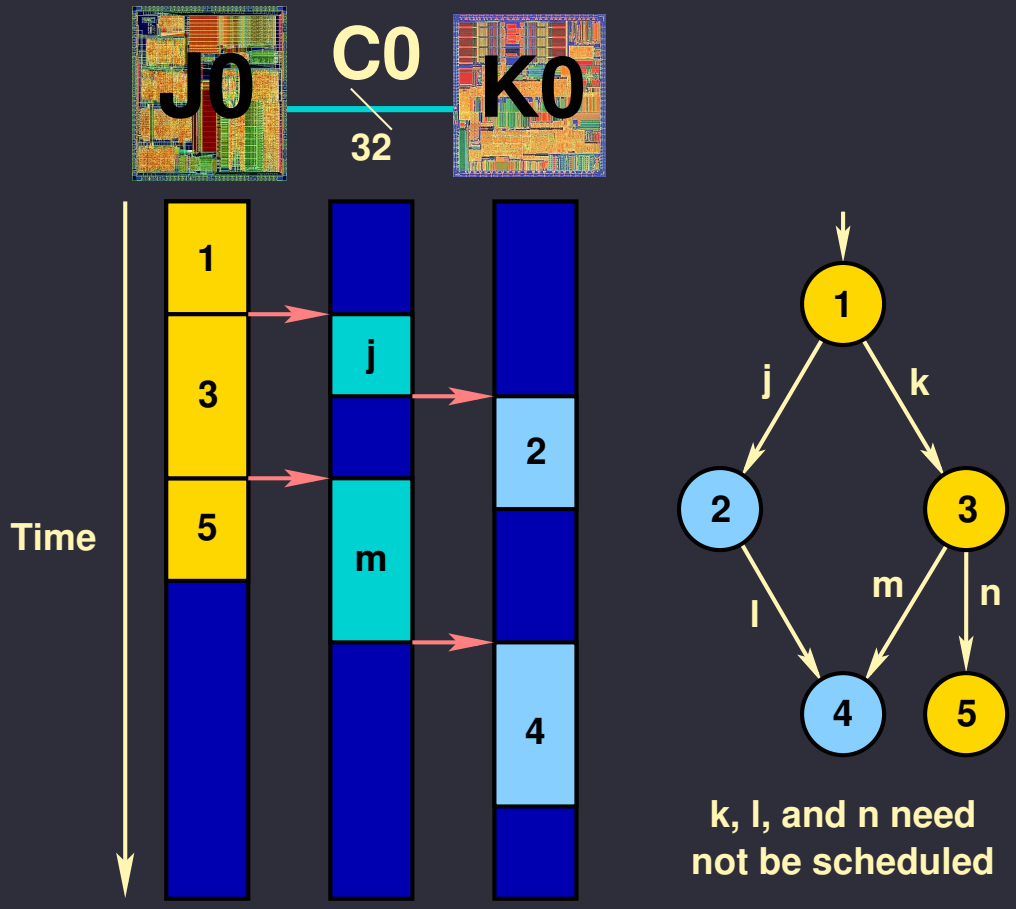
- Assignment of tasks to PEs
- Connection of communication resources to PEs

Assignment



- Assignment of tasks to PEs
- Connection of communication resources to PEs

Schedule


J0
C0
32
K0
Time
1
3
5
j
m
2
4
1
j
k
2
3
m
n
l
4
5
k, l, and n need
not be scheduled

Costs

Soft constraints:

- price
- power
- area
- response time

Hard constraints:

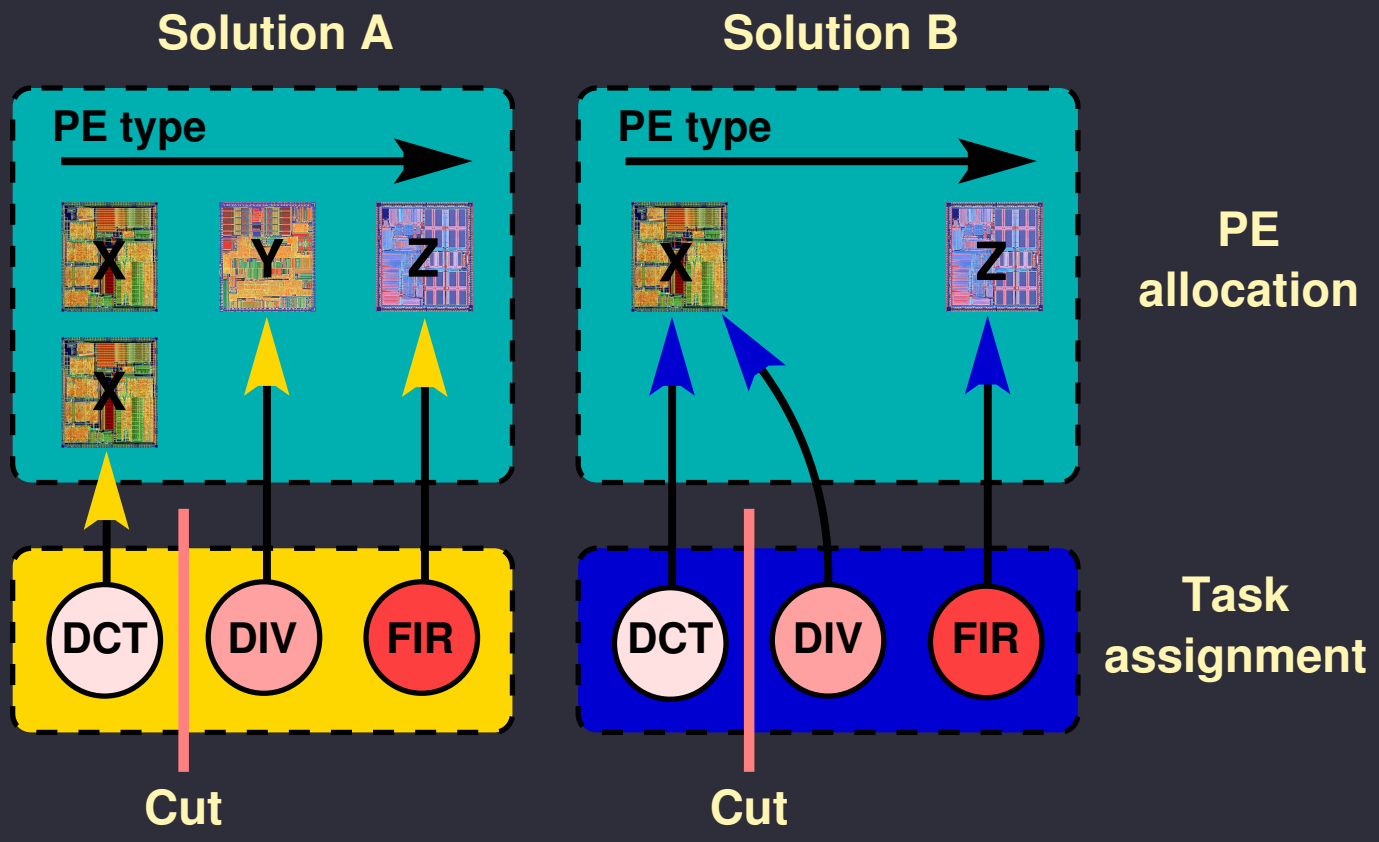
- deadline violations
- PE overload
- unschedulable tasks
- unschedulable transmissions

Solutions which violate hard constraints not shown to designer – pruned out.

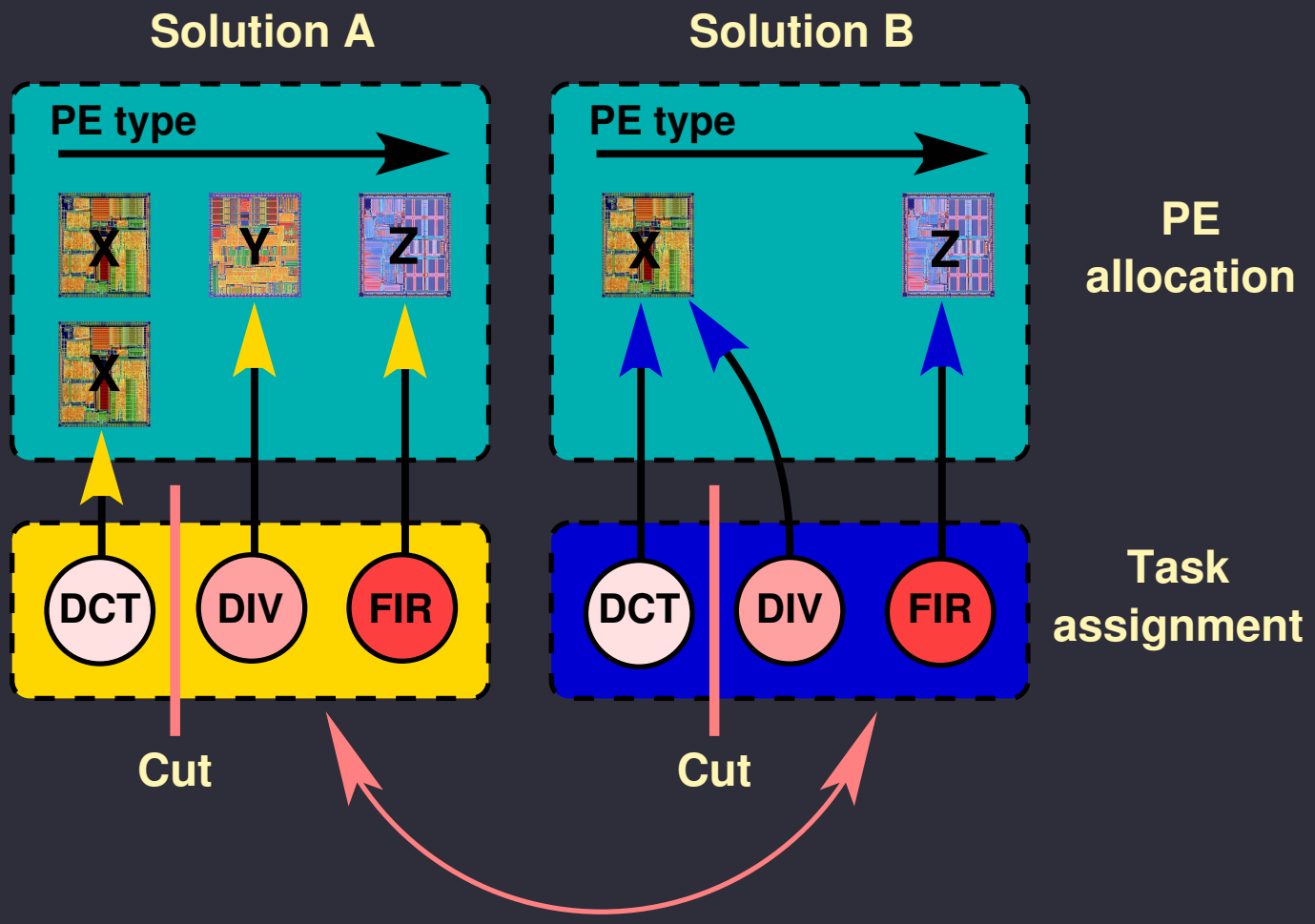
Genetic algorithms

- Multiple solutions
- Local randomized changes to solutions
- Solutions share information with each other
- Can escape sub-optimal local minima
- Scalable

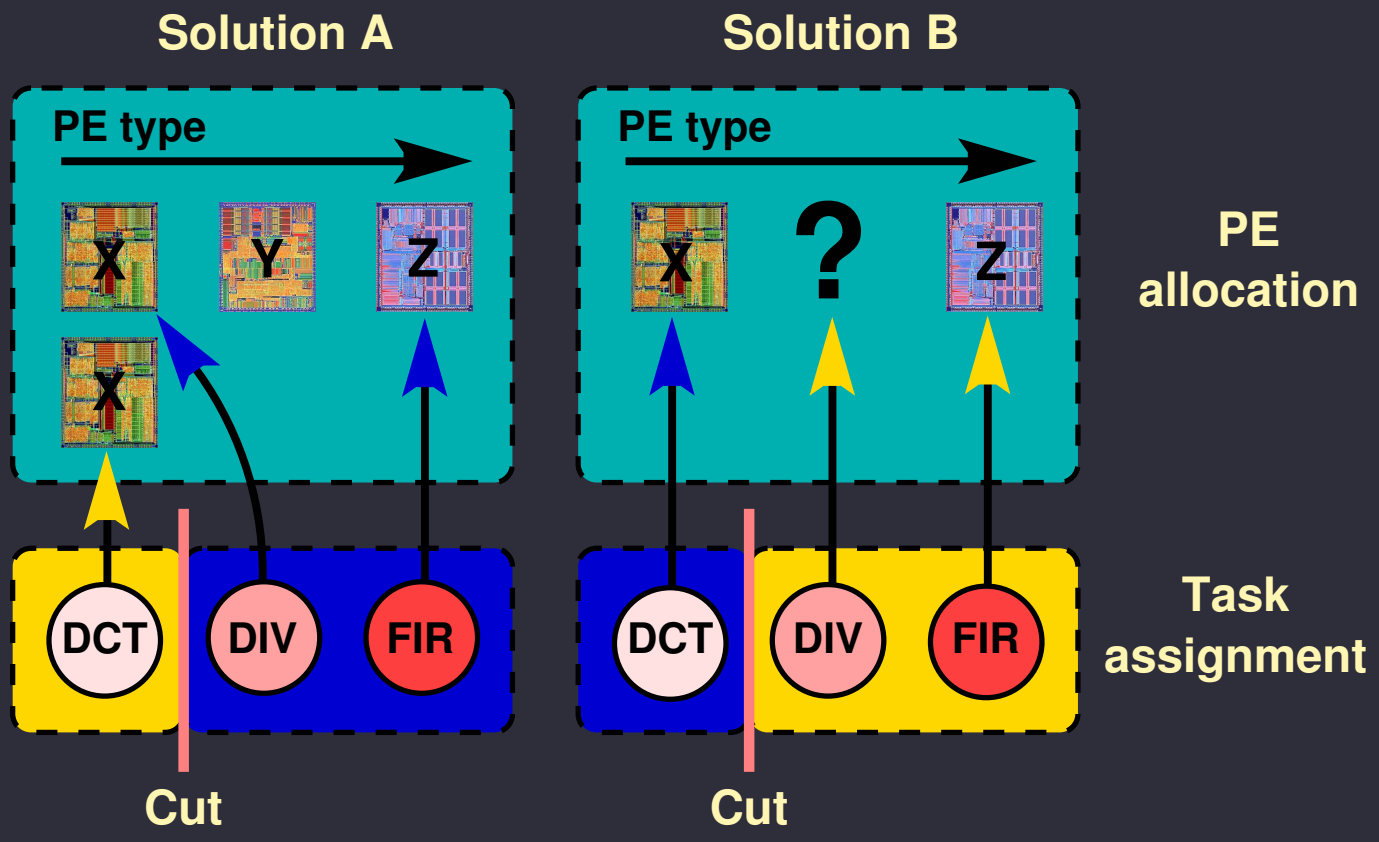
Cluster genetic operator constraints motivation



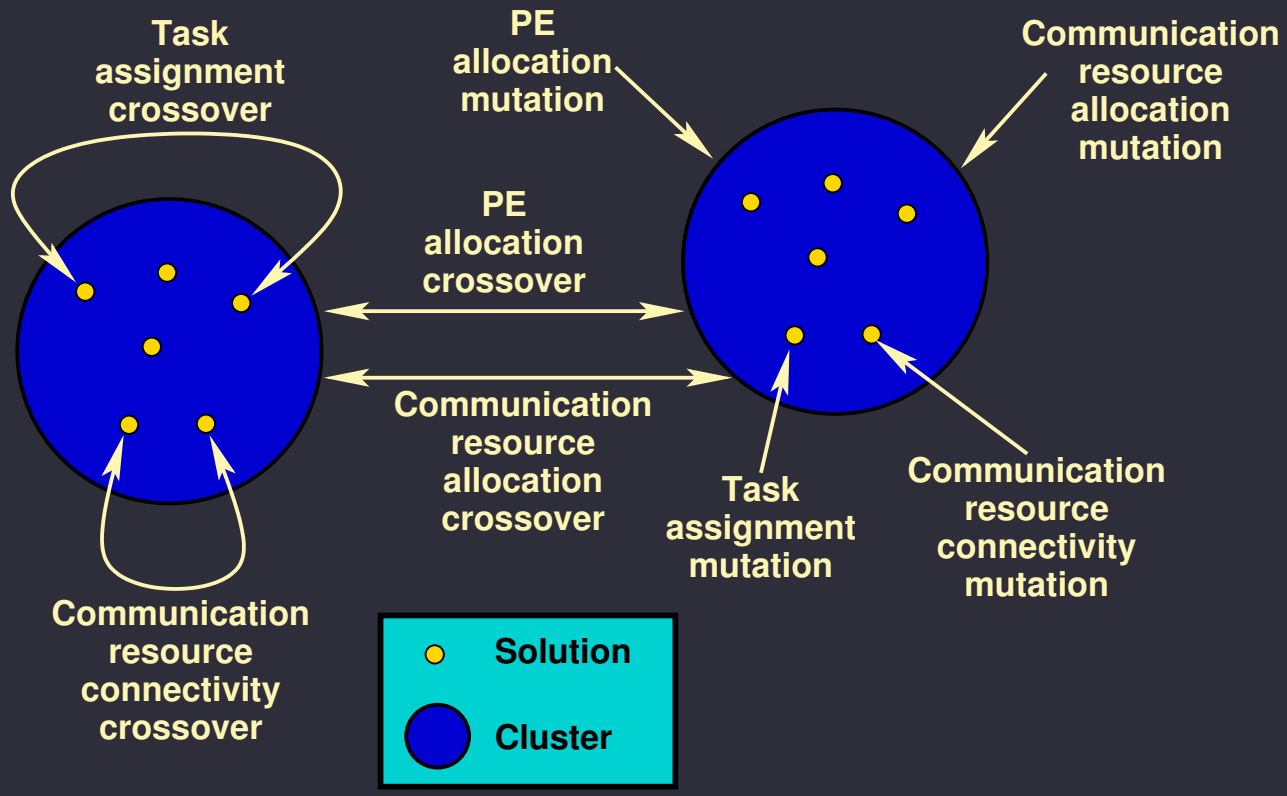
Cluster genetic operator constraints motivation



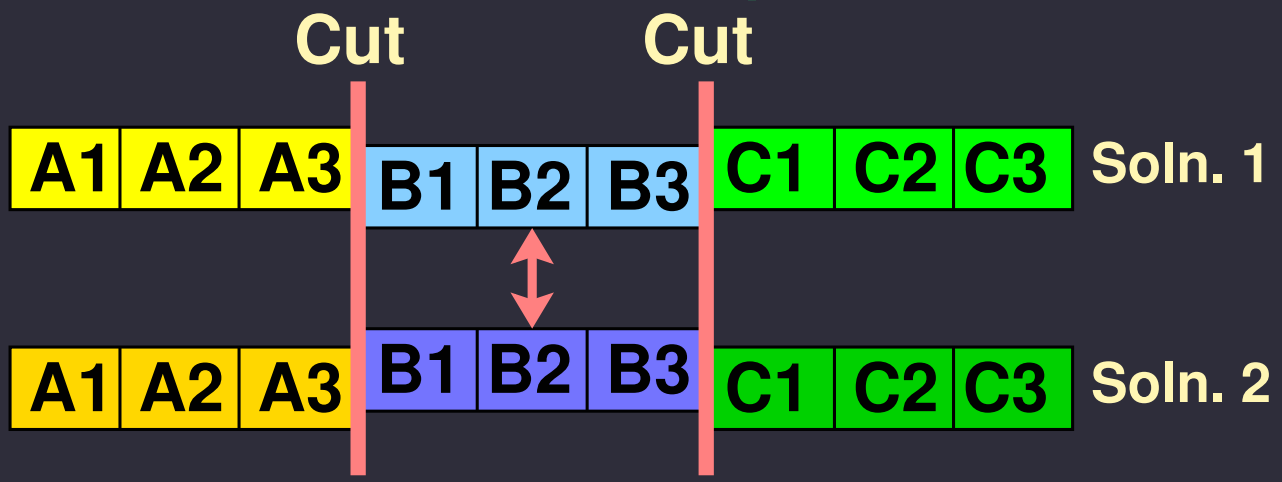
Cluster genetic operator constraints motivation



