Diffusion-Driven Congestion Reduction for Substrate Topological Routing

Shenghua Liu [1], Guoqiang Chen [2], Tom Tong Jing [3]

Lei He [3], Robby Dutta [2], Xian-Long Hong [1]

[1] Tsinghua University, Beijing, 100084, China

[2] Magma Design Automation, Inc., San Jose, CA 95110, USA

[3] UCLA, Los Angeles, CA, 90095, USA

Speaker: Shenghua Liu

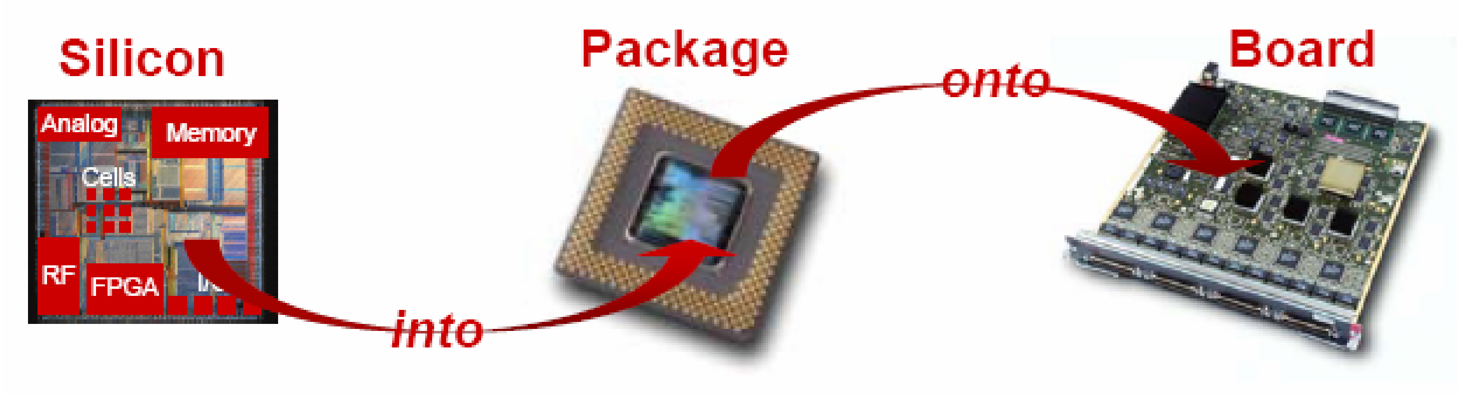
Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Package



- Package substrate
  - PGA (pin grid array)
  - BGA (ball grid array)
- Two techniques to mount the die to the substrate
  - wire bonding, WB
  - flip chip, FC

Substrate Routing

- Packaging in BGA with wire-bonding technique
  - chip is put into the cavity of substrate
  - chip I/Os are connected to bonding pads around the cavity
  - substrate routing connects bonding pads with balls
- Packaging in BGA with flip-chip technique
  - re-distribution layer, RDL, routing connects chip I/Os to bump array
    - [J. W.Fang et al., DAC, 2007] [J. W.Fang et al., ICCAD, 2005]
  - escape routing breaks bumps out to substrate routing layer
  - break points lay on the escape boundary
  - substrate routing connects break points to balls