Cluster genetic operator constraints

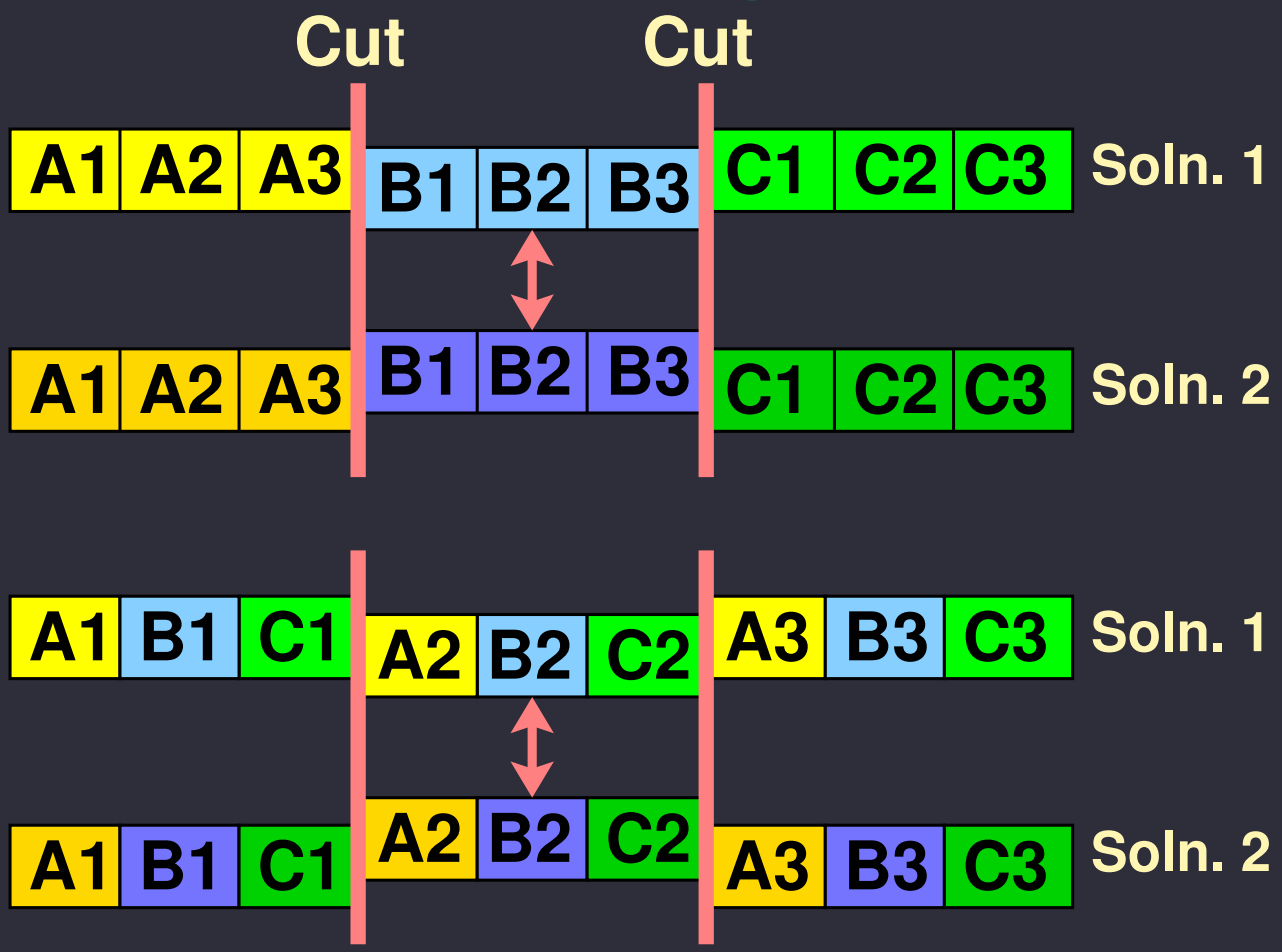


Locality in solution representation

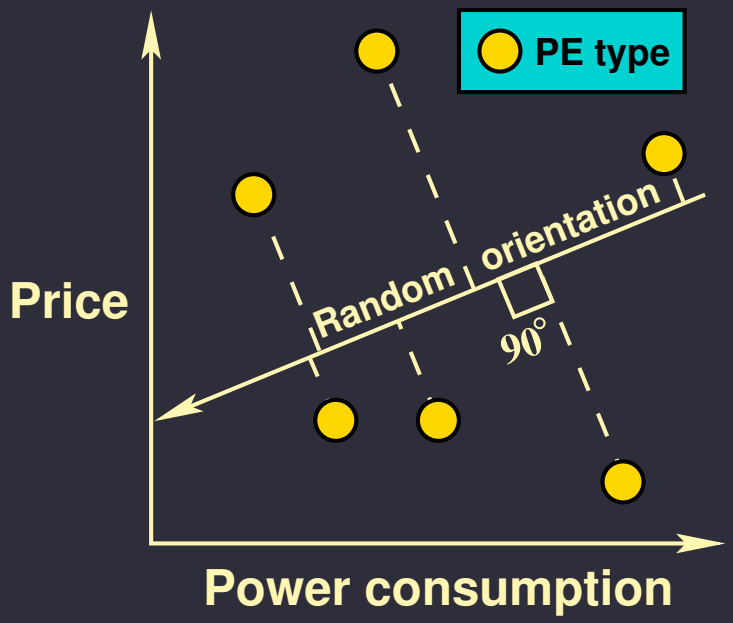


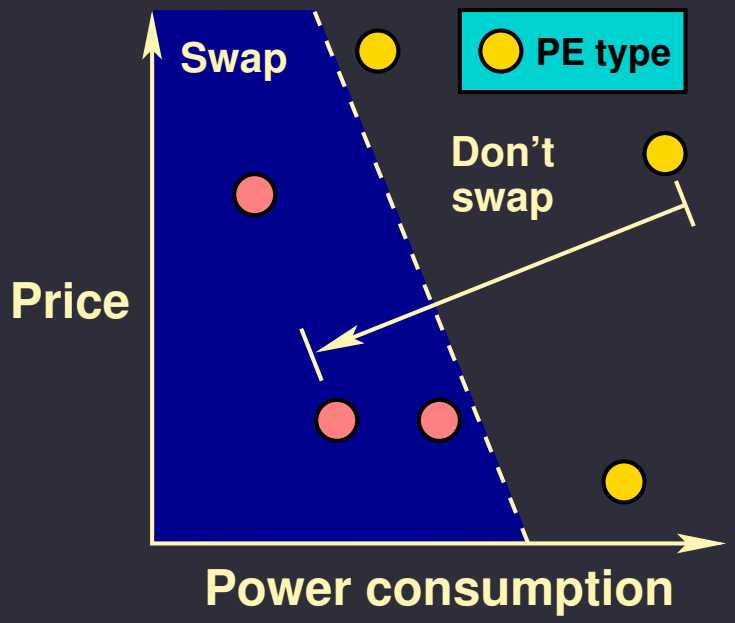
A, B, and C attributes each solve sub-problems

Locality in solution representation

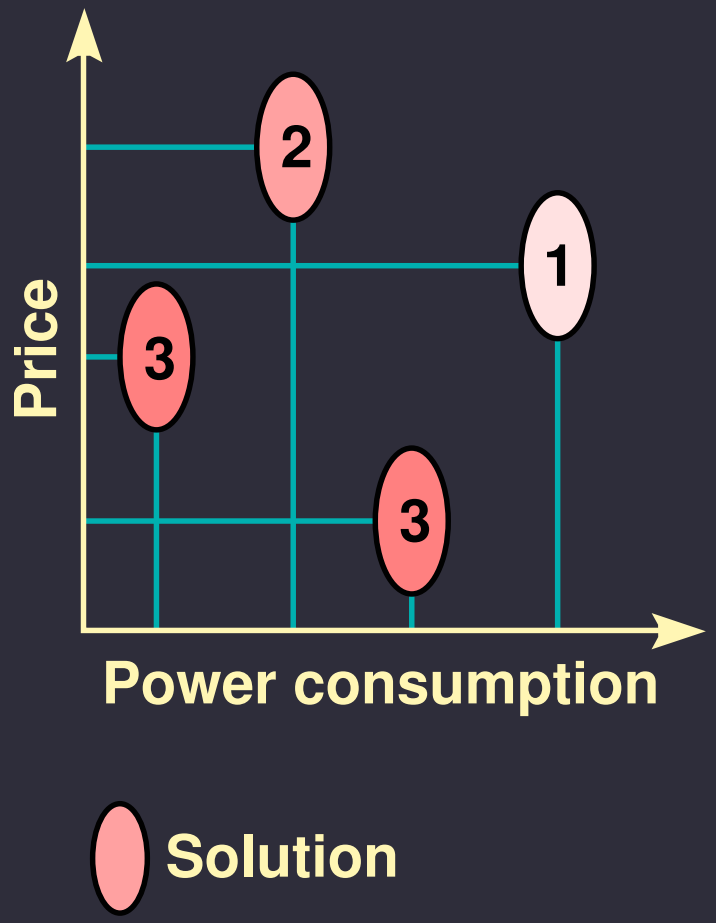


Information trading





Ranking



A solution dominates another

if all its costs are lower, i.e.,

$\mathrm{dom}_{\mathrm{a,b}} =$

$\forall_{i=1}^{n} cost_{a,i} < cost_{b,i} \wedge a \neq b$

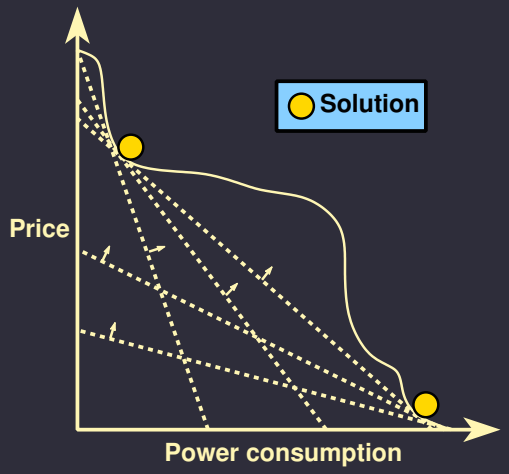
A solution's rank is the number

of other solutions which do not

dominate it, i.e.,

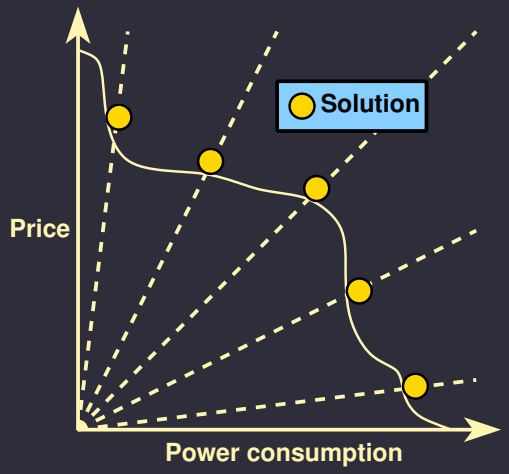
$\mathrm{rank}_{\mathrm{s'}} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{not\ dom}_{\mathrm{s_i,s'}}$

Multiobjective optimization



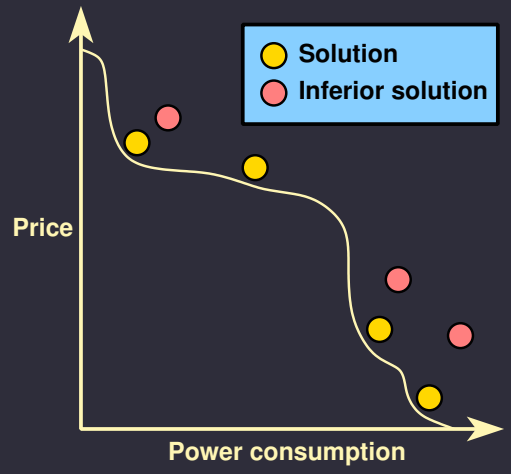
Linear cost functions

$\sum_{i=1}^{n} wt_i \cdot cost_i$



Non-linear cost functions

$\max_{i=1}^{n} wt_i \cdot cost_i$



Pareto-rank cost function

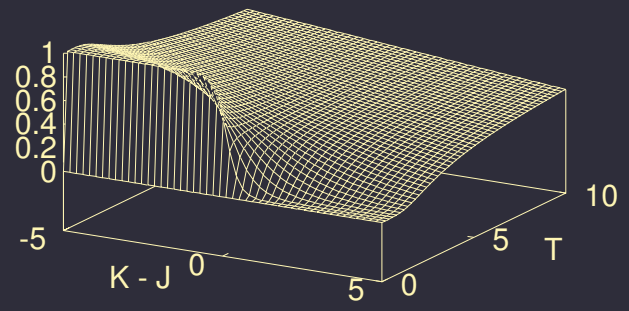
$\sum_{i=1}^{n} \text{not dom}_{s_i,s'}$

Reproduction

Solution are selected for reproduction by conducting Boltzmann trials between parents and children.

Given a global temperature T, a solution with rank J beats a solution with rank K with probability:

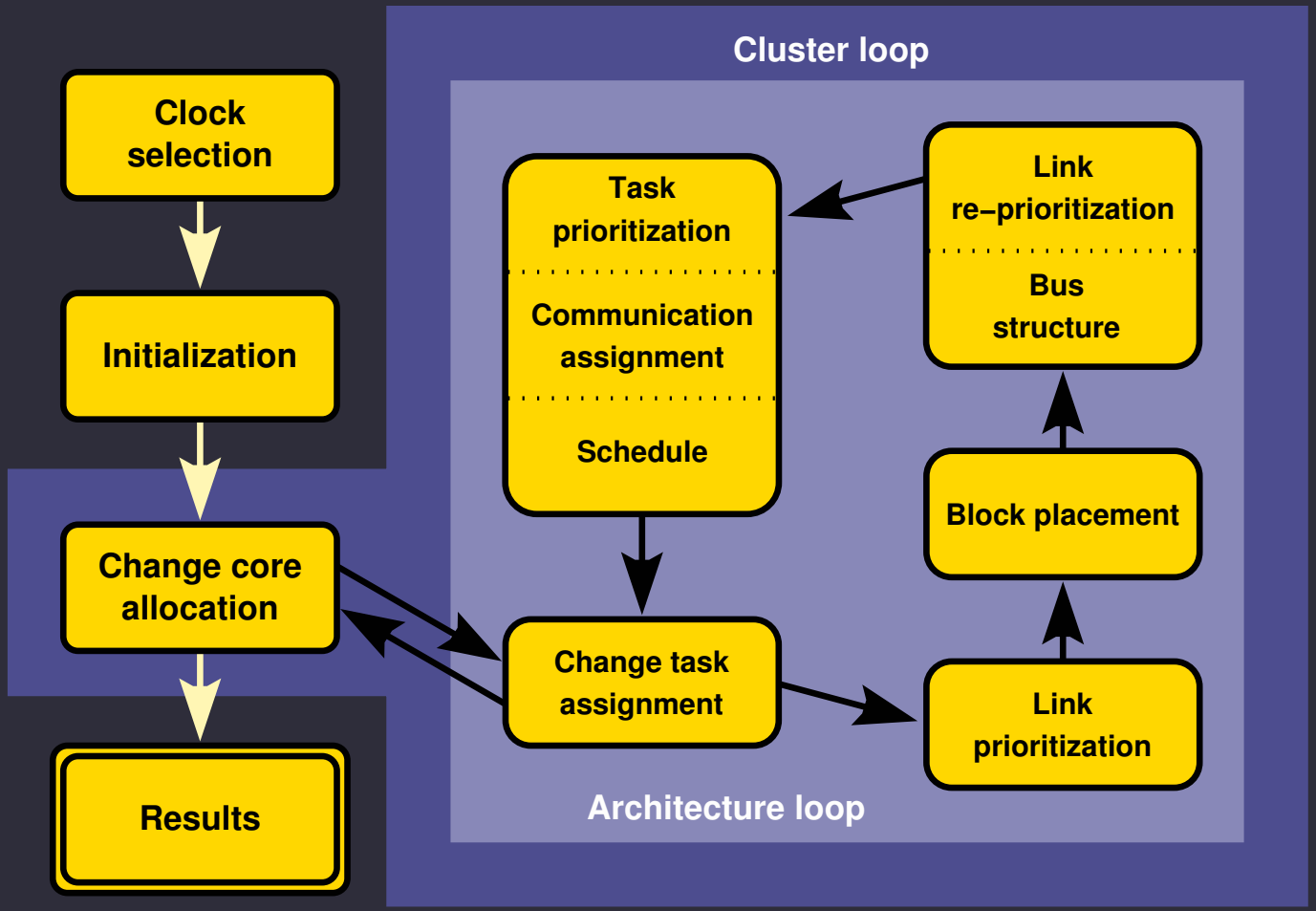
$$\frac{1}{1 + e^{(K\text{-}J)/T}}$$



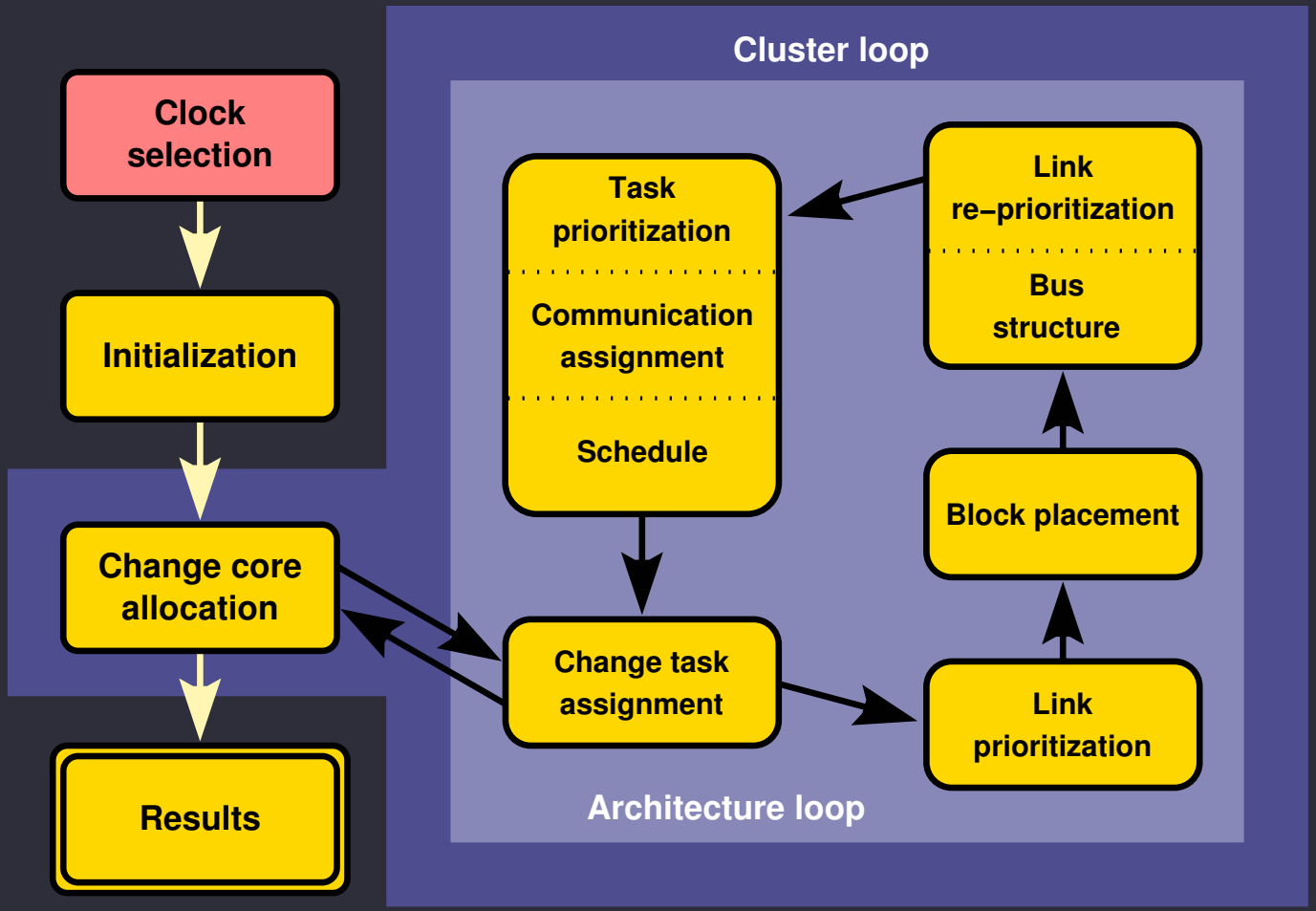
MOCSYN related work

- Floorplanning block placement – Fiduccia and Mattheyses, 1982 – Stockmeyer, 1983
- Parallel recombinative simulated annealing – Mahfoud and Goldberg, 1995
- Linear interpolating clock synthesizers – Bazes, Ashuri, and Knoll, 1996
- Interconnect performance estimation models – Cong & Pan, 2001

MOCSYN algorithm overview



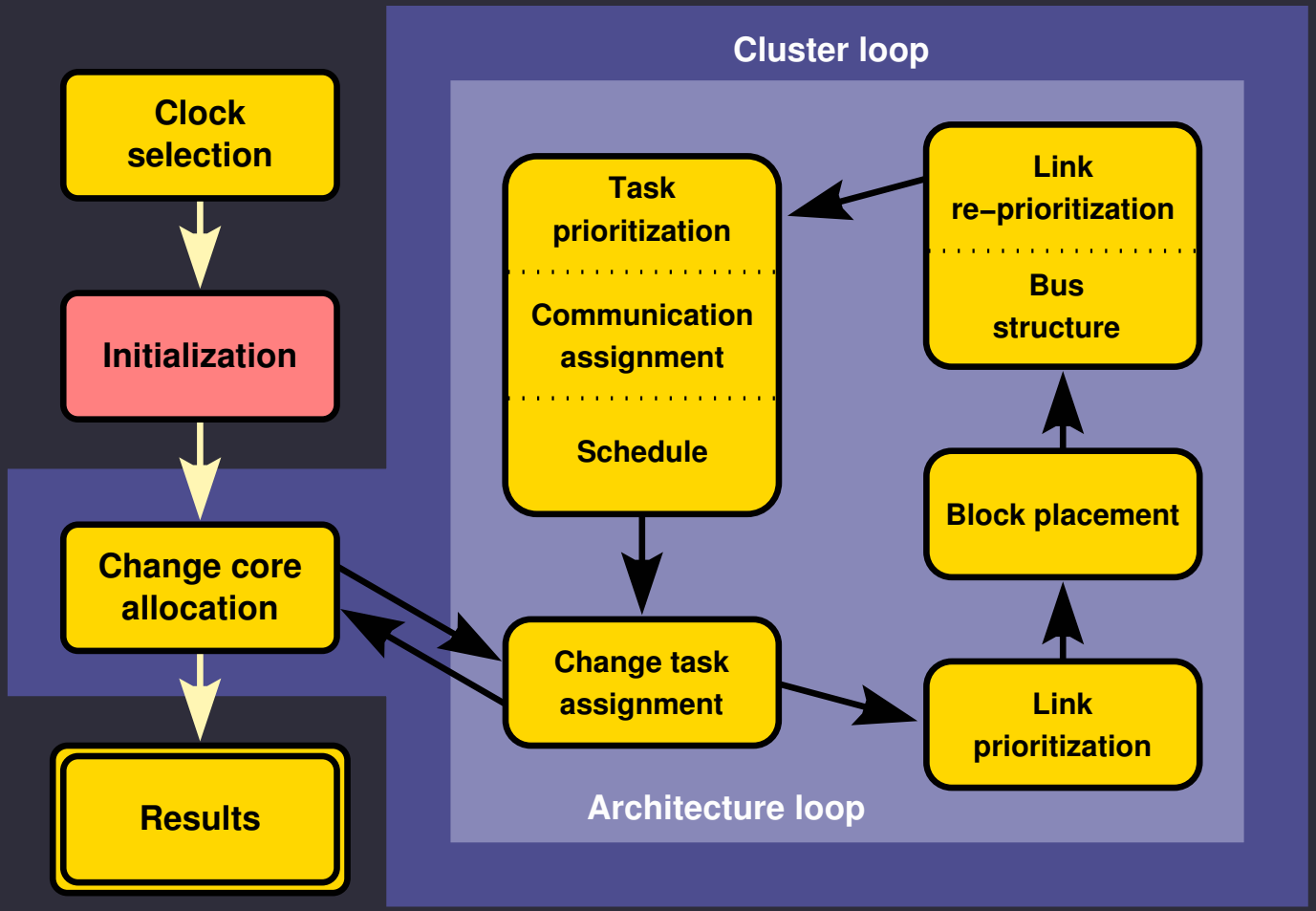
MOCSYN algorithm overview



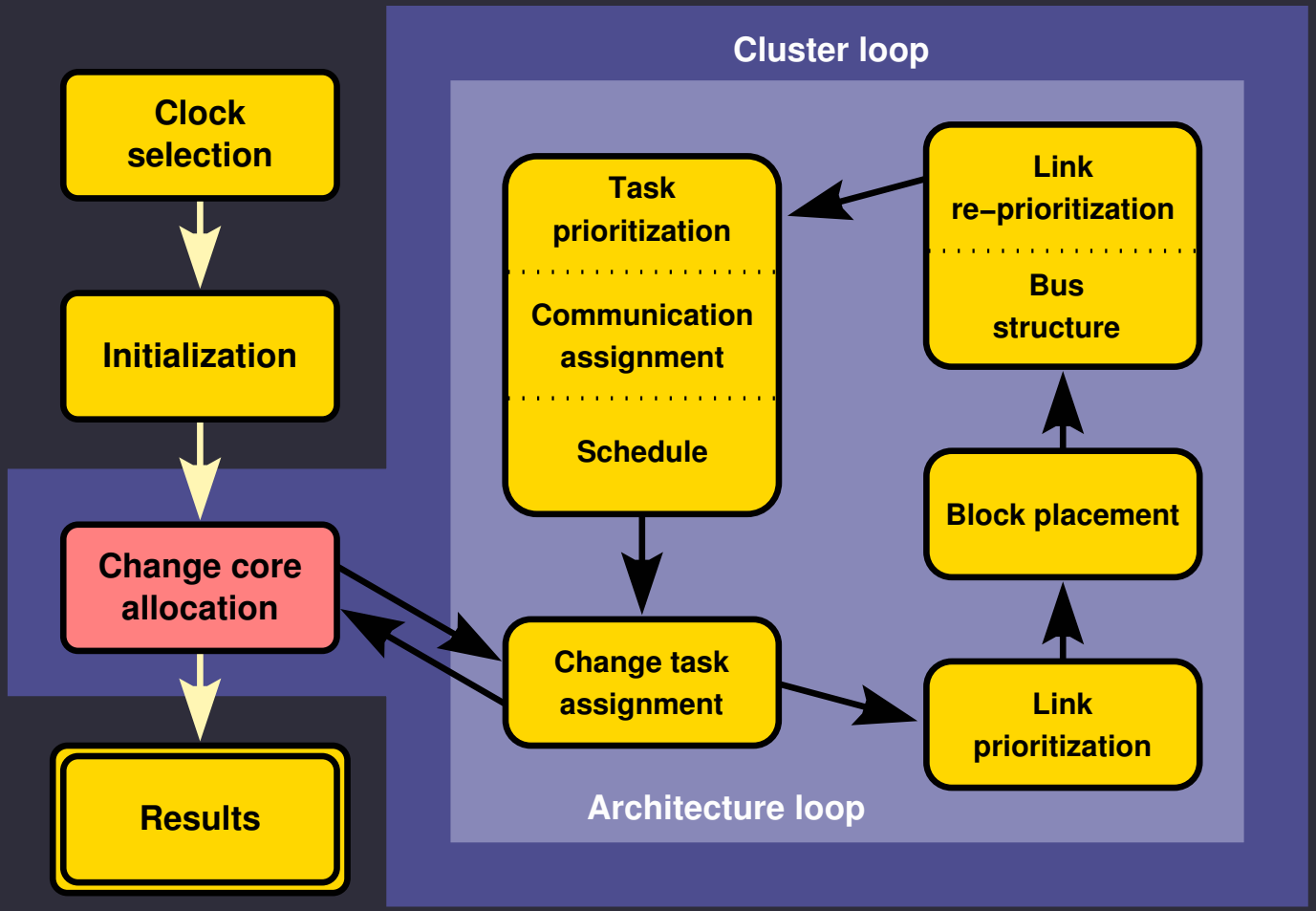
Clock selection

- Cores have different maximum frequencies
- Globally synchronous system forces underclocking
- Multiple crystals too expensive
- Use linear interpolating clock synthesizers
  - Standard CMOS process
  - Each core runs near highest speed
  - Global clock frequency can be low to reduce power
- Optimal clock selection algorithm in pre-pass