Examples

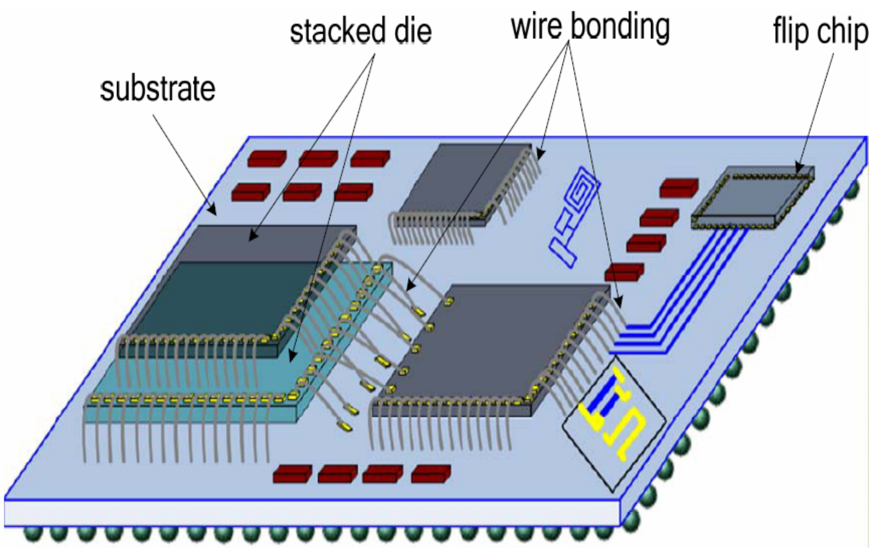
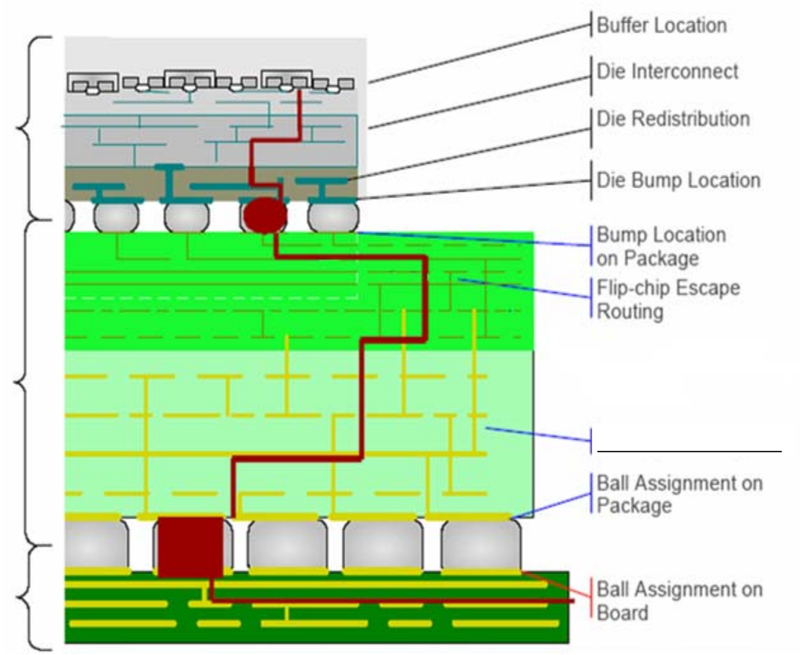


Fig. An example of IC package.



An Example [Cadence]

- BGA + flip-chip
- Substrate routing

Substrate Topological Routing

- Substrate routing usually has two steps: topological routing and detailed routing
- [Chen and Lee, TCAD 1996] [W. W. Dai et al., DAC 1991] discussed detailed routing
- This paper studies topological routing
- Substrate routing is preferred to be planar, even though multiple routing layers are available [Xiong et al., ASPDAC 2006]

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Existing Work (1)

- A very recent substrate topological routing algorithm [Liu et al., DAC 2008] [Liu et al., TCAD 2009]
  - had the best reported routability in the literature
  - is used in a state of the art commercial tool
  - proposes “dynamic pushing” to tackle the routing order problem
  - proposes “flexible via staggering” to improve the routability
  - resulted in 3.5% net unrouted for nine industrial designs
- However, the congestion reduction method of iteratively avoiding routing through congested area, limited its advantage in routability

Existing Work (2)

- The earlier substrate routing Surf system [Staepelaere et al. 1993]
  - applied topological routing to generate rubber-band sketch [Dai et al., DAC 1991]
  - transformed sketch first to spoke sketch and then to precise geometrical layout
  - Surf assumed a fixed end point
  - Surf completed topological routing with a global routing stage followed by a local routing.
- Our formulation uses end-zone
  - more flexible and therefore increases routability.
- Our router (named D-Router)
  - uses iterative congestion reduction by diffusion without partitioning
  - avoids the problem of fixing congestion only within each bin

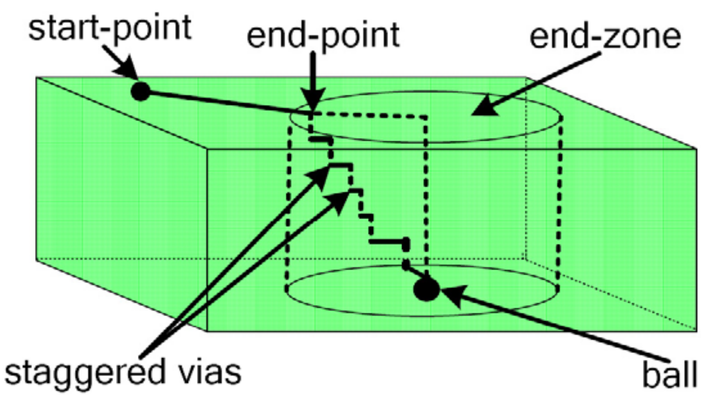
Existing Work (3)

- A recent on-chip router, BoxRouter [M. Cho and D. Pan, DAC 2006] achieves good routability
  - all nets within a congested window are ripped-up as a whole
  - all nets rerouted simultaneously by an integer linear programming (ILP) method.
  - the ILP method assumes Man-hattan routing, and extension to non-Manhattan substrate routing is unclear.
- D-Router
  - essentially rips-up and reroutes wire segments net-by-net, and not necessarily reroutes all nets inside a window.
  - iterates window by window while BoxRouter expands the window
  - can solve non-Manhattan substrate routing

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Staggered via and end-zone



- When dropping signal vias
  - close to the positions above assigned destination ball
  - vias need to be staggered
  - required offsets between staggered vias
- End-zone
  - center *oz* is aligned with the ball
  - radius $R = \sum_i pd_i$ where pd_i is the maximal staggered via pitch in the layer with index i

Problem formulation

- Given
  - start-points,
  - end-zones (associated with assigned balls in the bottom layer),
  - netlist (definition of connections between start-points and end-zones),
  - and obstacles (including the escape area for escape routing, the pre-routed connections, vias, and other obstacles in the layer),
- Find
  - a topological routing solution
- Such that
  - routed nets have no intersections
  - satisfy the capacity constraints
  - and have minimal length

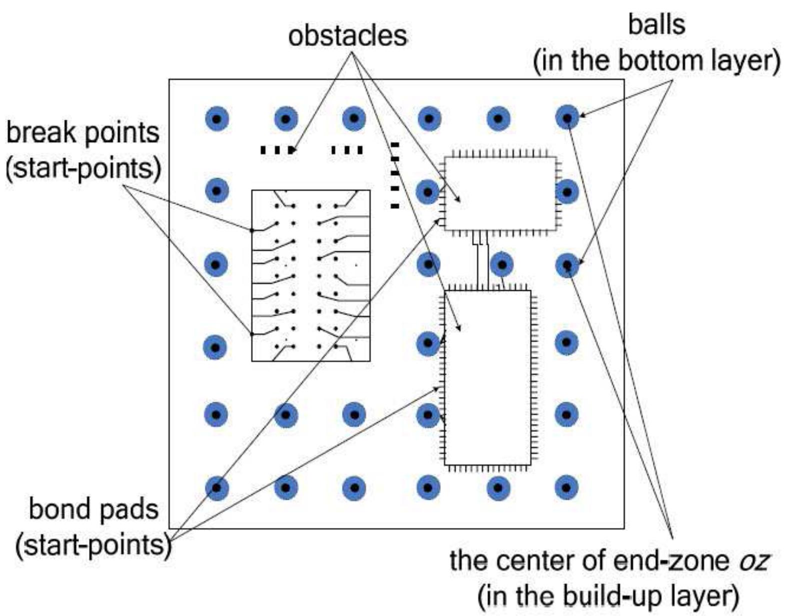
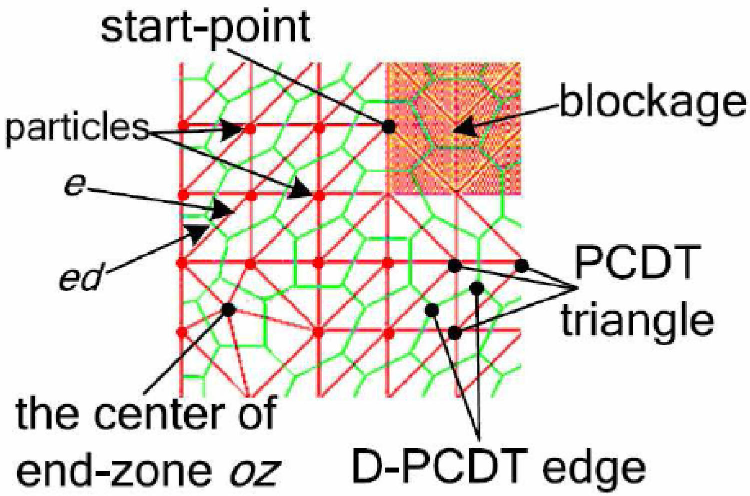