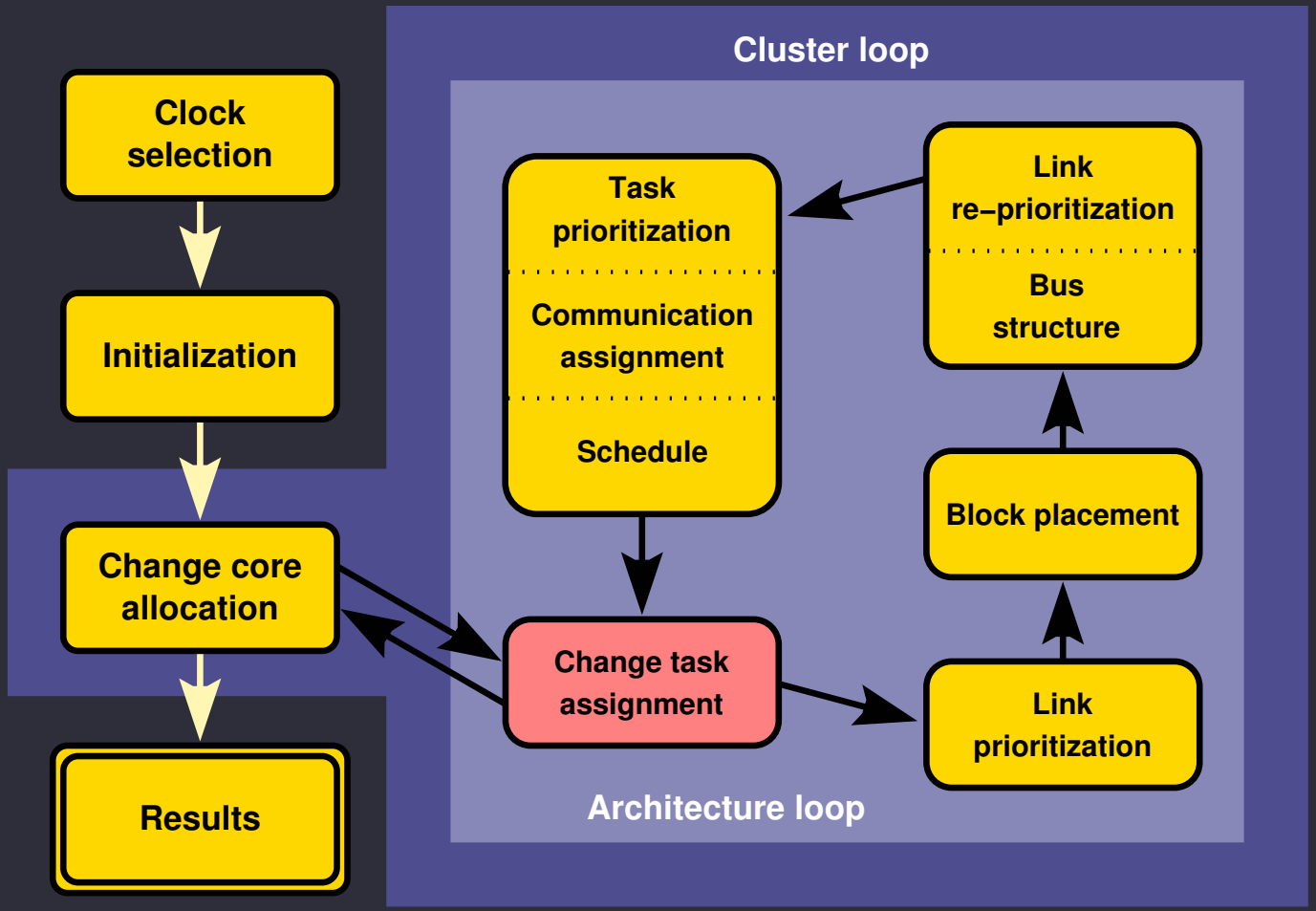
MOCSYN algorithm overview



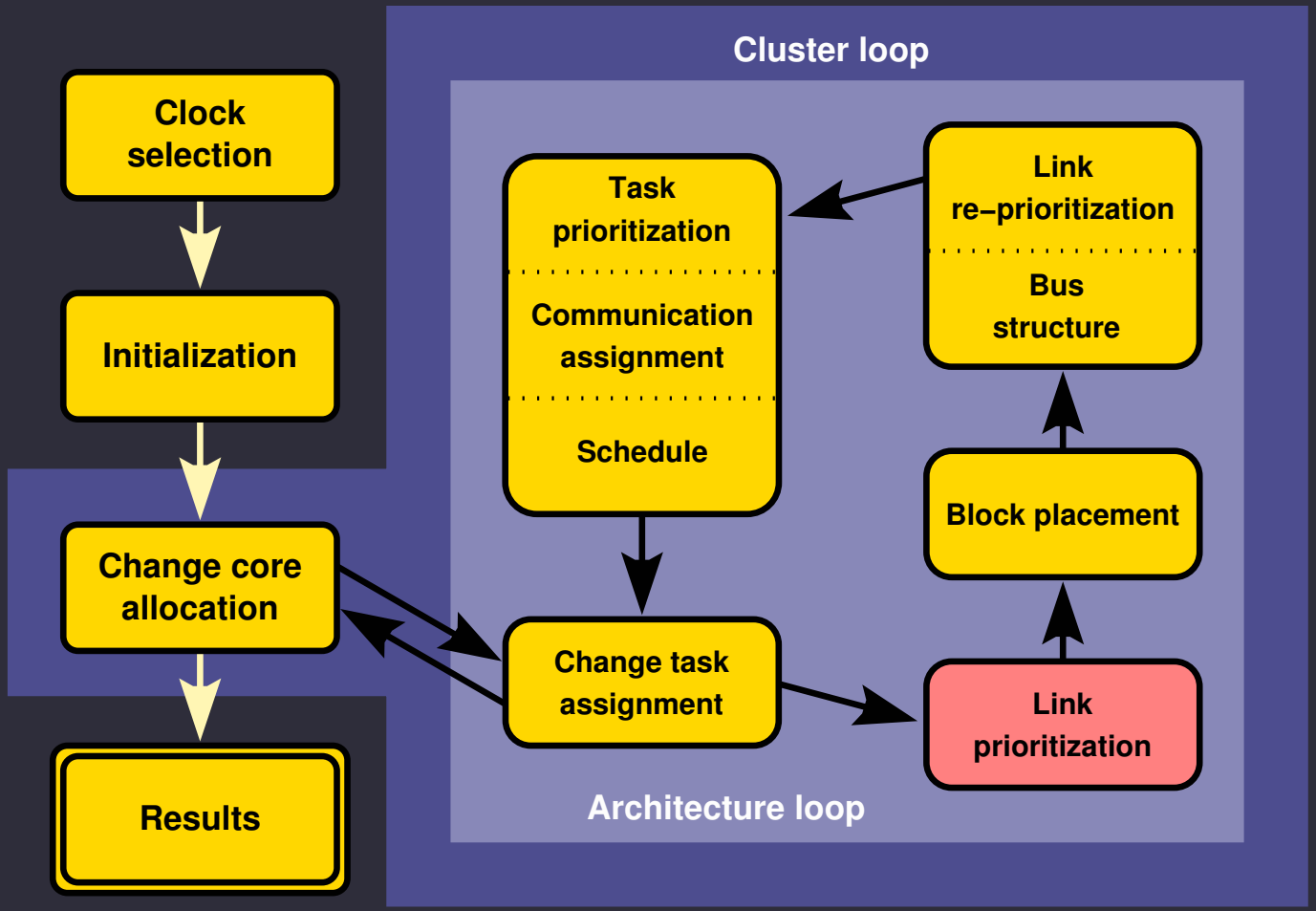
MOCSYN algorithm overview



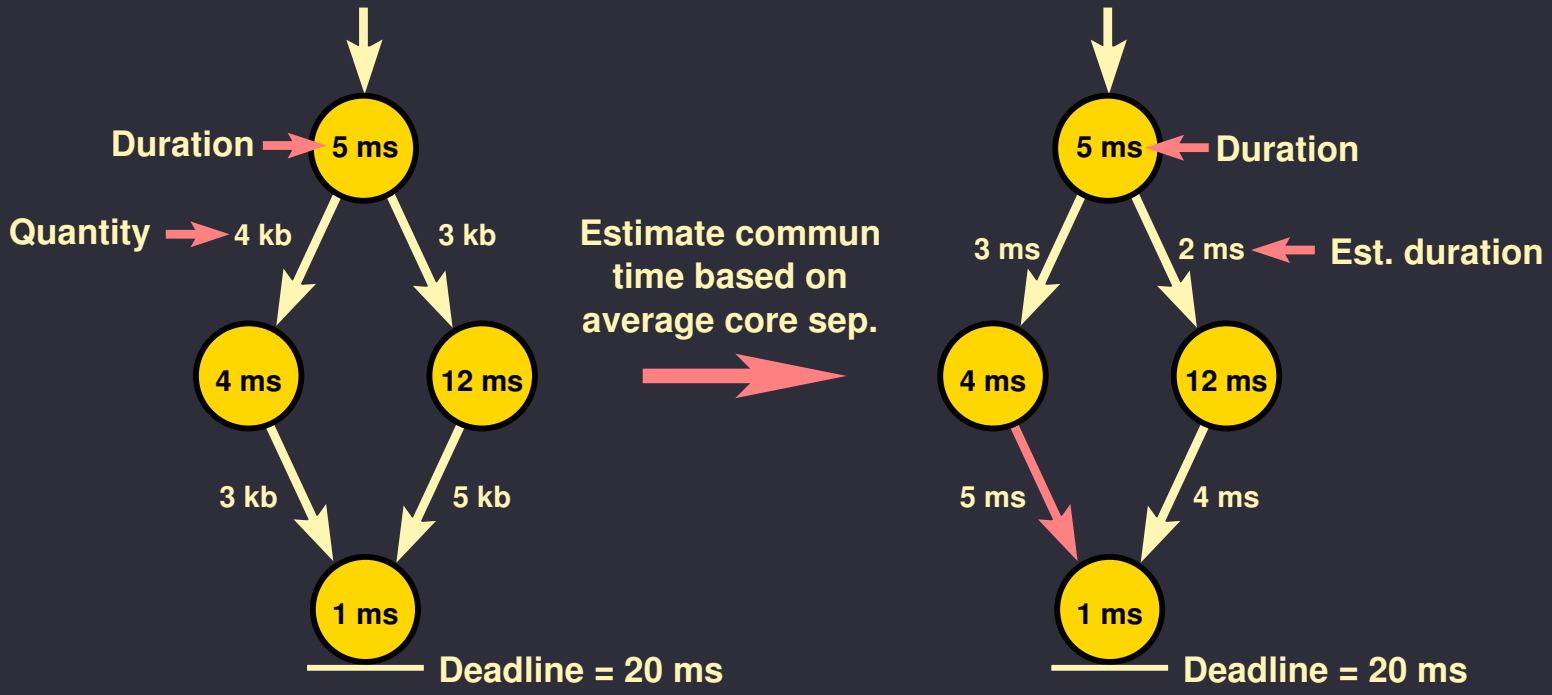
MOCSYN algorithm overview



MOCSYN algorithm overview

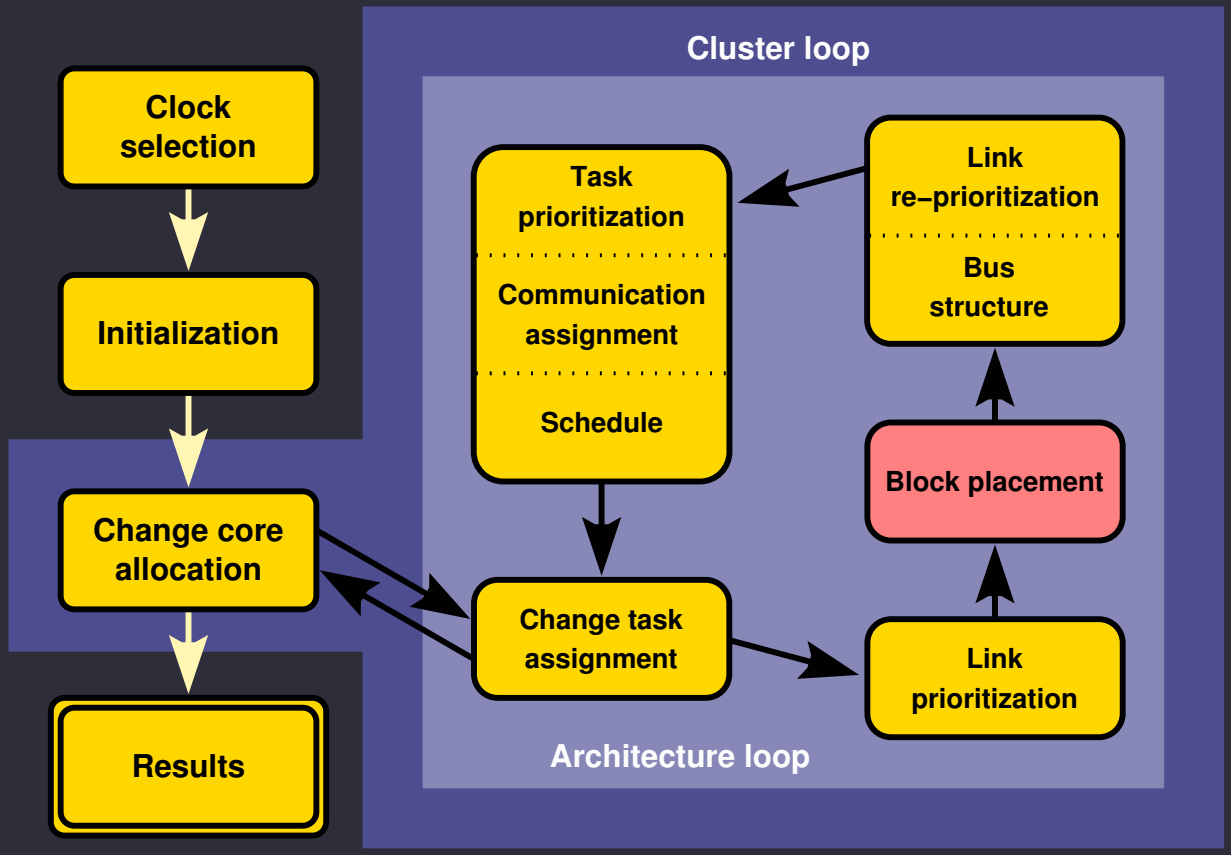


Link prioritization



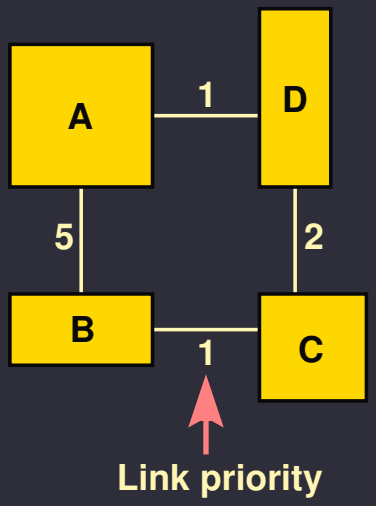
Slack = 2 ms
Priority = –2

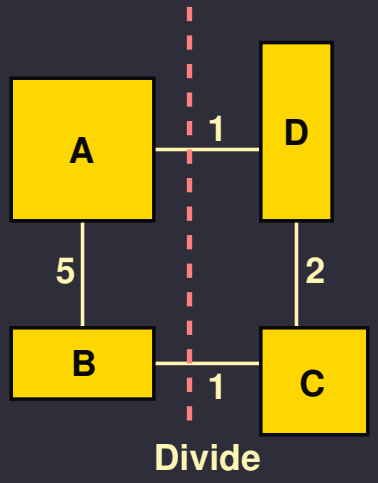
MOCSYN algorithm overview



Block placement to determine communication time, energy

Floorplanning block placement



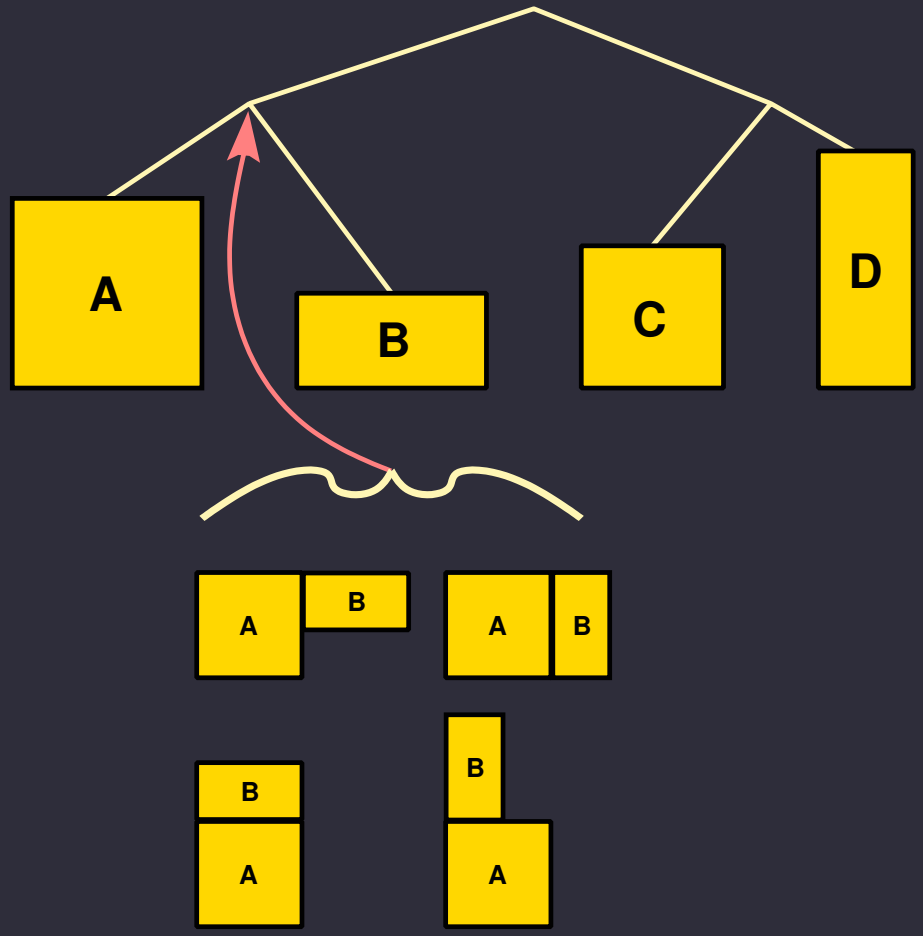


Balanced binary tree of cores formed

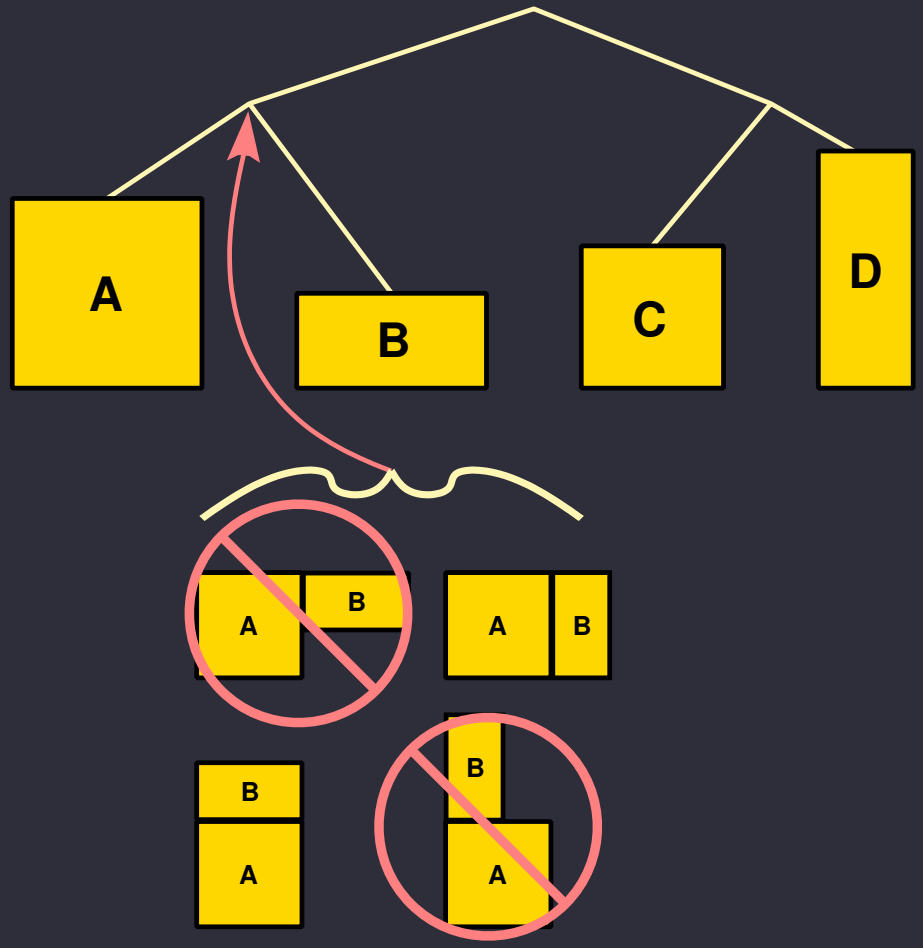
Division takes into account:

- Link priorities
- Area of cores on each side of division

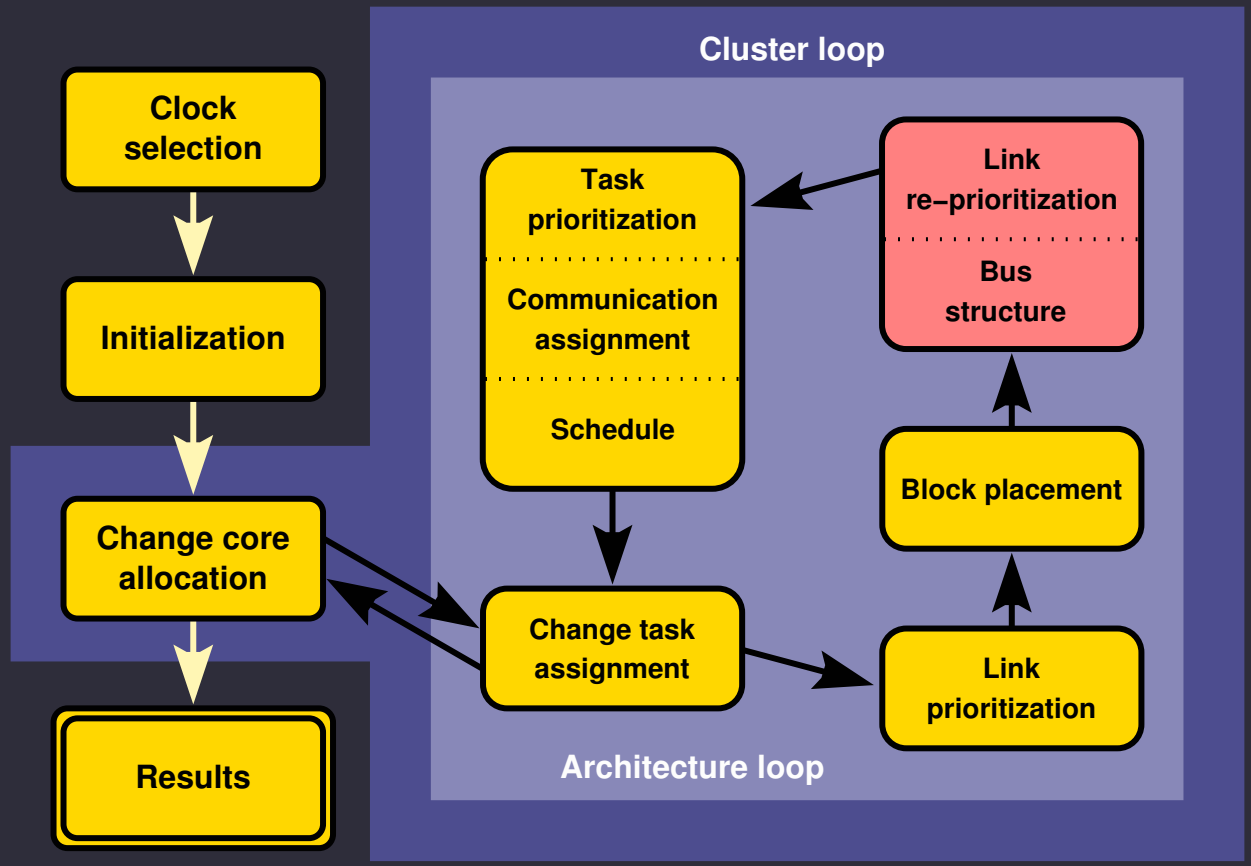
Floorplanning block placement



Floorplanning block placement

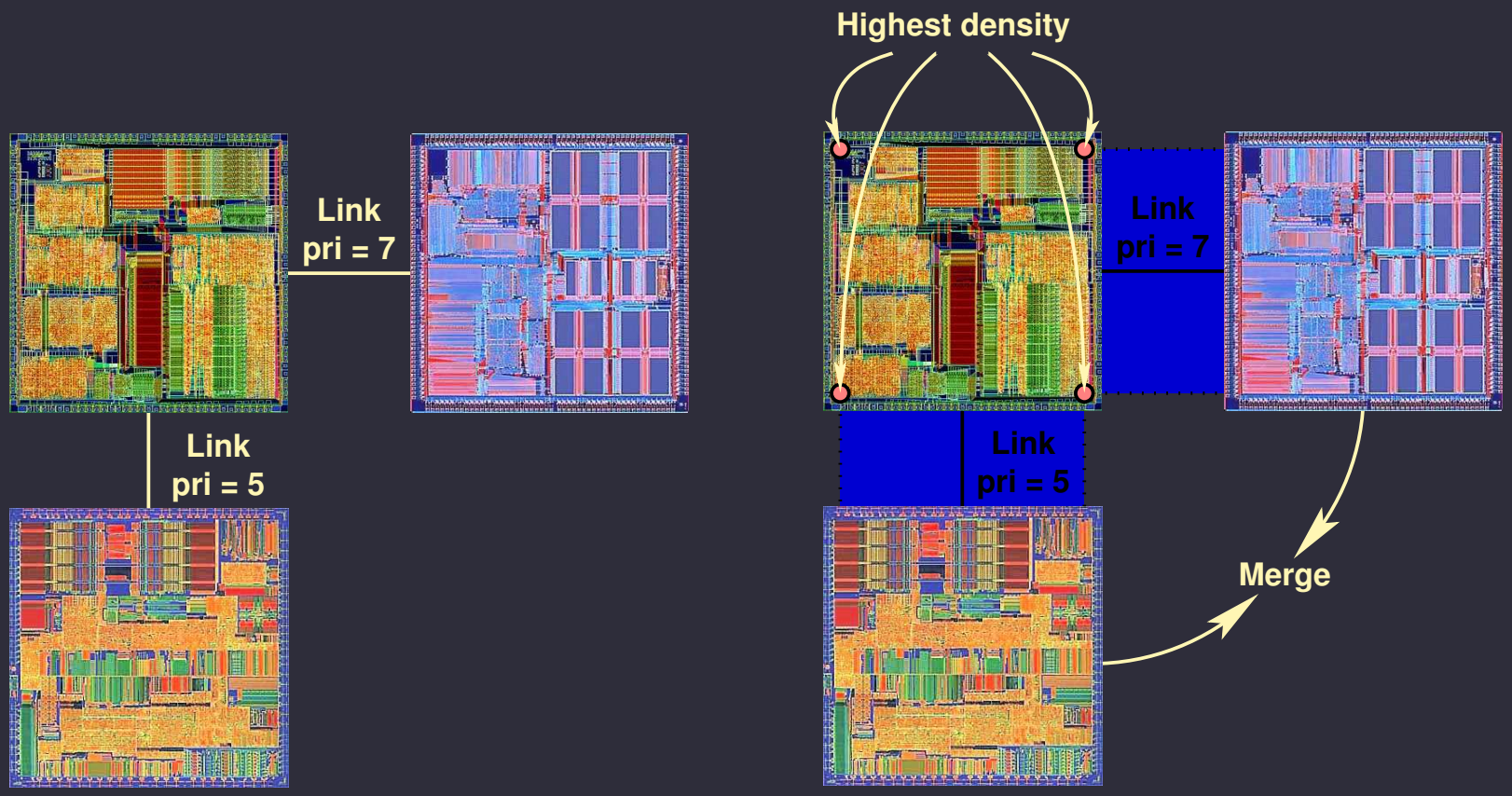


MOCSYN algorithm overview



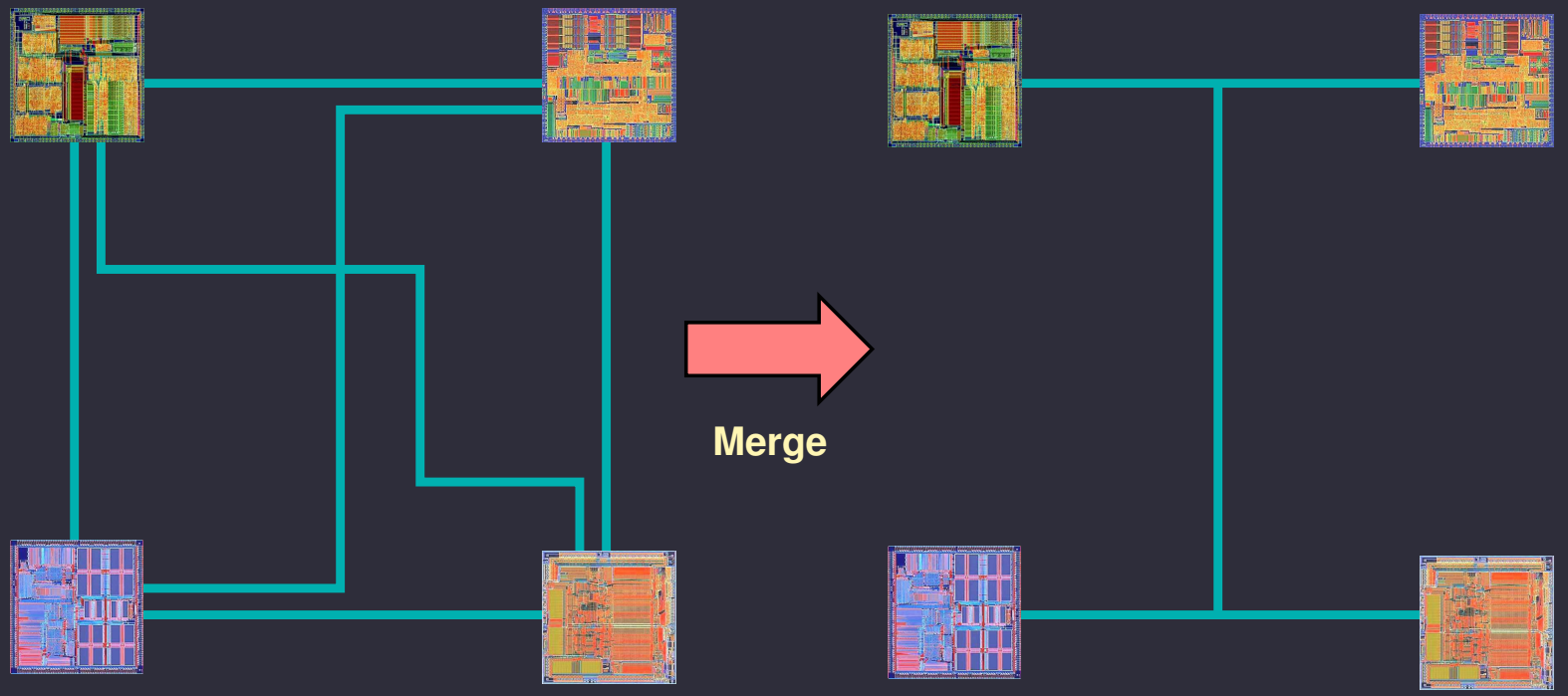
Bus topology generation: minimize contention under routability constraints

Bus formation



Use efficient red-black tree data structure for intersection tests

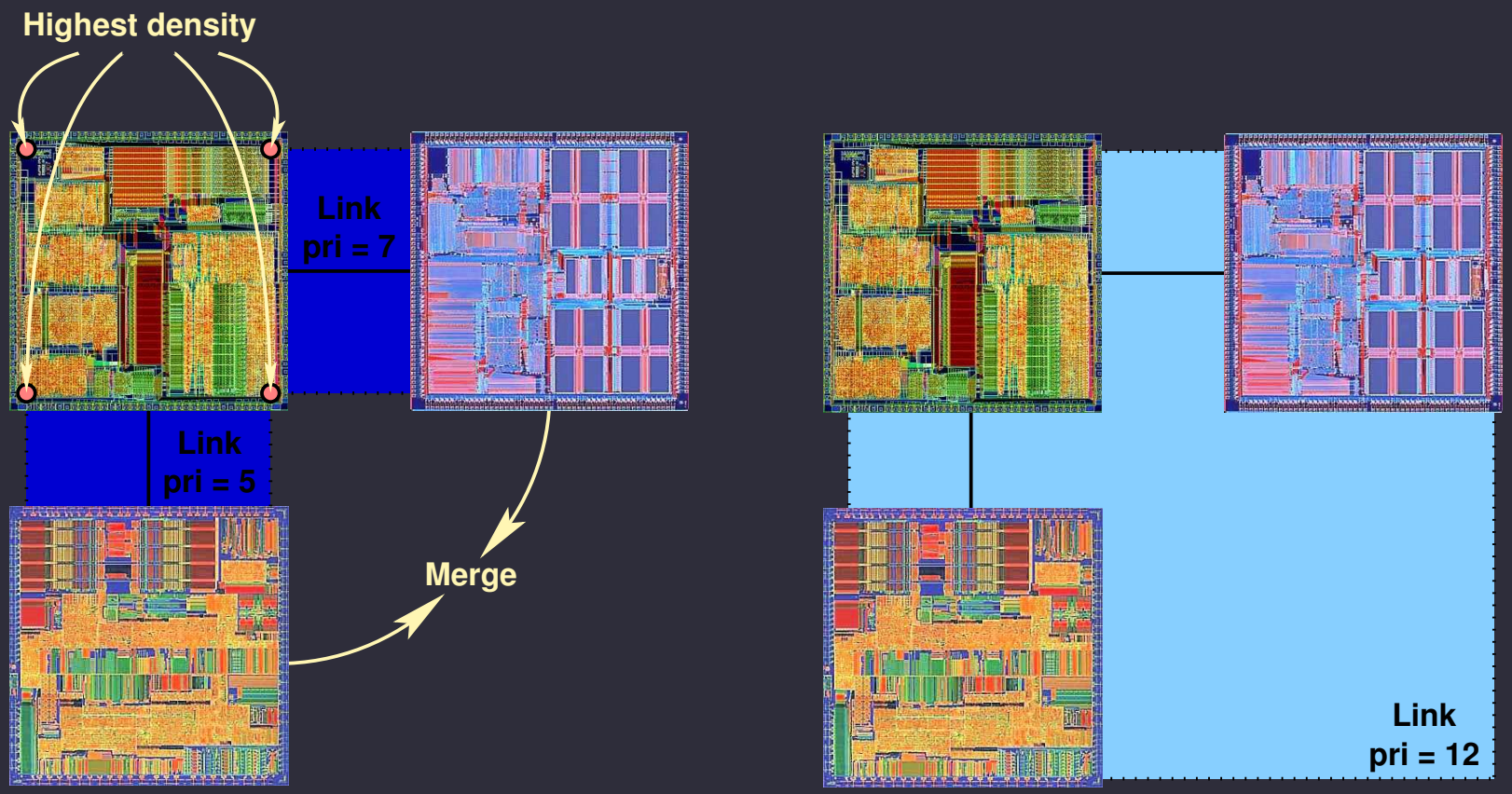
RMST bus length reduction



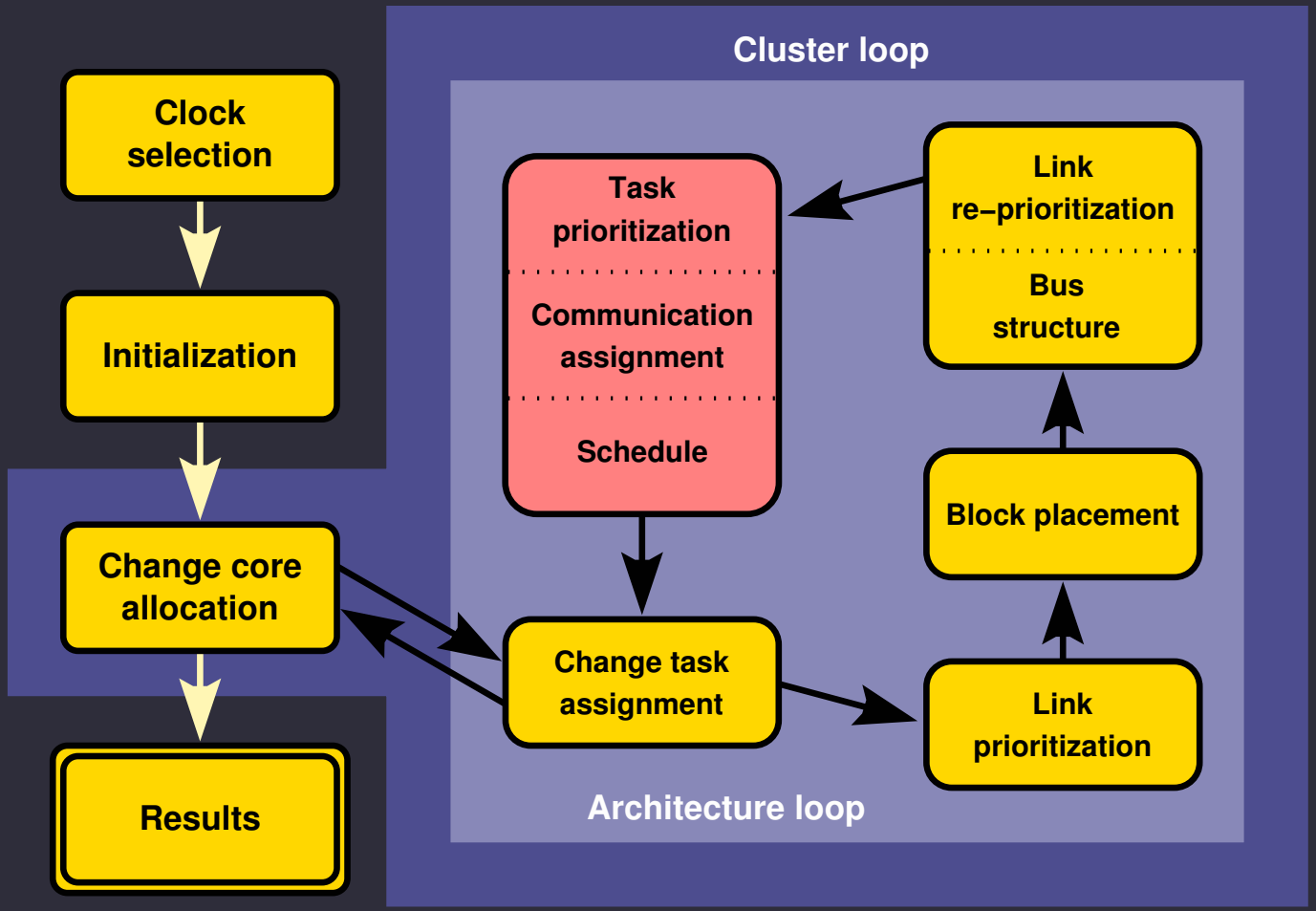
Total length = 5.6 mm

Total length = 2.1 mm

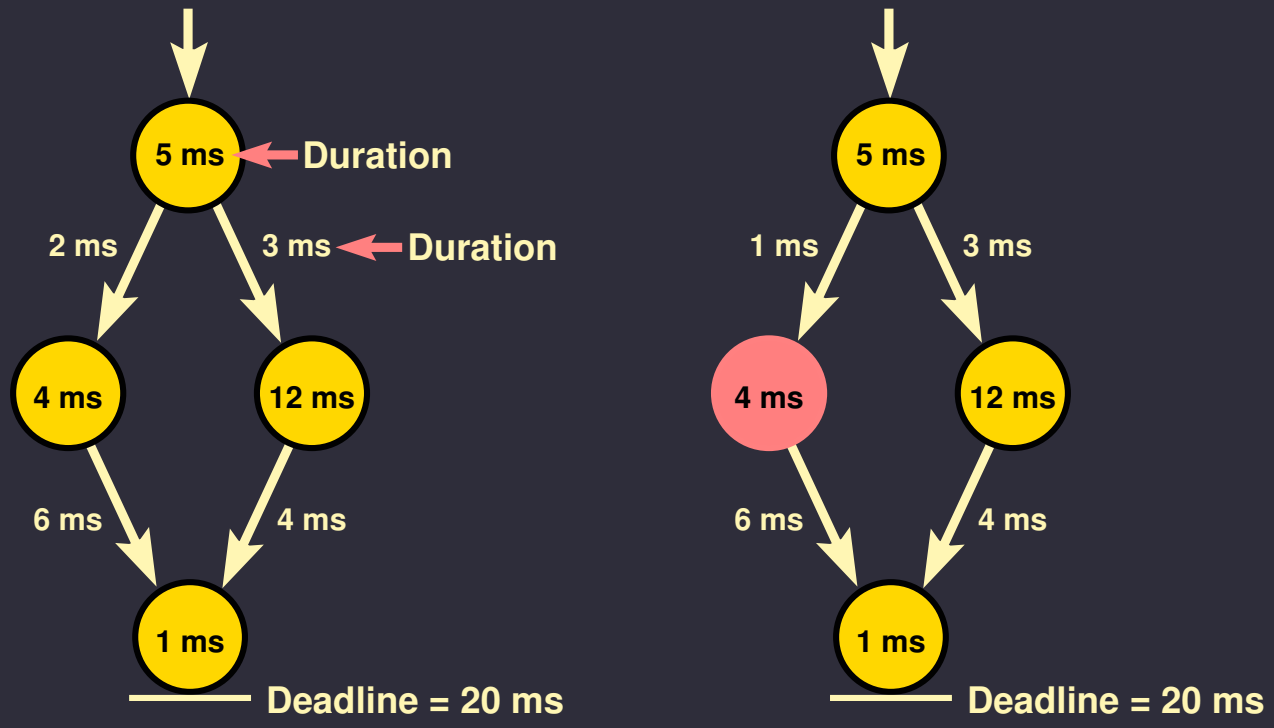
Bus formation



MOCSYN algorithm overview

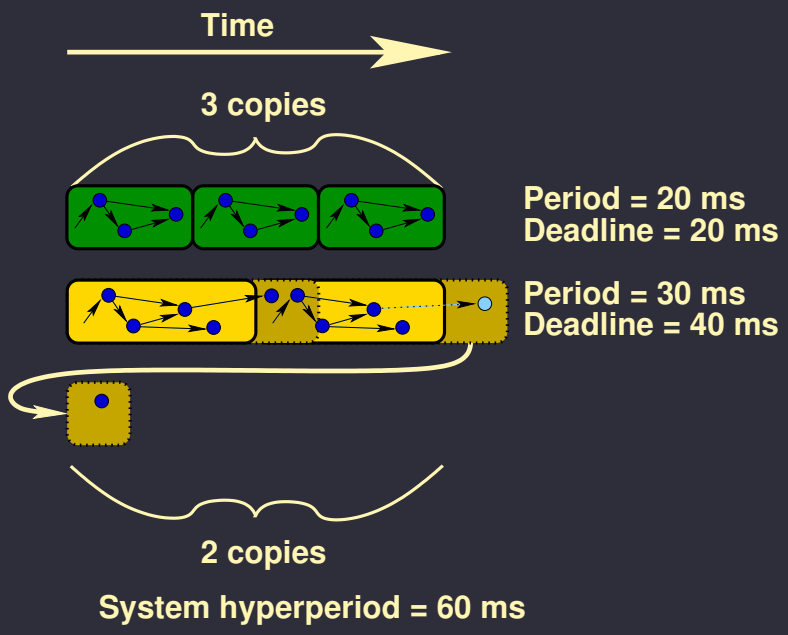


Task prioritization



Slack = 3 ms
Priority = –3

Scheduling

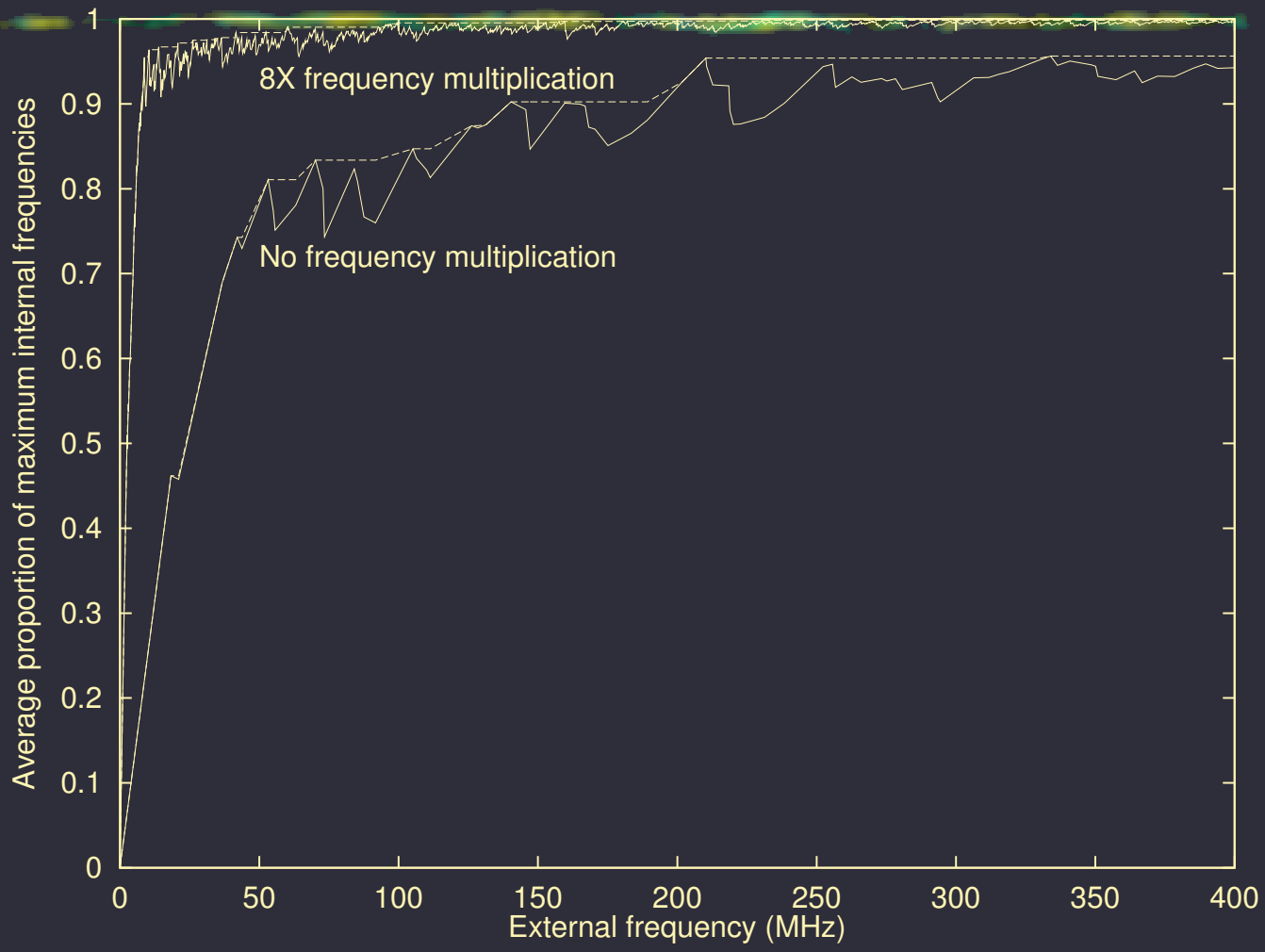


- Fast list scheduler
- Multi-rate
- Handles period $<$ deadline as well as period $\geq$ deadline
- Uses alternative prioritization methods: slack, EST, LFT
- Other features depend on target

Cost calculation

- Price
- Average power consumption
- Area
- PE overload
- Hard deadline violation
- Soft deadline violation
- etc.