Fig. Substrate routing graph (SRG) in a signal layer

Data Structure

- The substrate routing plane (SRG) is triangle-meshed by constraint Delaunay triangulation (CDT)
- Uniformly spreading points are added for particle-insertion-based CDT



- Capacity $C_e = l_e$ is the length of edge *e*
- Congestion $\eta_{ed} = \frac{\sum_i (w_i + s_i)}{C_{ed}}$
  - where w_i and s_i are the wire segment/end-point (*i.e.* via) width and space of net *i* that passes through edge *e*, respectively.

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Baseline Algorithms (1)

- [Liu et al., DAC, 2008] is a very recent published substrate topological routing
  - Same problem formulation with D-router
  - It routes net by net based on A* algorithm with dynamic pushing and flexible via-staggering.
  - It also applies post-routing rip-up-and-reroute iteration for congestion reduction.
  - It claimed that good routing topology could be achieved at the beginning for routing convergence.
- D-router chooses its first routing iteration as an initial routing
  - Congestion is not considered firstly

Baseline Algorithms (2)

- Negotiation-based substrate routing is also compared
  - Negotiation-based algorithm has obtained high-quality solutions to on-chip routing of FPGA [McMurchie and Ebeling 1995] and ASIC [Roy and Markov 2007] [Cho et al. 2007]
  - Negotiation-based cost function was implemented based on the work [Roy and Markov 2007]

$$NC_e^{'} = (rc + h_e) \times p_e + ec$$

where *rc* and *ec* are the realized and estimated costs, p_e reflects the present congestion, and h_e represents the congestion history. h_e is given by

$$h_e^{k+1} = \begin{cases} h_e^k + h_{inc}, & \text{if } e \text{ has overflow} \\ h_e^k, & \text{otherwise} \end{cases}$$

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

D-Router Scheme

- The scheme of D-Router
  - starts with any initial routing solution
    - One iteration of routing in [Liu et al., DAC, 2008] is used
  - then finds out each highly congested area
  - spreads out net wires to its neighbors for congestion reduction

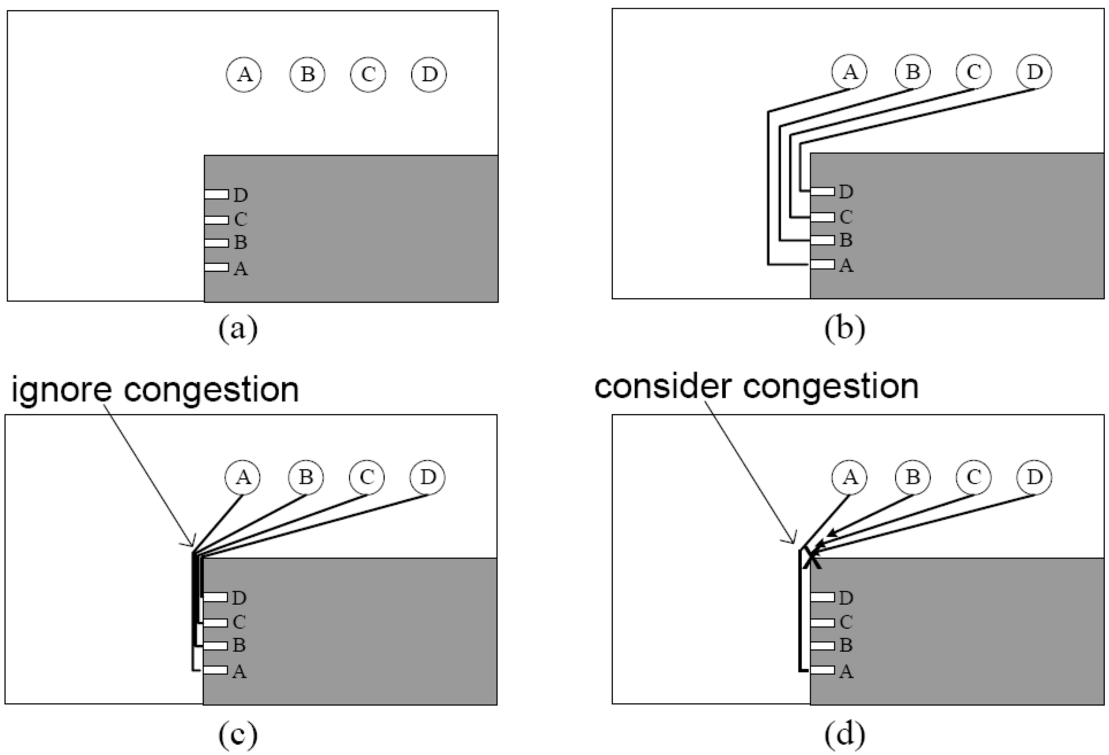
Motivating Examples

- The example in Fig (a) illustrates why D-Router is free of the routing order

  ✓ routing order D-C-B-A generates solution (b)

  ✓Routing order A-(BCD) get solution (c) firstly, but (d) in the later iteration by A* and Maze based router

  ✓D-router spreads congested nets in (c), and achieves (b)



A Routing Puzzle

- The routing order problem can become harder even in a two-net case.
- Figure bellow gives a routing puzzle for the algorithm [Liu et al., DAC, 2008]

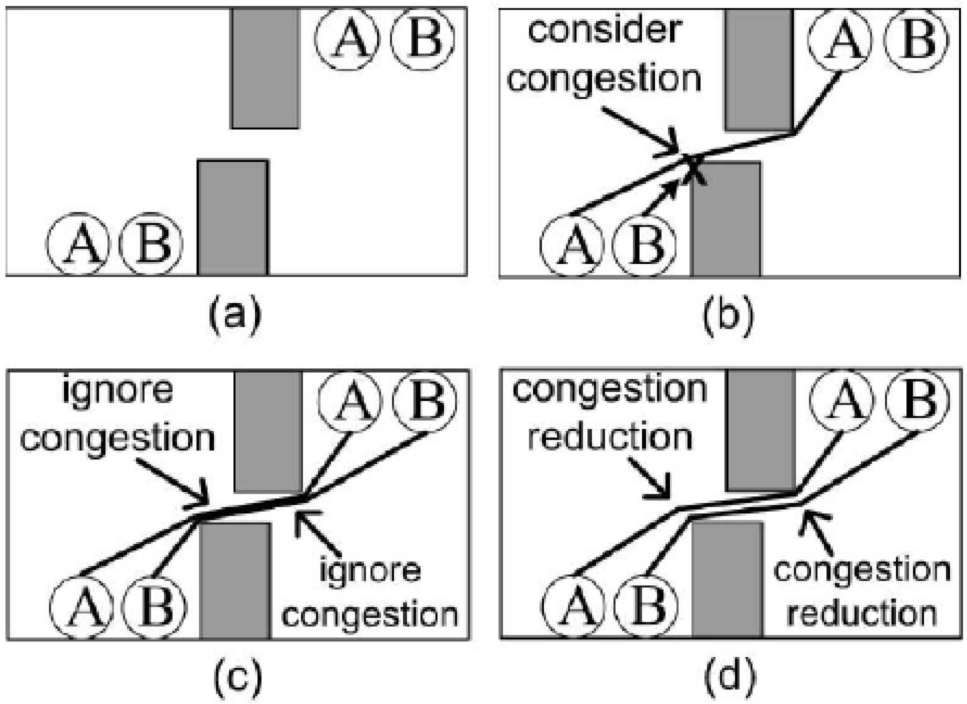


Fig. An example without a valid net ordering.