Clock selection quality



MOCSYN feature comparisons experiments

Example	MOCSYN price ($)	Worst-case commun. price ($)	Best-case commun. price ($)	Single bus price ($)
...	...	...	...	...
15	216	n.a.	n.a.	n.a.
16	138	n.a.	n.a.	177
17	283	n.a.	n.a.	n.a.
18	253	n.a.	n.a.	253
19	211	n.a.	n.a.	n.a.
...	...	...	...	...
Better		38	44	28
Worse		3	1	9

17 processors, 34 core types, five task graphs, 10 tasks each, 21 task types from networking and telecomm examples.

MOCSYN multiobjective experiments

Example	Price ($)	Average power (mW)	Soft DL viol. prop.	Area (mm^2)
automotive-industrial	91	120	0.60	3.0
	91	120	0.61	2.0
	110	113	0.88	4.0
	110	115	0.60	4.0
networking	61	72	0.94	38.4
telecomm	223	246	2.31	9.9
	223	246	2.76	6.0
	233	255	3.47	4.5
	236	247	2.29	9.9
	236	249	2.60	8.0
	242	221	2.67	3.0
	242	230	2.44	25.9
	242	237	1.72	6.0
	272	226	2.22	192.1
	272	226	2.34	9.4
	353	258	1.23	4.0
consumer	134	281	1.40	34.1
	134	281	1.50	21.6
office automation	64	370	0.23	36.8
	66	55	0.00	7.2

MOGAC run on Hou's examples

Example	Yen's System		MOGAC		
	Price ($)	CPU Time (s)	Price ($)	CPU Time (s)	Tuned CPU Time (s)
Hou 1 & 2 (unclustered)	170	10,205	170	5.7	2.8
Hou 3 & 4 (unclustered)	210	11,550	170	8.0	1.6
Hou 1 & 2 (clustered)	170	16.0	170	5.1	0.7
Hou 3 & 4 (clustered)	170	3.3	170	2.2	0.6

Robust to increase in problem complexity.

2 task graphs each example, 3 PE types

Unclustered: 10 tasks per task graph Clustered: approx. 4 tasks per task graph

MOGAC run on Prakash & Parker's examples

Example ⟨Perform⟩	Prakash & Parker's System		MOGAC		
	Price ($)	CPU Time (s)	Price ($)	CPU Time (s)	Tuned CPU Time (s)
Prakash & Parker 1 ⟨4⟩	7	28	7	3.3	0.2
Prakash & Parker 1 ⟨7⟩	5	37	5	2.1	0.1
Prakash & Parker 2 ⟨8⟩	7	4,511	7	2.1	0.2
Prakash & Parker 2 ⟨15⟩	5	385,012	5	2.3	0.1

Quickly gets optimal when getting optimal is tractable.

3 PE types, Example 1 has 4 tasks, Example 2 has 9 tasks

MOGAC run Yen's large random examples

Example	Yen's System		MOGAC		
	Price ($)	CPU Time (s)	Price ($)	CPU Time (s)	Tuned CPU Time (s)
Random 1	281	10,252	75	6.4	0.2
Random 2	637	21,979	81	7.8	0.2

Handles large problem specifications.

No communication links: communication costs = 0

Random 1: 6 task graphs, approx. 20 tasks each, 8 PE types

Random 2: 8 task graphs, approx. 20 tasks each, 12 PE types

MOCSYN contributions, conclusions

First core-based system-on-chip synthesis algorithm

- Novel problem formulation
- Multiobjective (price, power, area, response time, etc.)
- New clocking solution
- New bus topology generation algorithm

Important for system-on-chip synthesis to do

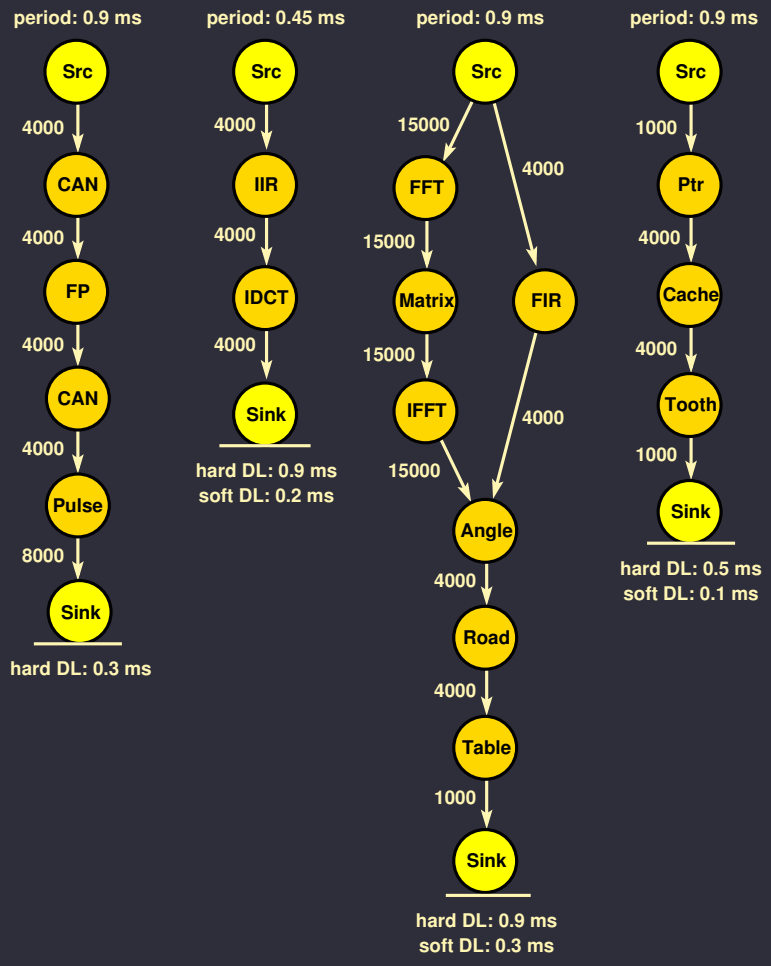
- Clock selection
- Block placement
- Generalized bus topology generation

Research contributions

- **TGFF**: Used by a number of researchers in published work
- **MOGAC**: Real-time distributed embedded system synthesis
  - First true multiobjective (price, power, etc.) system synthesis
  - Solution quality $\geq$ past work, often in orders of magnitude less time
- **CORDS**: First reconfigurable systems synthesis, schedule reordering
- **COWLS**: First wireless client-server systems synthesis, task migration

EEMBC-based embedded benchmarks

Automotive-Industrial



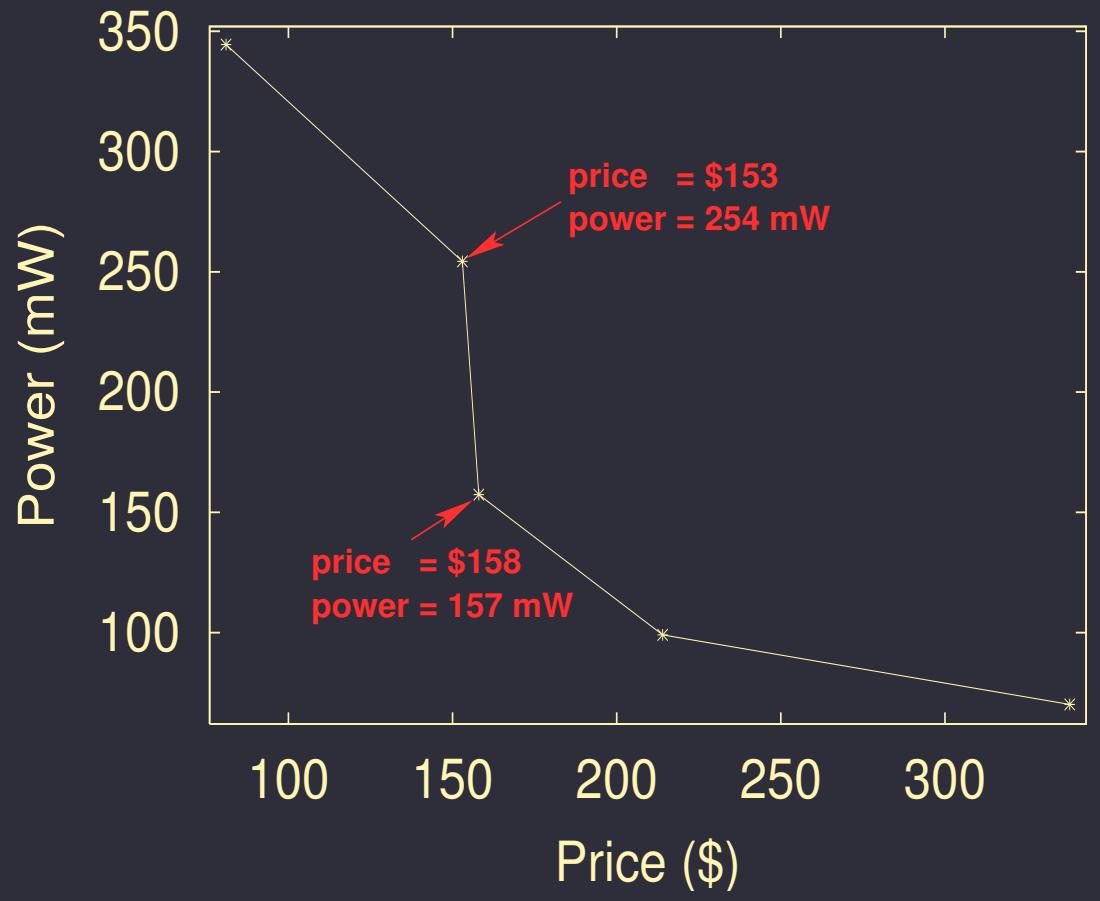
Processors

- AMD ElanSC520 133 MHz
- AMD K6-2 450 MHz
- AMD K6-2E 400MHz/ACR
- AMD K6-2E+ 500MHz/ACR
- AMD K6-IIIE+ 550MHz/ACR
- Analog Devices 21065L 60 MHz
- IBM PowerPC 405GP 266 MHz
- IBM PowerPC 750CX 500 MHz
- IDT32334 100 MHz
- IDT79RC32364 100 MHz
- IDT79RC32V334 150 MHz
- IDT79RC64575 250 MHz
- Imsys Cjip 40 MHz
- Motorola MPC555 40 MHz
- NEC VR5432 167 MHz
- ST20C2 50 MHz
- TI TMS320C6203 300MHz

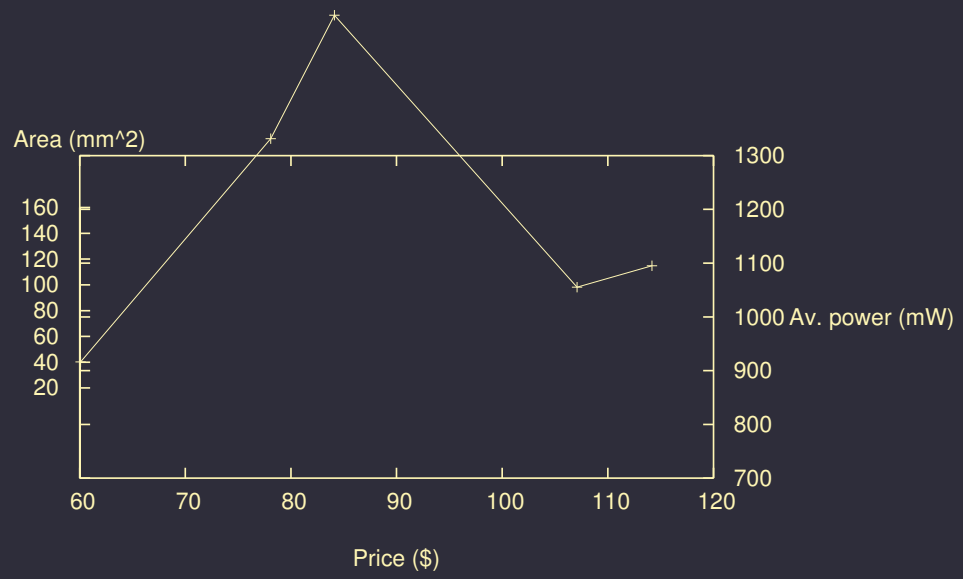
Recently started and future work

- Market-based energy allocation in low-power wireless mobile networks
  - paper under review
- Evolutionary algorithms for multi-dimensional optimization
  - future work
- Task and processor characterization
  - EEMBC-based resource database completed will publicly release
- Tightly coupling low-level, high-level design automation algorithms
  - recently started work in this area

MOGAC run on Yen's second large random example

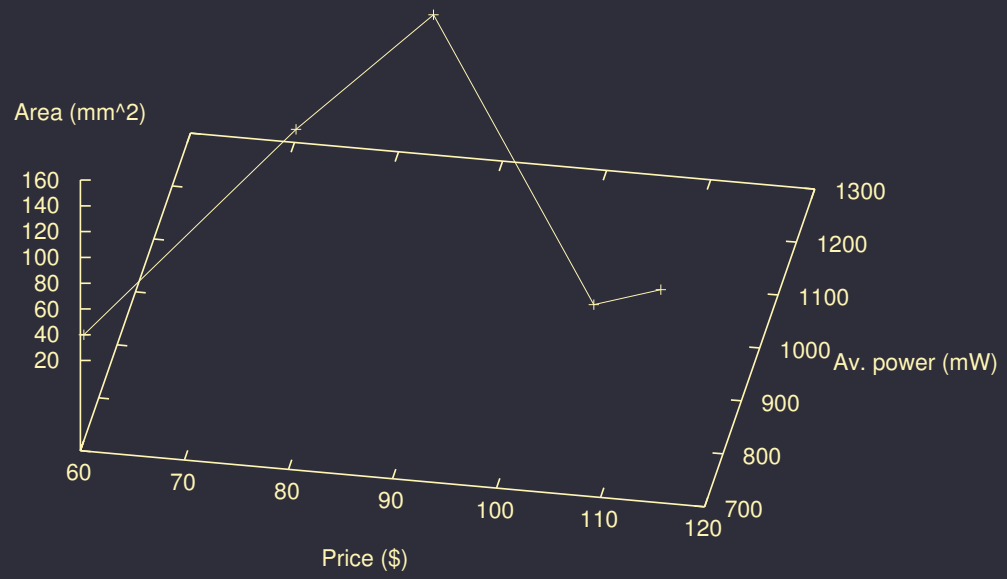


MOCSYN Networking example



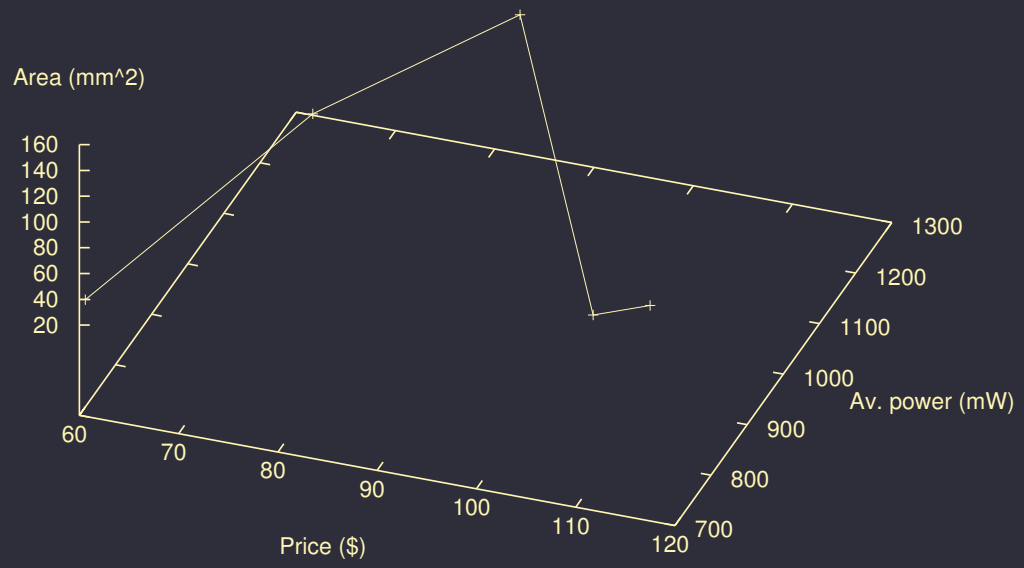
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



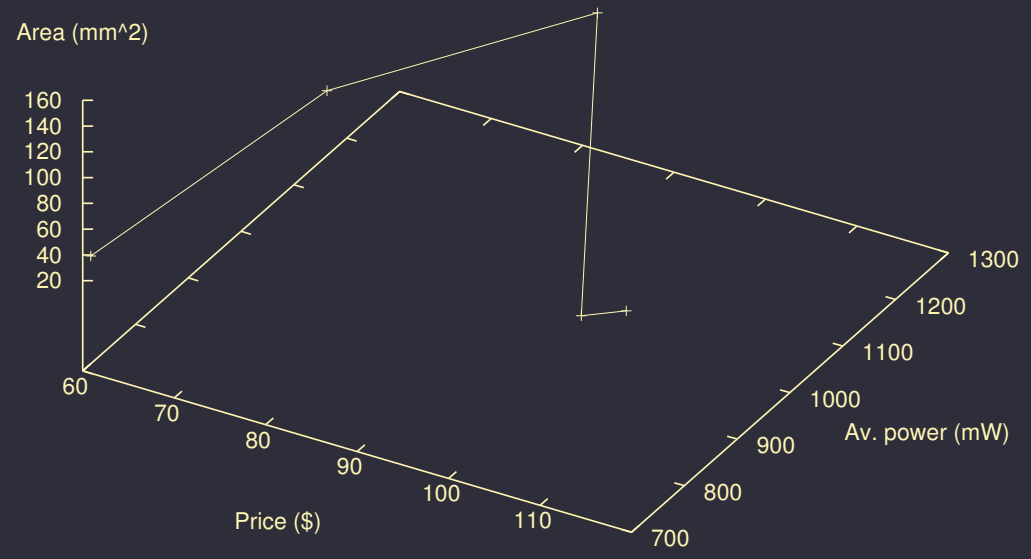
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



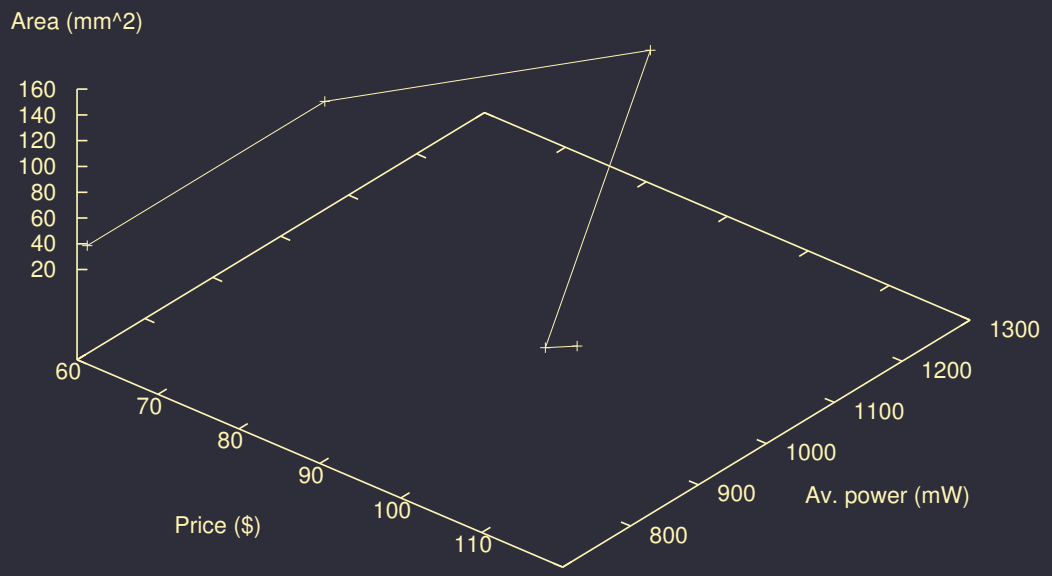
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



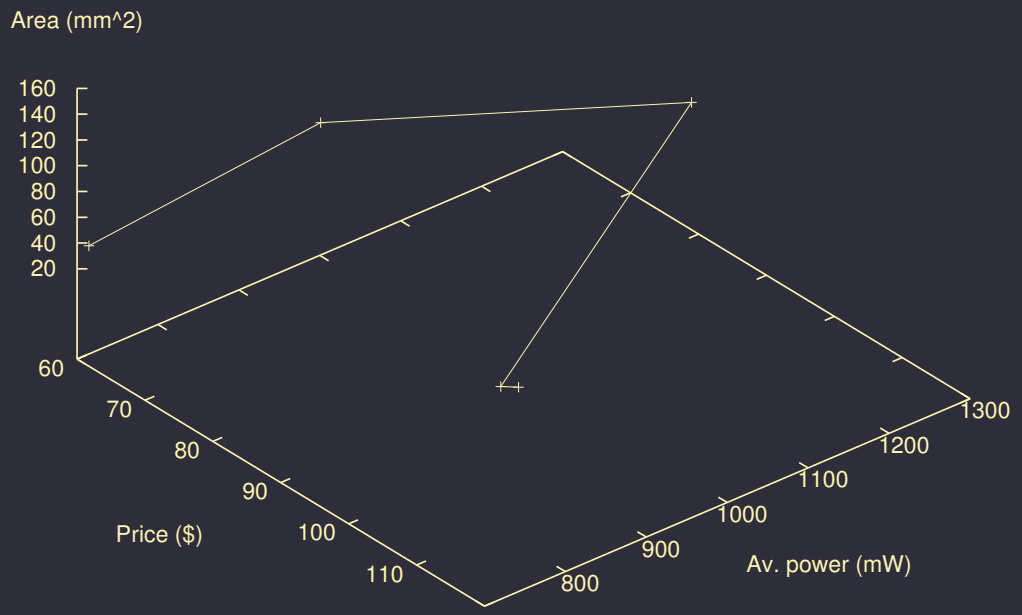
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



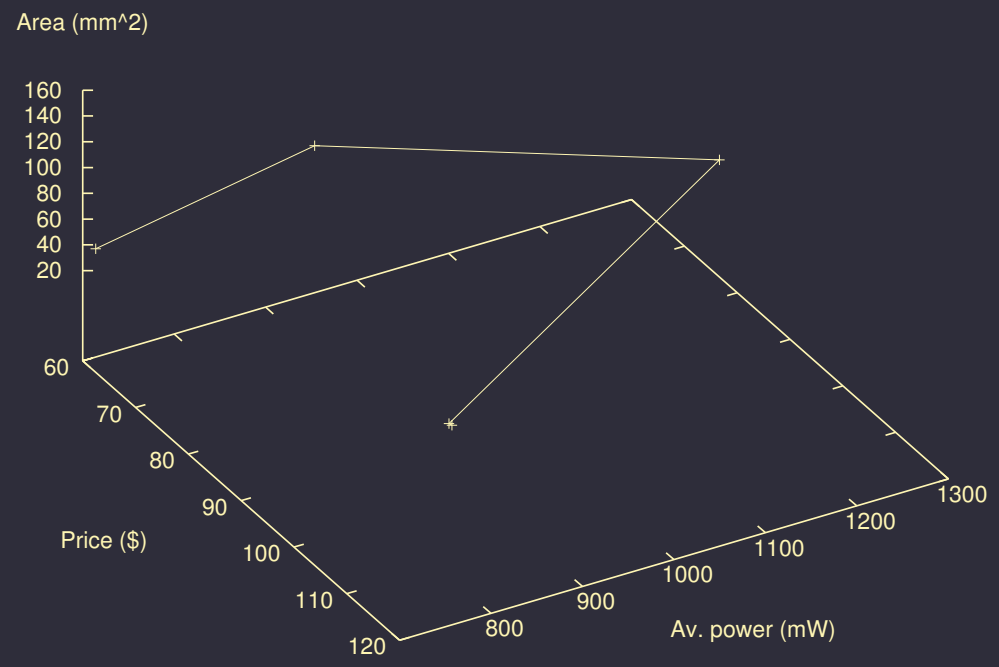
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



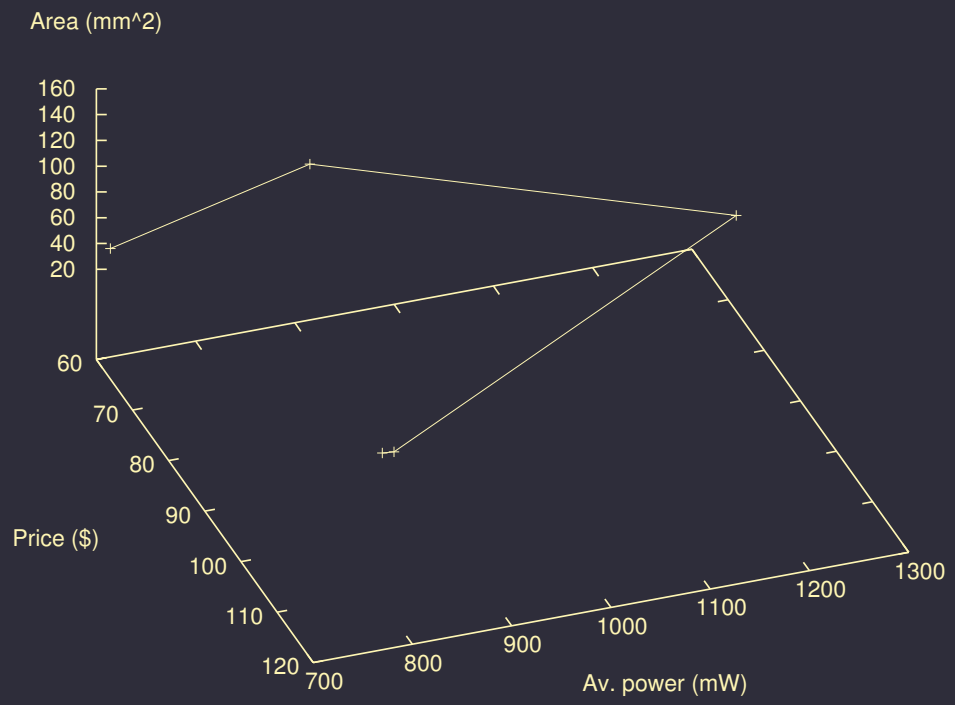
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



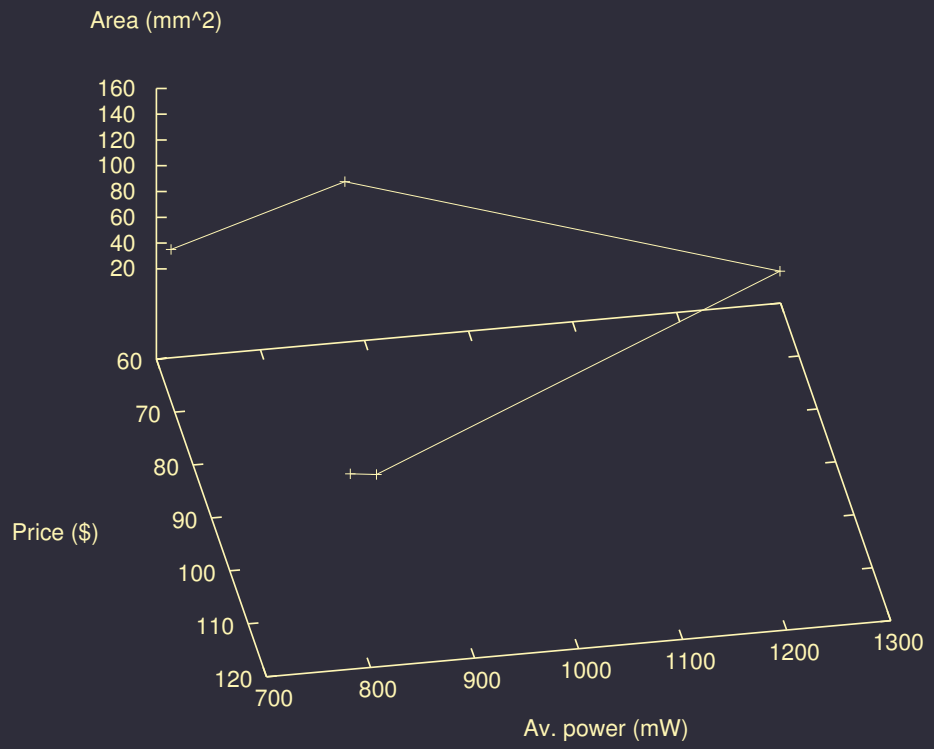
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



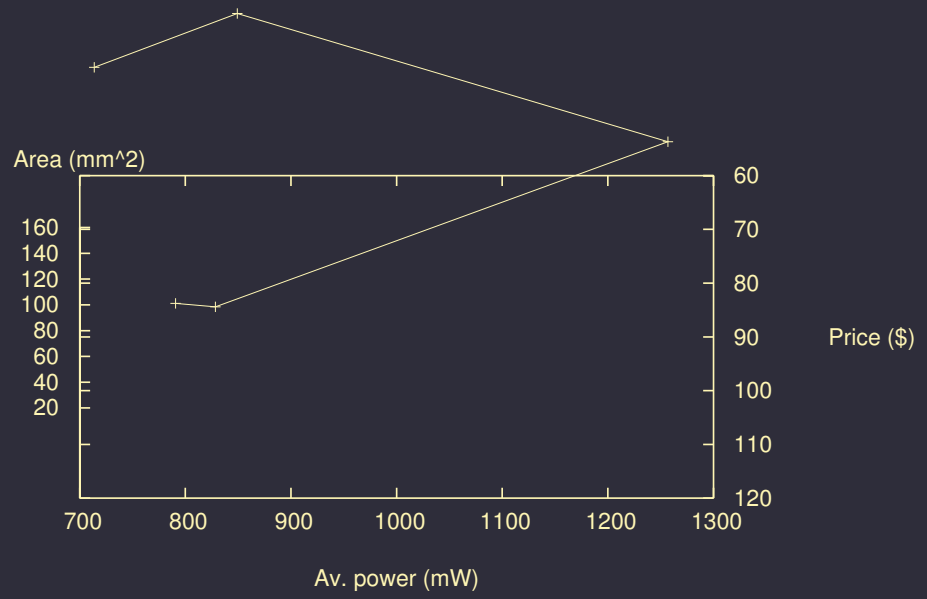
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



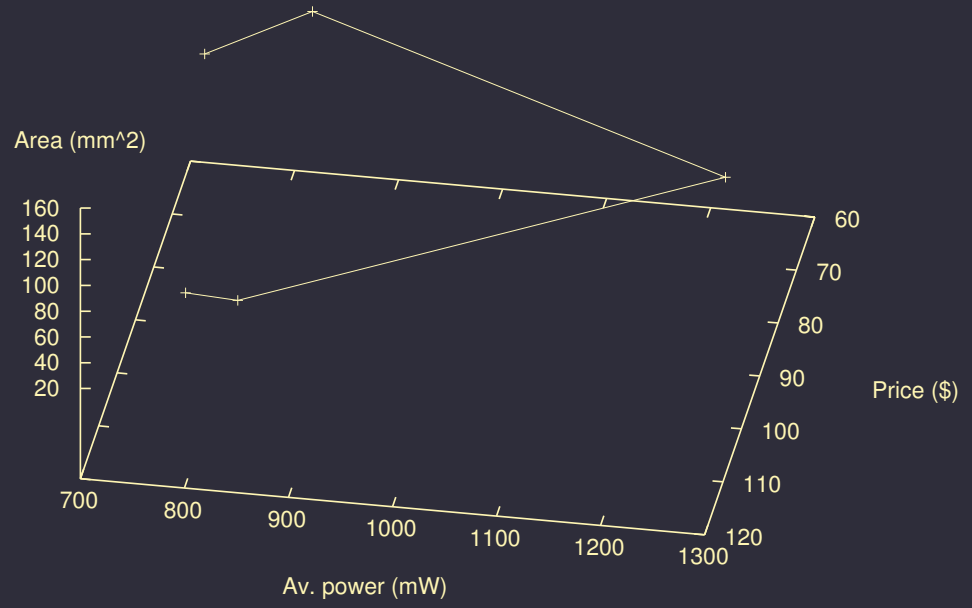
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



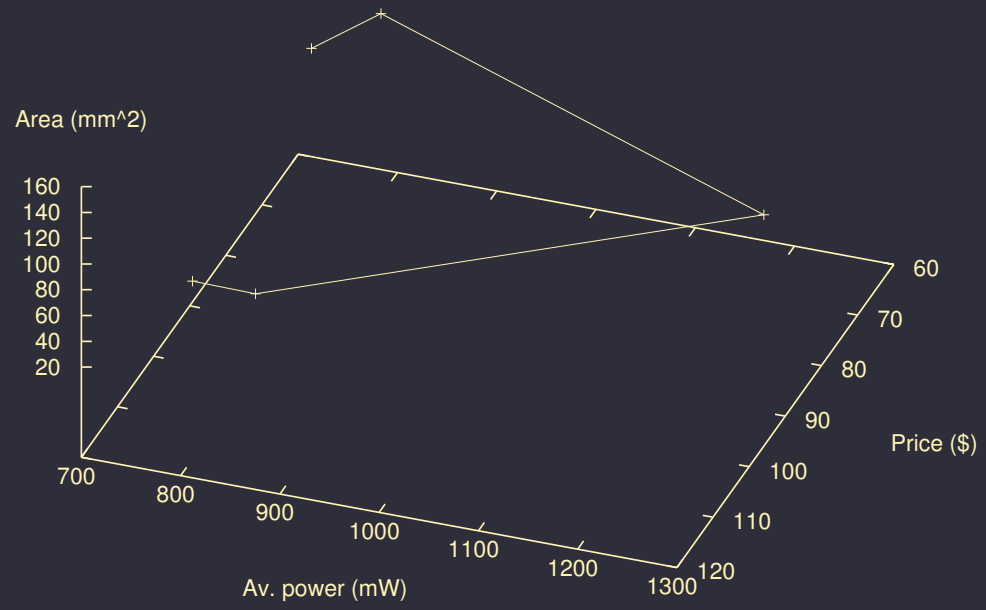
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



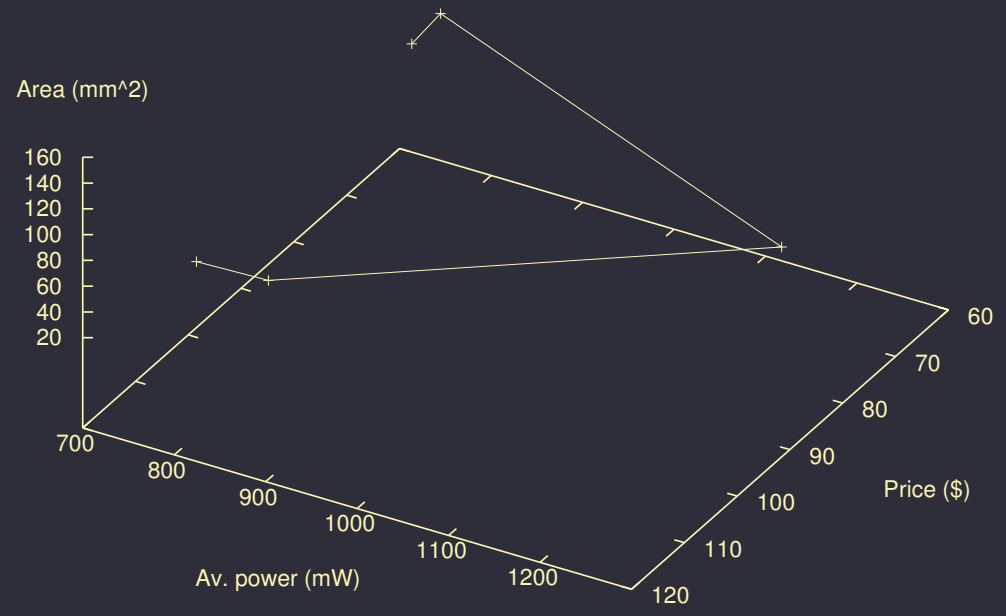
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



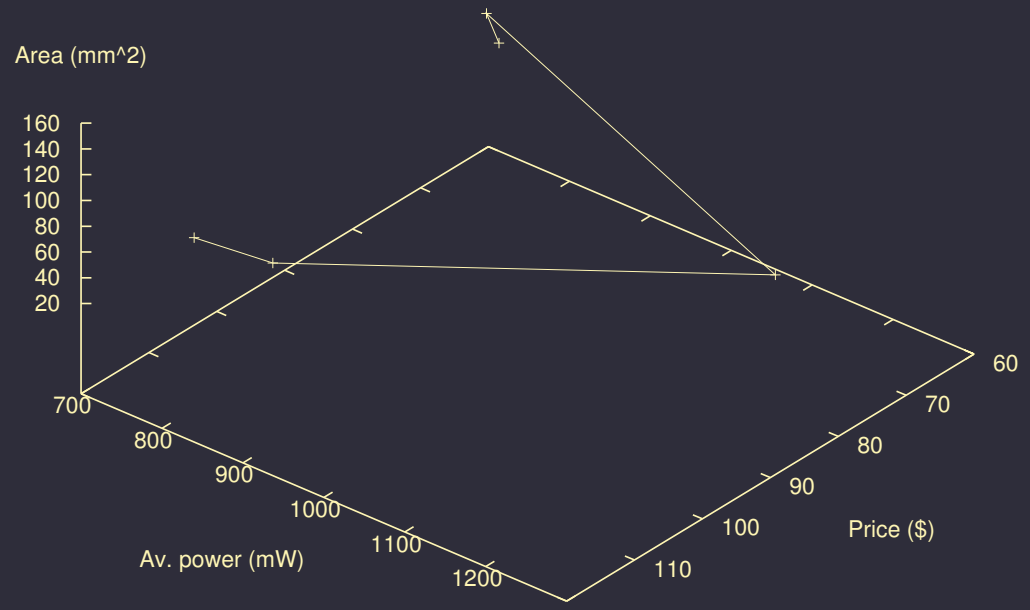
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



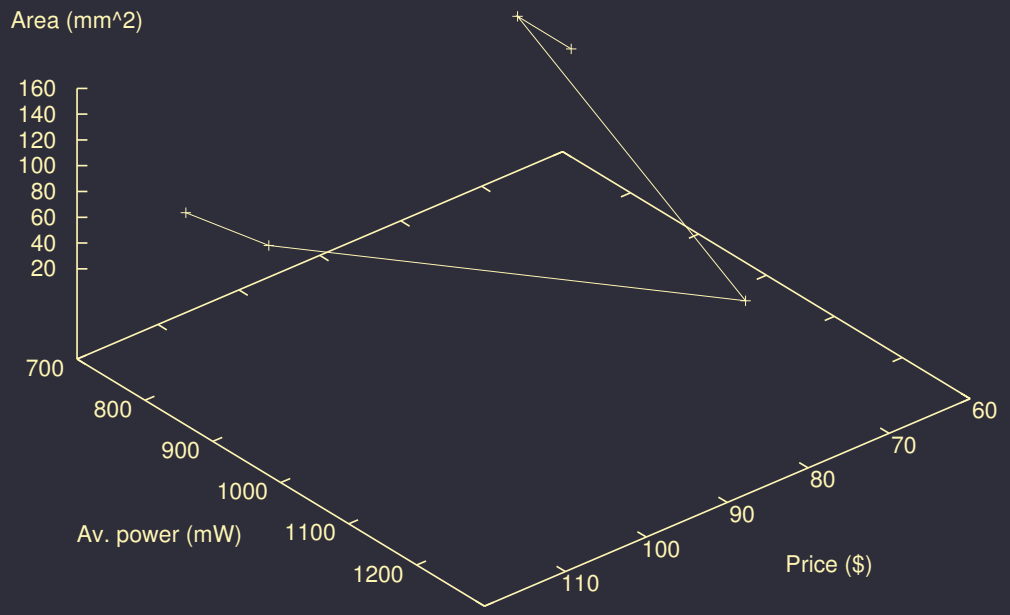
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



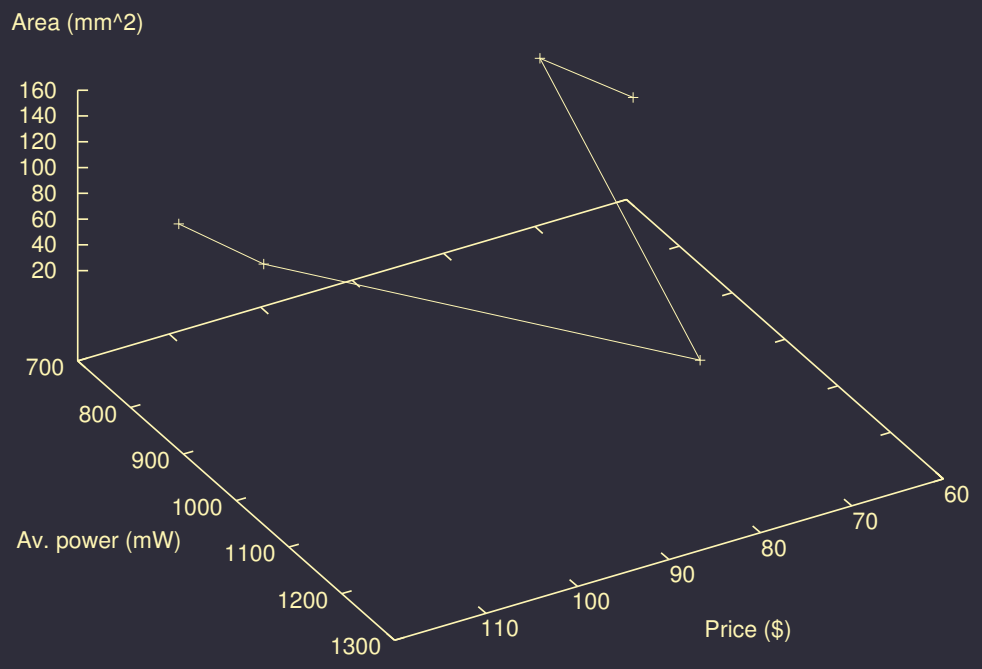
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



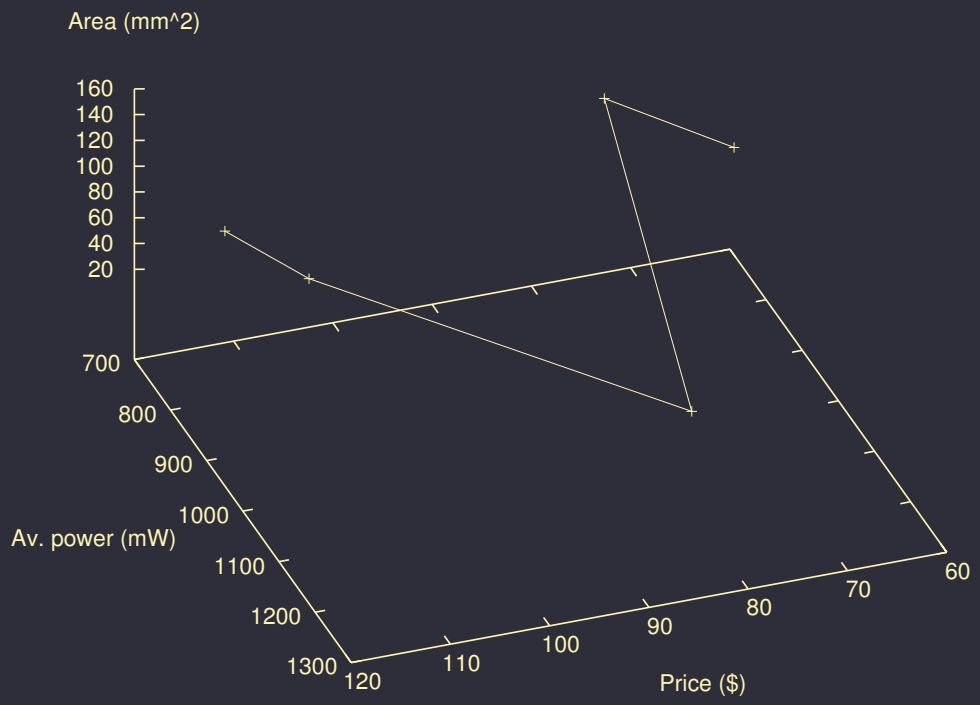
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