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Diffusion

- Congestion reduction in D-Router simulates the process of dopant diffusion
- Each triangle edge is an atomic location unit for net movement.
- The atomic diffusion is to move one net segment or end-point to adjacent triangle edges
- D-Router is based on an localized and non-analytical diffusion model

Diffusion window

- Define the *concentration* $d_e(t)$ of PCDT edge *e* as congestion on edge *e* for moment *t*

$$d_e(t) = \eta_e(t)$$

- *Diffusion window* is an isolated area for congestion reduction
  - a highly congested PCDT edge *e* as a *diffusion source*
  - diffusion window includes edge *e* itself and adjacent edges sets E_1 and E_2

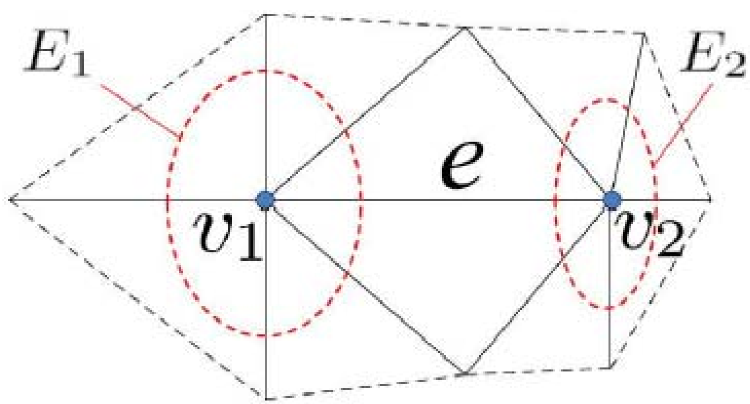


Fig. A diffusion window for edge *e*.

Diffusion velocity and direction

- Diffusion concentration inside window
  - when a net moves towards set E_1 or E_2, it may pass through more than one edge
  - *diffused edges Edf* is defined as the edges in *E1* or *E2* through which the net passes
  - the concentration value of *Edf*