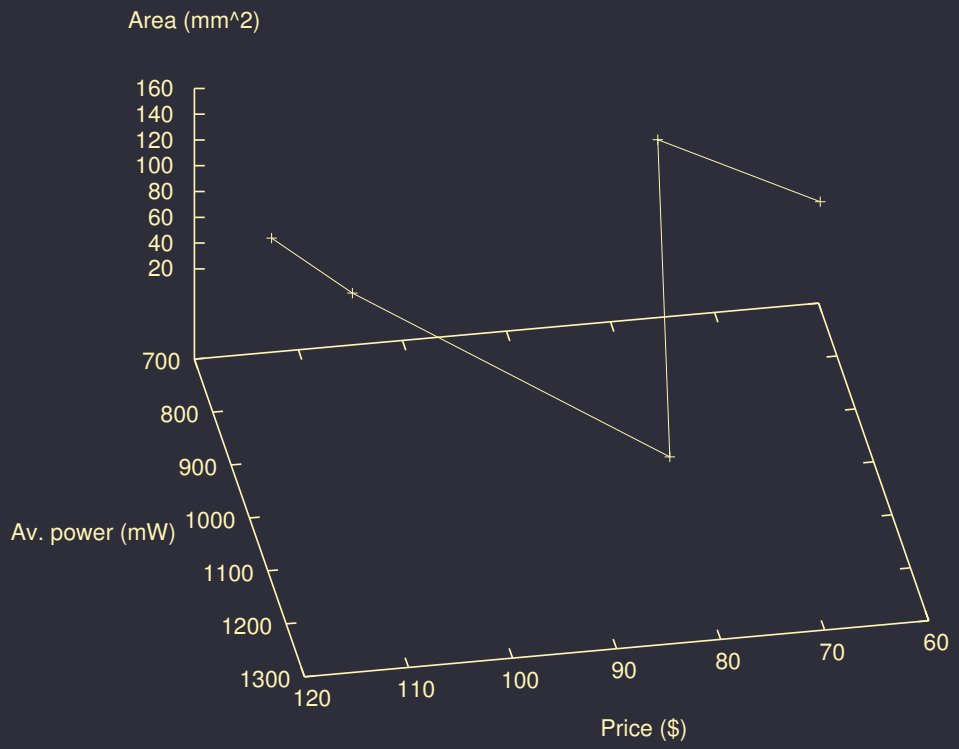
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



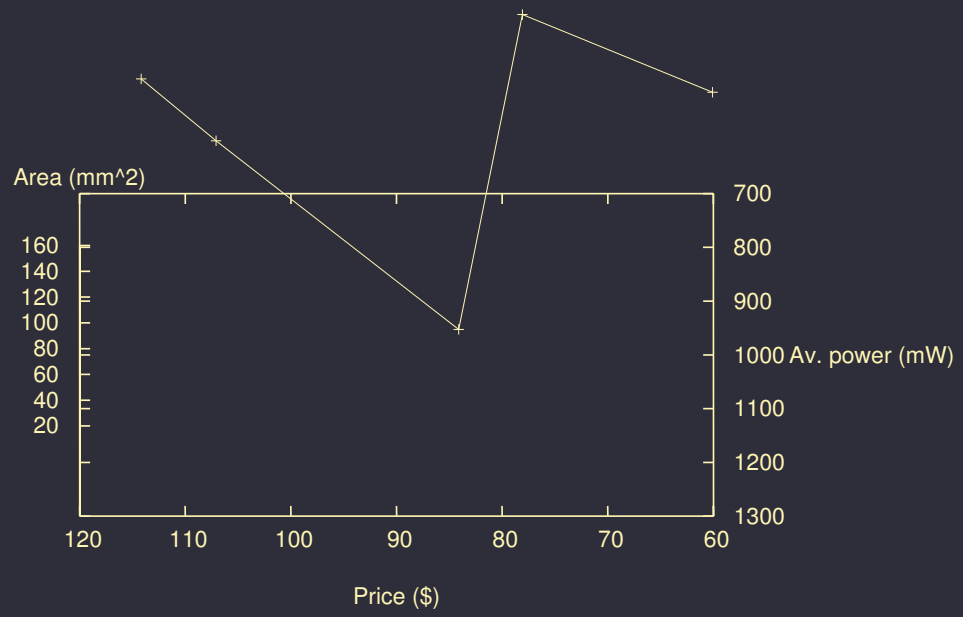
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



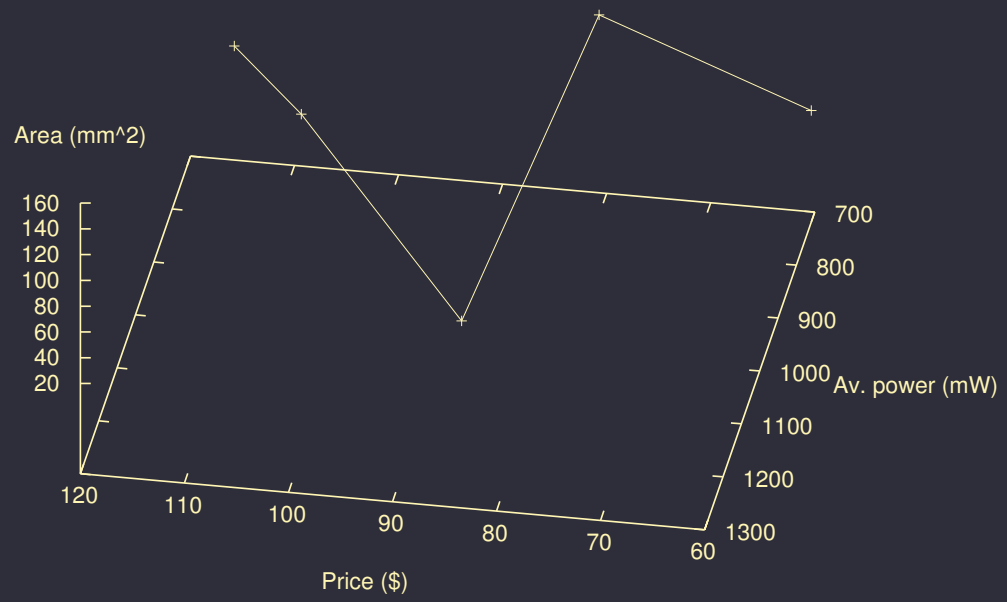
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



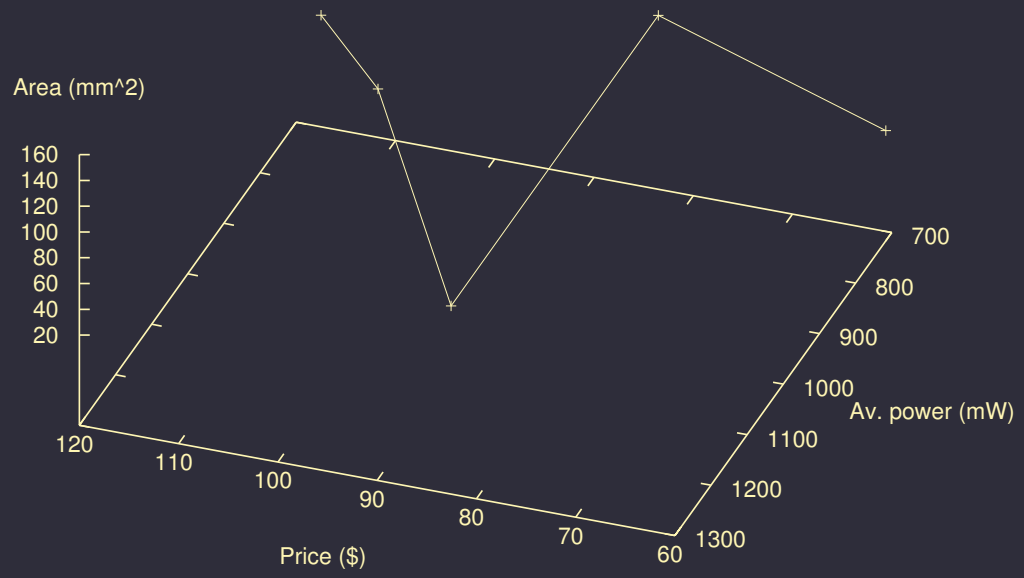
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



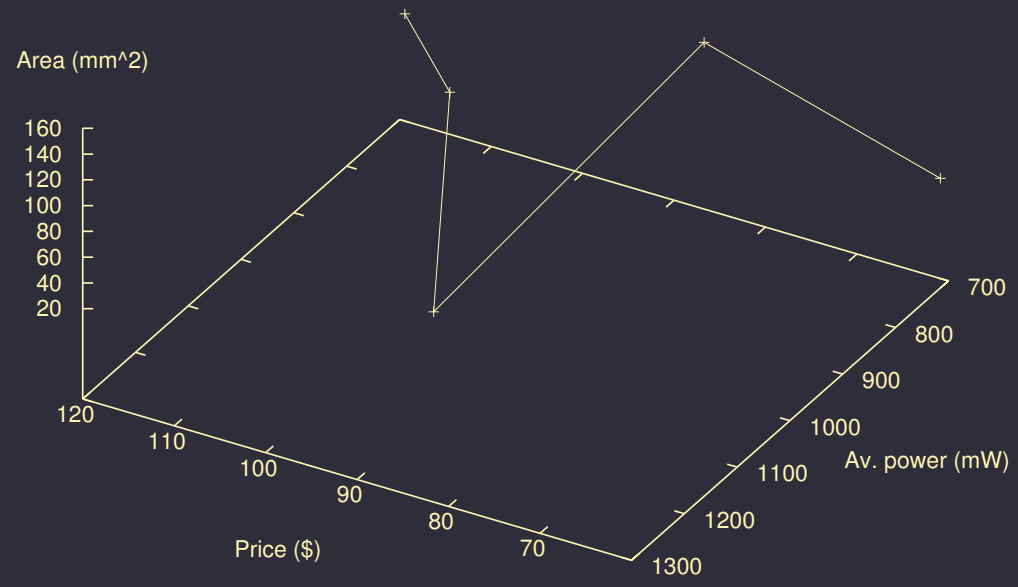
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



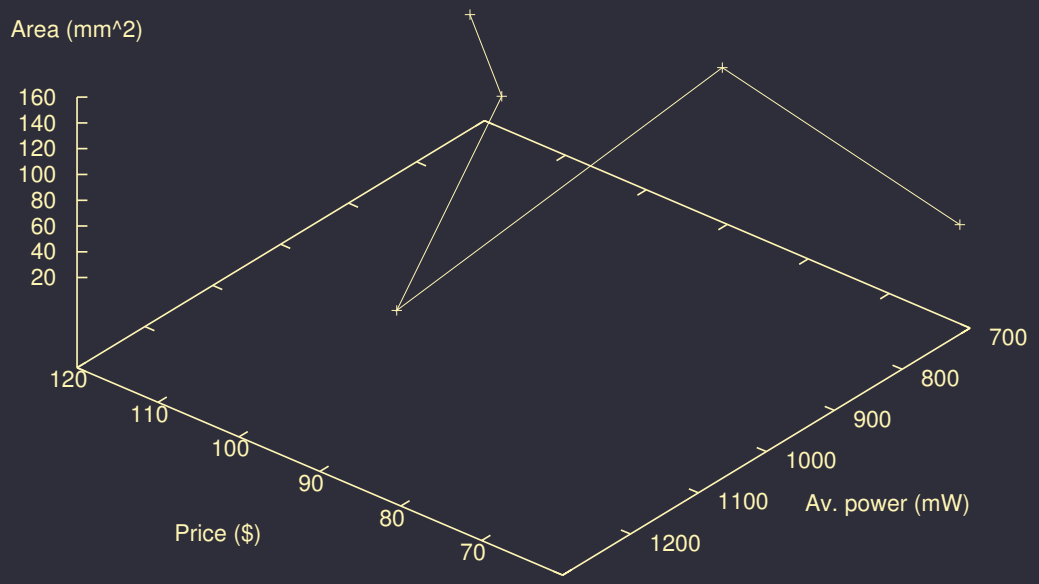
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



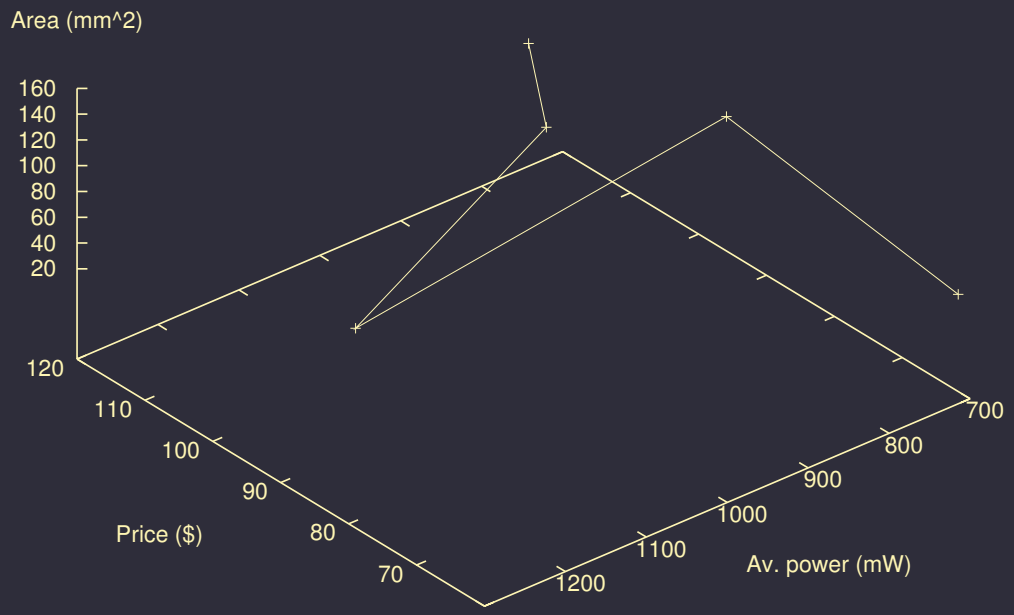
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



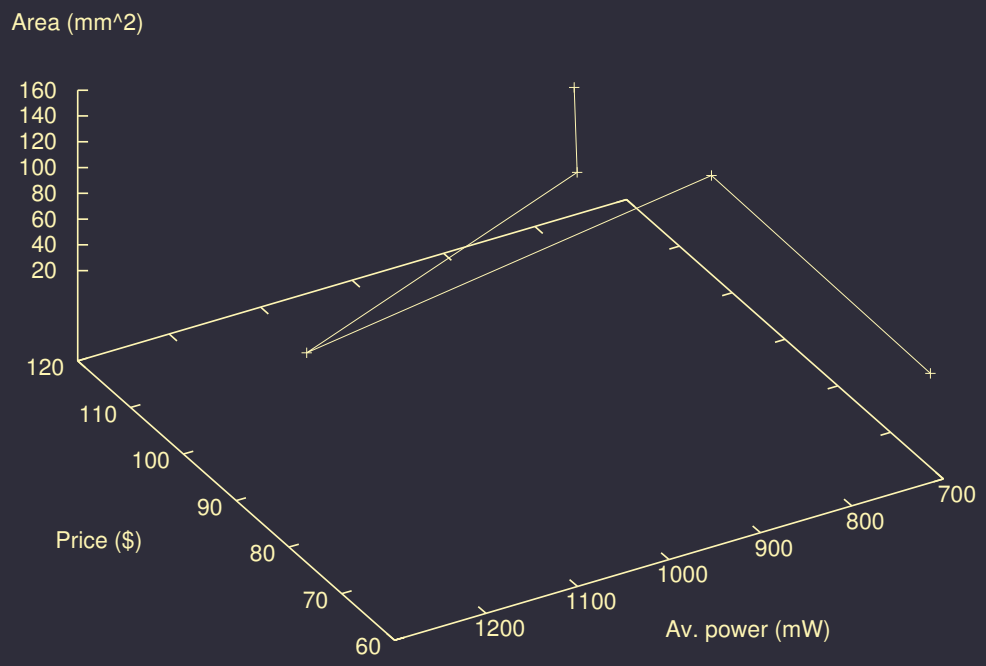
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



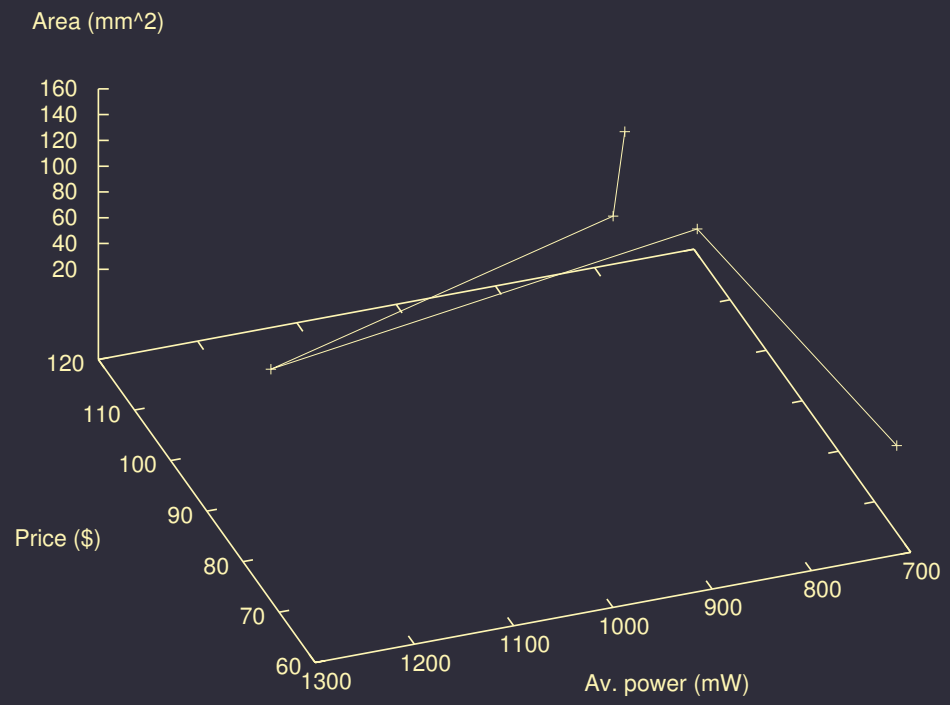
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



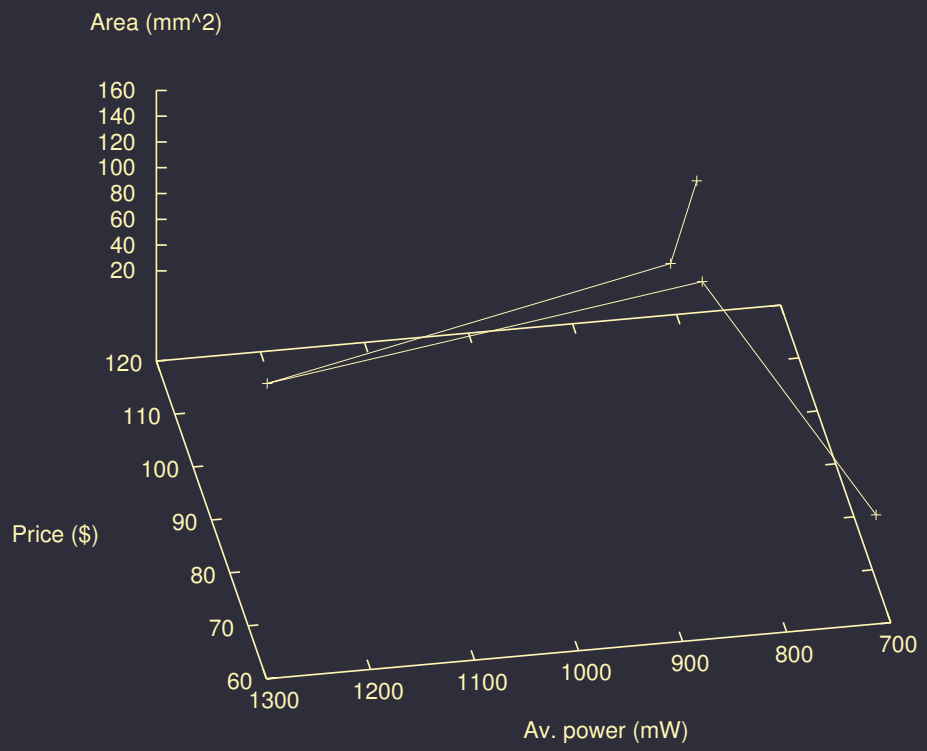
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



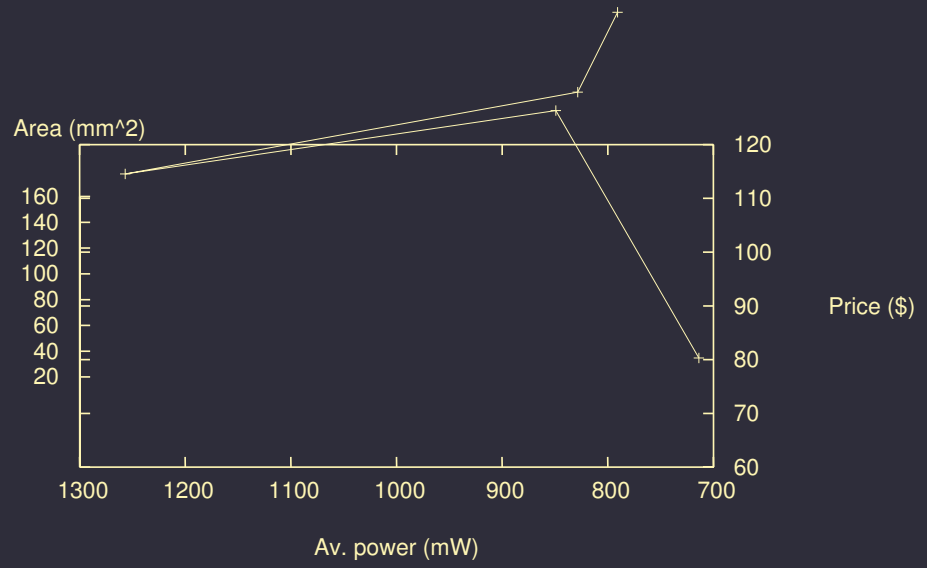
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



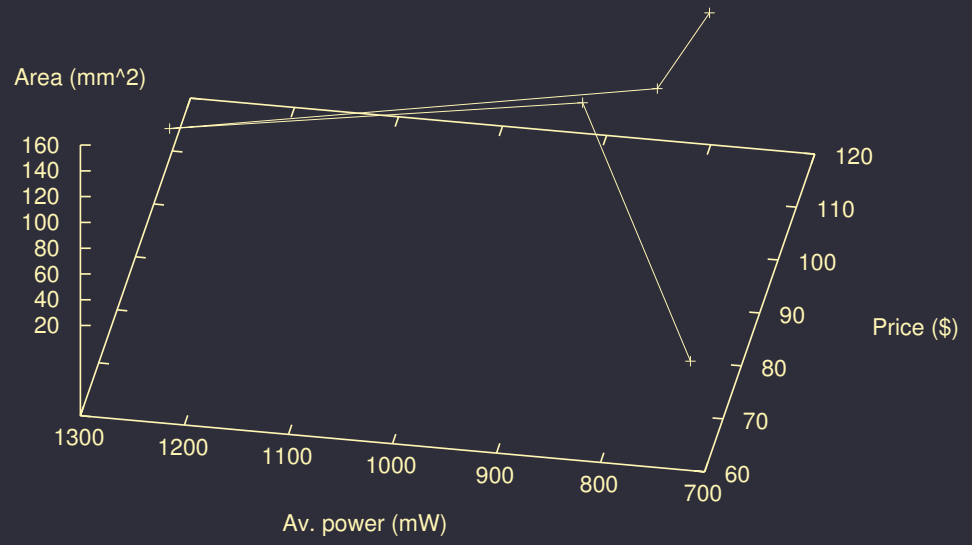
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