$$d_{Edf}(t) = \max_{e_i \in Edf}\{\eta_{e_i}(t)\}$$

- Diffusion direction
  - Diffusion velocity

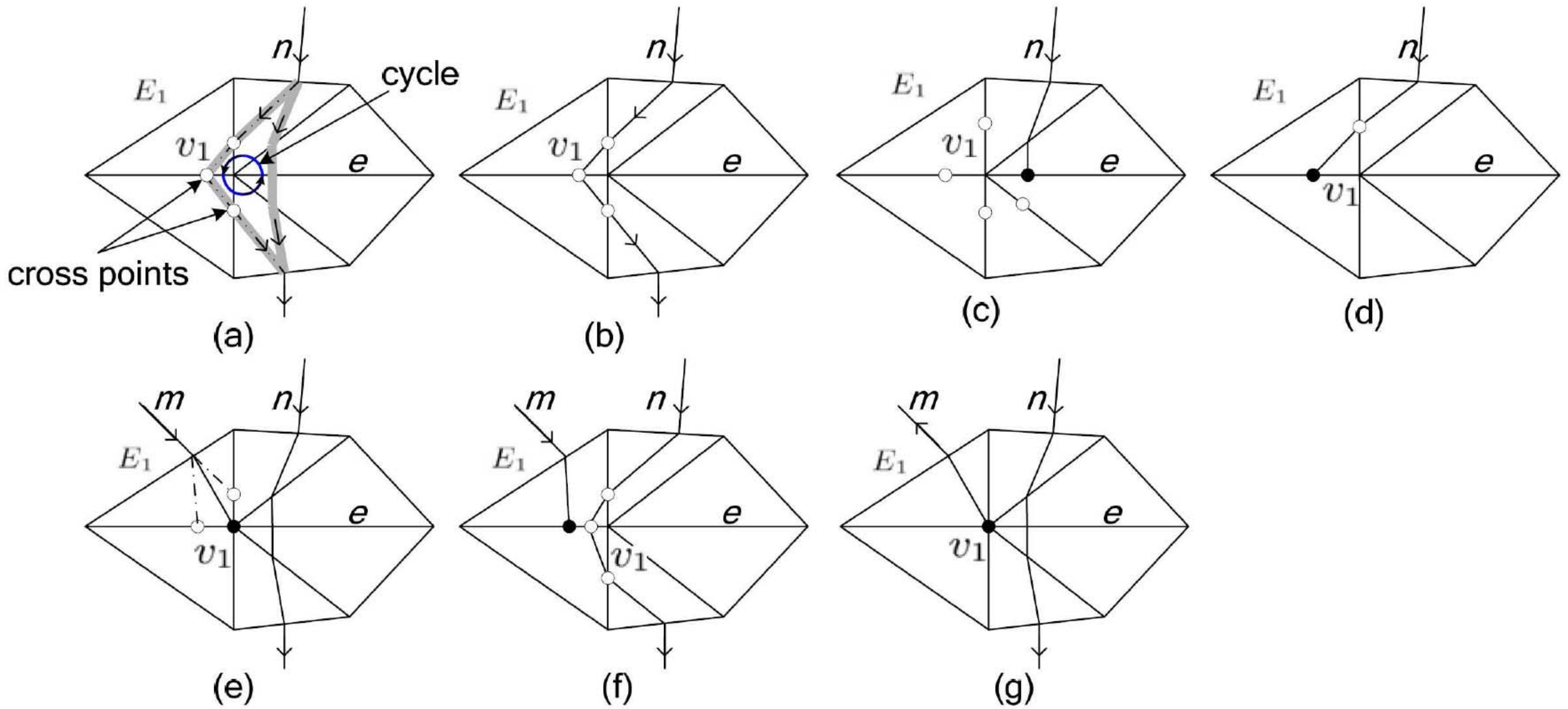
$$v_{e+}(t) = -(d_{Edf+}(t) - d_e(t))/d_e(t)$$
$$v_{e-}(t) = -(d_{Edf-}(t) - d_e(t))/d_e(t)$$

  where Edf^+ and Edf^- are the diffused edges in *E1* and *E2*

  - we select the direction with higher speed to perform the diffusion at each moment
  - It means towards low concentration and low congestion.

Momentary-diffusion operations

- Each operation is an atomic net segment/end-point movement from a diffusion source to selected diffusion direction.



Case 1: Normal, Fig (a)-(b);
Case 2: Net *n* stops at a diffusion source *e,* Fig (c)-(d)
Case 3: Net *m* stops at vertex *v*1 beside net *n,* Fig (e)-(f)
Case 4: Net *m* starts from vertex *v*1 beside net *n,* Fig (g)

Diffusion equilibrium and convergence

- Condition I: If the congestion constraint is satisfied on edge *e*, diffusion reaches equilibrium.
- Condition II: Diffusion reaches equilibrium when next momentary-diffusion is over diffusion
  - *Over diffusion* is a momentary-diffusion that makes the diffusion source less congested than diffused edges, *Edf*.
- Condition III: Both diffusion directions are blocked or forbidden
- A heap *H* and a taboo list *Tb* are maintained for the process of diffusion
  - *H* maintains all possible diffusion sources and is heapified by edge congestion
  - *Tb* maintains all the edges that are no longer allowed to diffuse congestion
  - When a diffusion source in *H* reaches equilibrium due to Condition II, it is added into *Tb* until any neighbor edge reduces congestion.

Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Test cases

- Table 1 summarizes the test case characteristics
  - package type and size, die size, and
  - total number of nets 6415.

Table 1: TEST CASE CHARACTERS
(*: Package size and Die(s) size are given by width × length (μm) in rectangle.)