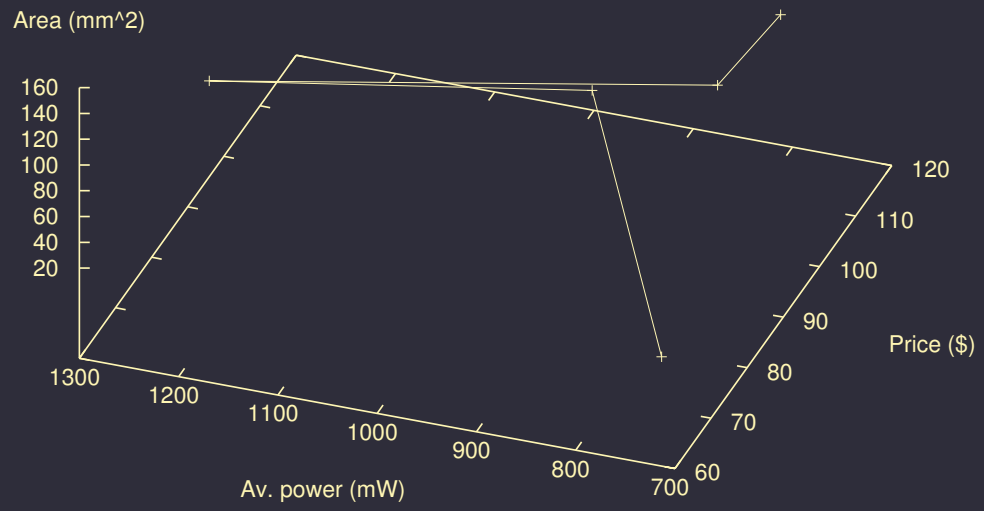
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



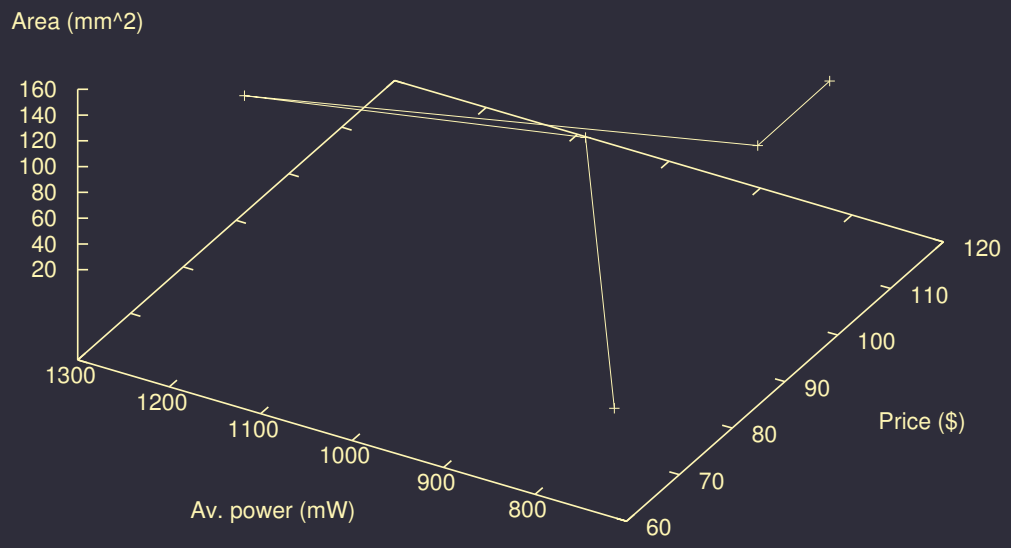
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



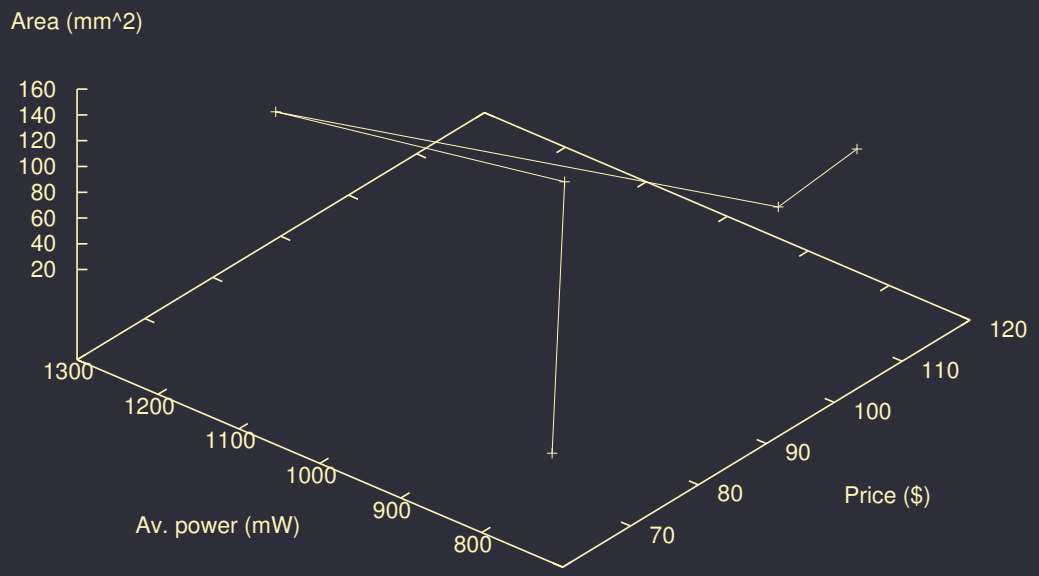
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



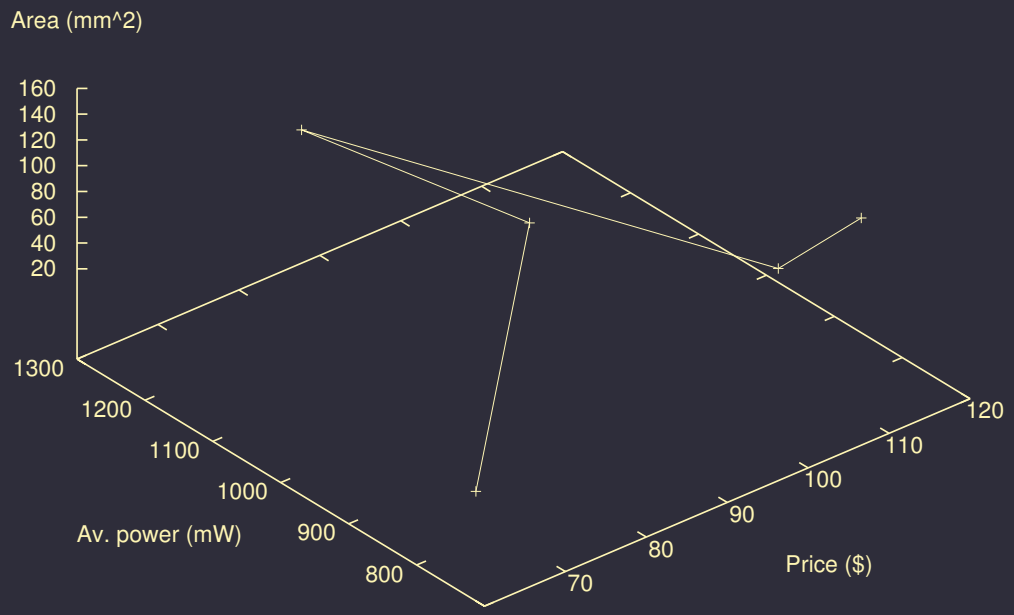
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



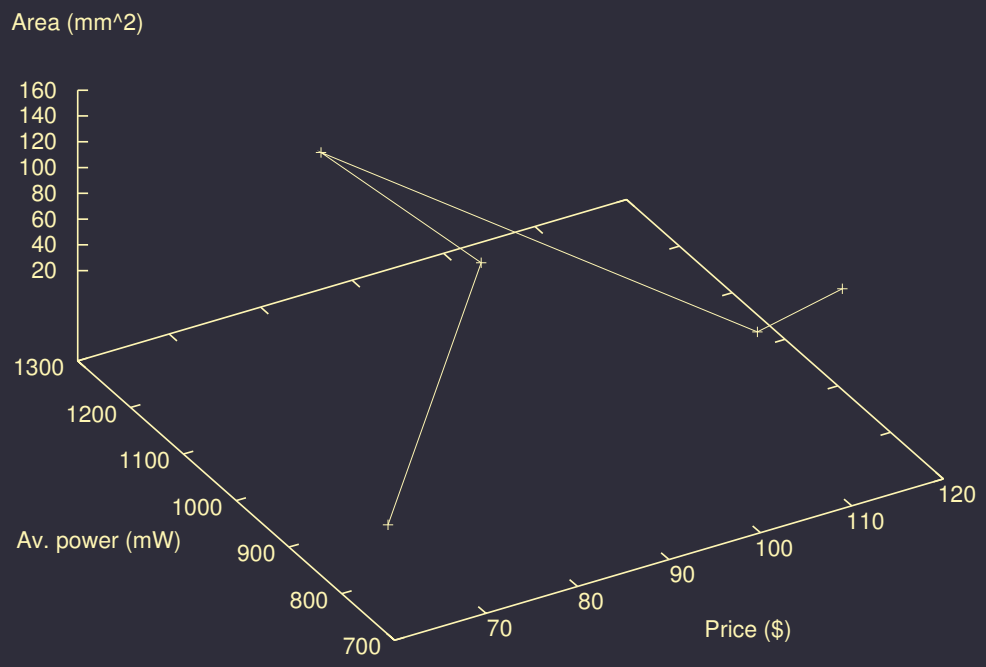
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



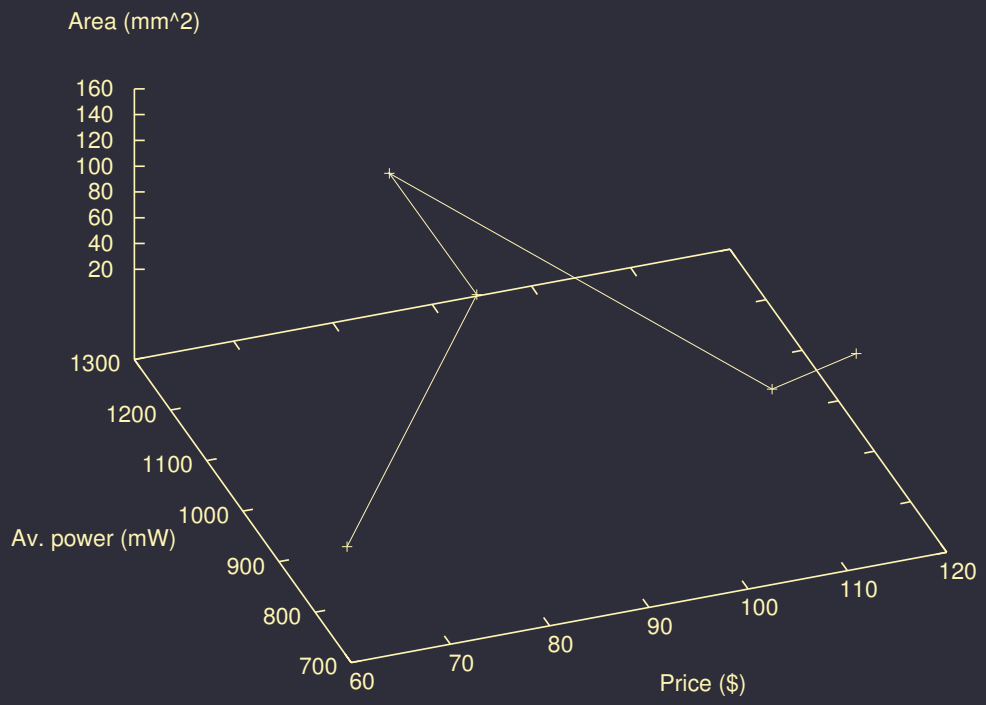
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



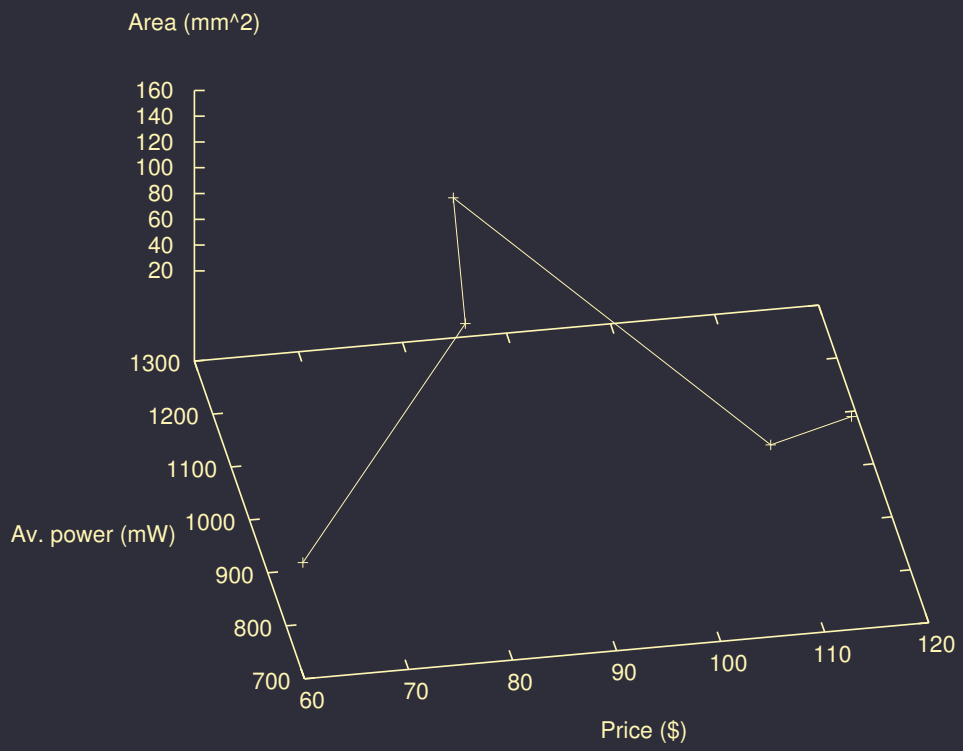
Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



Price, power, and area only. Soft deadline violation omitted.

MOCSYN Networking example



Price, power, and area only. Soft deadline violation omitted.

Problem complexity

Allocations:

$$max\_PE\_per\_type^{max\_PE\_types} \cdot max\_link\_per\_type^{max\_link\_types}$$

Assignments:

$$\mathcal{O}\left(PE\_count^{task\_count}\right)$$

Link Connectivities:

- Consider each PE to be a node in a graph
- Each link is a group which can contain up to $max\_contacts\_per\_link$ nodes

$$\mathcal{O}\left(C(PE\_count, max\_contacts\_per\_link)^{link\_count}\right)$$

Take a simple system:

$$\begin{aligned} max\_PE\_per\_type = max\_link\_per\_type &= 3 \\ max\_PE\_types = max\_link\_types &= 3 \\ PE\_count = link\_count &= 9 \\ task\_count &= 10 \\ max\_contacts\_per\_link &= 2 \end{aligned}$$

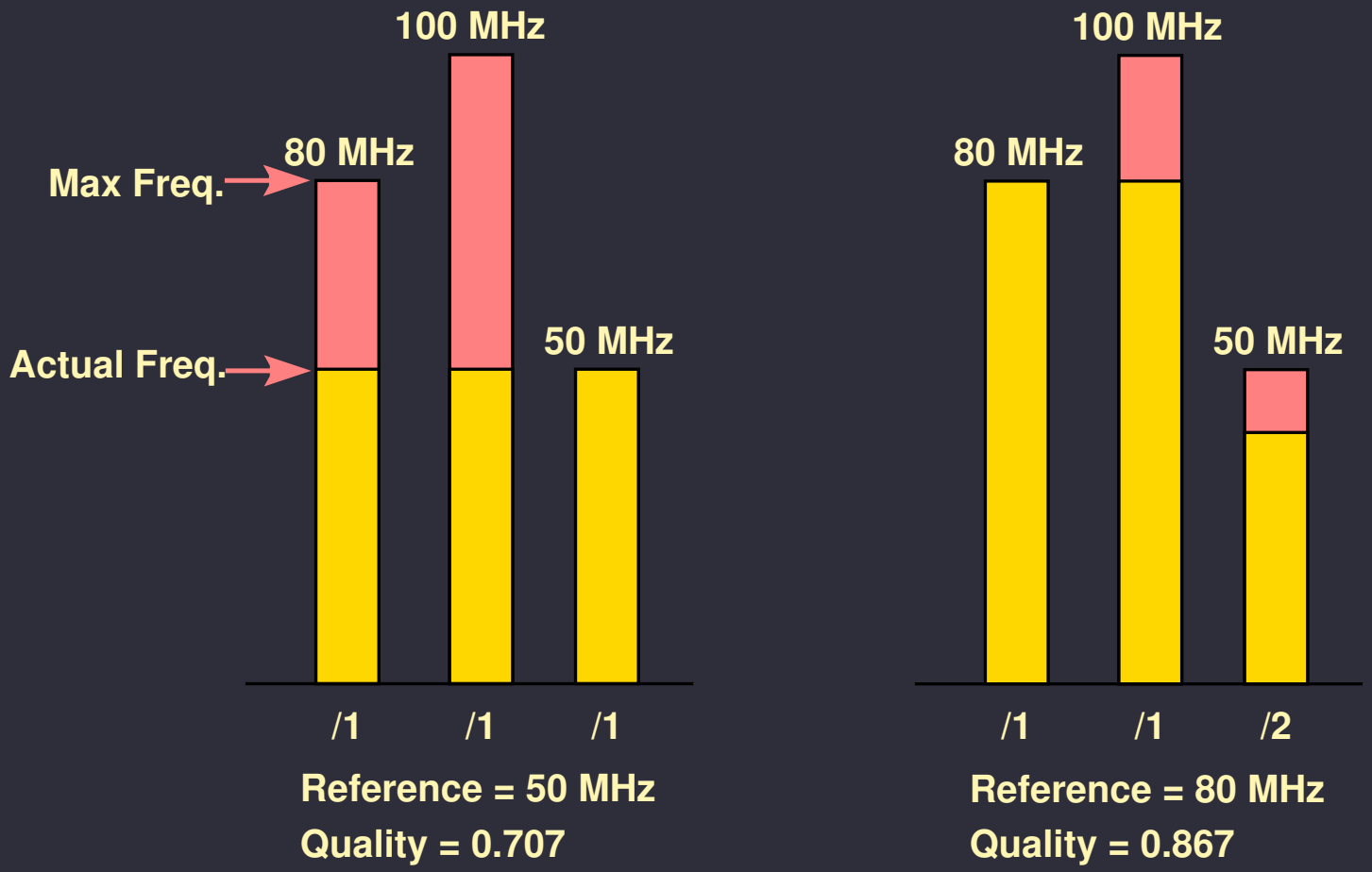
$allocations =$	$3^3 \cdot 3^3 = 27$	good
$assignments =$	$\mathcal{O}\left(9^{10}\right) = \mathcal{O}\left(3.49 \times 10^9\right)$	bad
$connectivities =$	$\mathcal{O}\left(C(9,2)^9\right) = \mathcal{O}\left(1.02 \times 10^{14}\right)$	worse

Number of architectures to evaluate:

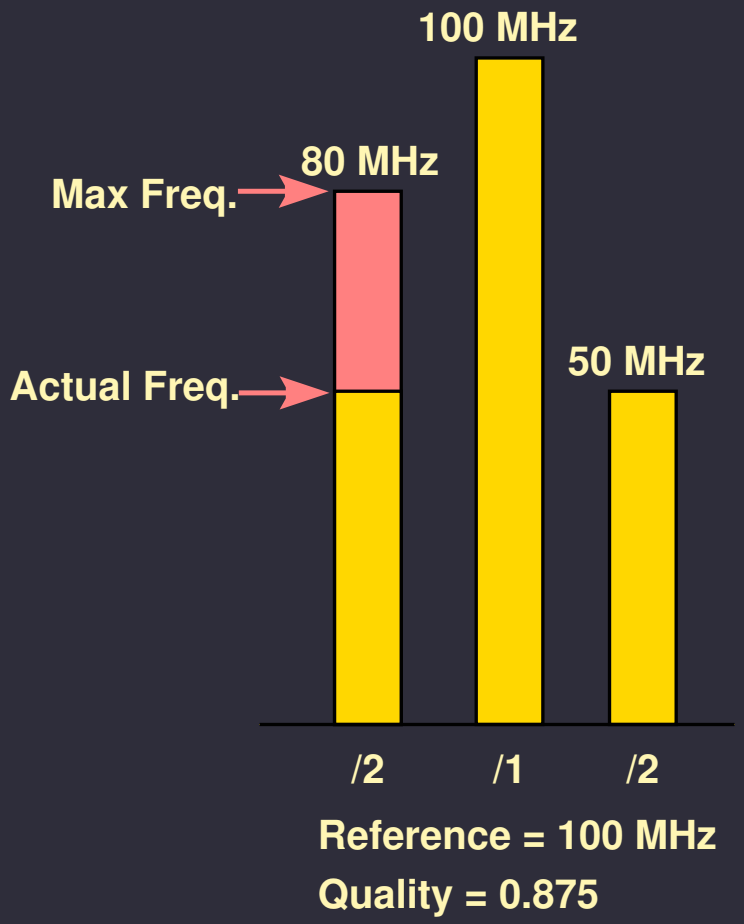
$$\mathcal{O}\left(27 \cdot 3.49 \times 10^9 \cdot 1.02 \times 10^{14}\right) = \mathcal{O}\left(9.57 \times 10^{24}\right)$$

. . . and this does not even take scheduling complexity or multi-core ICs into account

Counter-division only clock selection

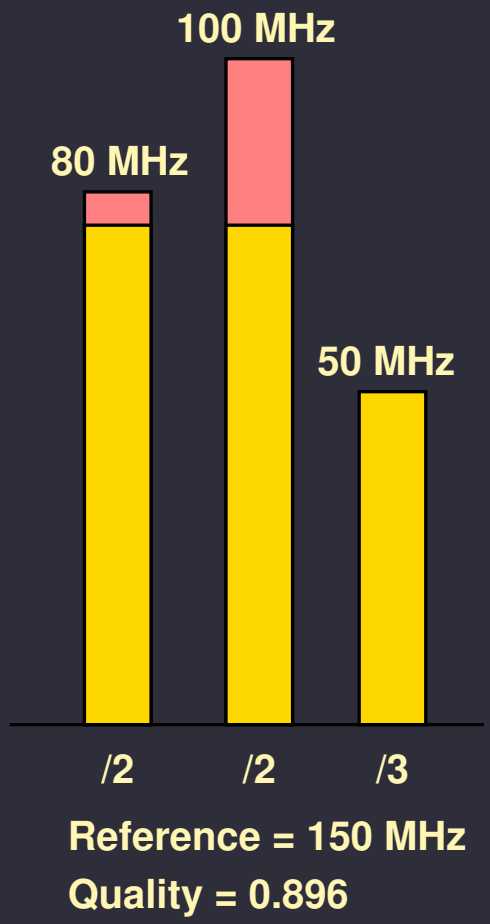


Counter-division only clock selection



Reference = 100 MHz

Quality = 0.875



Reference = 150 MHz

Quality = 0.896

Bus formation inner kernel

l is number of communicating core pairs

For each bus, i, intersecting with highest density point: $\mathcal{O}\left(l^2\right)$

  For each bus, j: $\mathcal{O}\left(l^3\right)$

    Tentatively merge i and j $\mathcal{O}\left(l^4\right)$

    Evaluate the density, *new_dens*, of *congest* $\mathcal{O}\left(l^3\right)$

    Evaluate new maximum contention estimate, *cont_est* $\mathcal{O}\left(l^4\right)$

If *new_dens* decreased for any tentative merge:

  Merge the pair with greatest *new_dens* decrease $\mathcal{O}\left(l^2\right)$

  Break ties by selecting merge with least *cont_est* increase.