Test case ID	Package type	Package size*	Die(s) size*	Number of nets
Q1	2-0-2	10000×10000	75007×700	315
B2	2-2-2	35000×35000	14000×15000	474
F3	2-2-2	30000×30000	9000×10500	543
P4	3-1-3	40000×40000	9300×9300	800
A5	3-2-3	35000×35000	12000×12000	506
A6	3-2-3	40000×40000	20000×22000	891
X7	4-2-4	40000×40000	20000×23000	990
A8	4-2-4	45000×45000	20000×19000	1009
S9	1-0-1	12000×12000	3900×6700 4400×5700 3200×4400	349
S10	2-2-2	37500×37500	11000×10000 4700×3800 4600×5500	538
total	—	—	—	6415

The last nine test cases

- The last nine test cases are from [Liu et al., DAC 2008]
- However, designers practically prefer some I/Os to connect to solder balls in specified regions for the sake of PCB design
- In our experiments, the solder balls are reassigned with such region constraint
- The netlist is changed, which becomes harder to solve by [Liu et al., DAC 2008]
- Thus, new names are given to the nine test cases in order to distinguish from those in [Liu et al., DAC 2008]

Comparison results

- Two alternative algorithms for comparison
  - [7] is the recently published substrate topological routing algorithm [Liu et al., DAC, 2008]
  - Nego is negotiation-based substrate routing introduced in the baseline algorithms

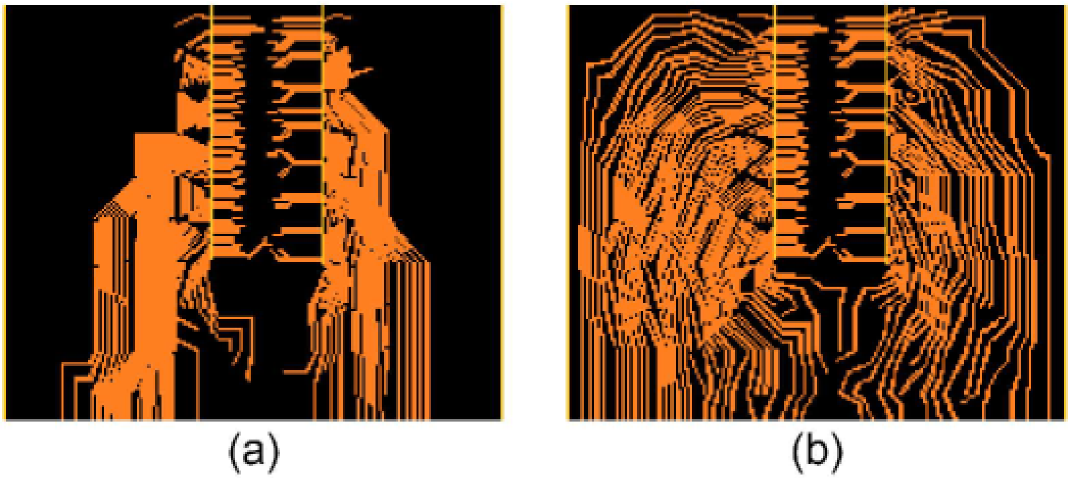
Table 2: EXPERIMENTAL RESULTS
(Nego: negotiation-based substrate routing)

Test case	Number of failed nets			Wire length (mm)			Runtime (s)		
	[7]	Nego	D-Router	[7]	Nego	D-Router	[7]	Nego	D-Router
Q1	51	41	26	1.64	1.69	1.70	5.34	9.79	7.17
B2	31	30	0	6.98	6.98	7.17	11.39	17.84	9.00
F3	24	22	0	7.79	7.80	7.96	14.36	14.41	16.91
P4	135	135	48	11.90	11.90	12.30	41.64	20.04	13.87
A5	64	63	7	14.90	14.90	16.30	15.27	17.65	10.92
A6	60	57	15	4.98	4.99	4.93	12.12	18.77	12.87
X7	45	45	8	6.55	6.54	6.53	39.51	45.11	25.38
A8	16	16	0	18.50	18.50	18.70	44.55	47.24	9.32
S9	22	20	0	1.67	1.67	1.65	2.11	3.2	0.96
S10	32	32	0	9.53	9.53	7.90	284.17	286.34	3.01
total	480	461	104 (1/4.6x)(1/4.4x)	—	—	—	—	—	—
average	—	—	—	8.45	8.46	8.51	46.05	47.04	10.94 (1/4.2x)(1/4.3x)

D-Router reduces the number of unrouted nets to 104, a 4.6x net number reduction,

also reduces runtime by an average 4.3x

Routing results

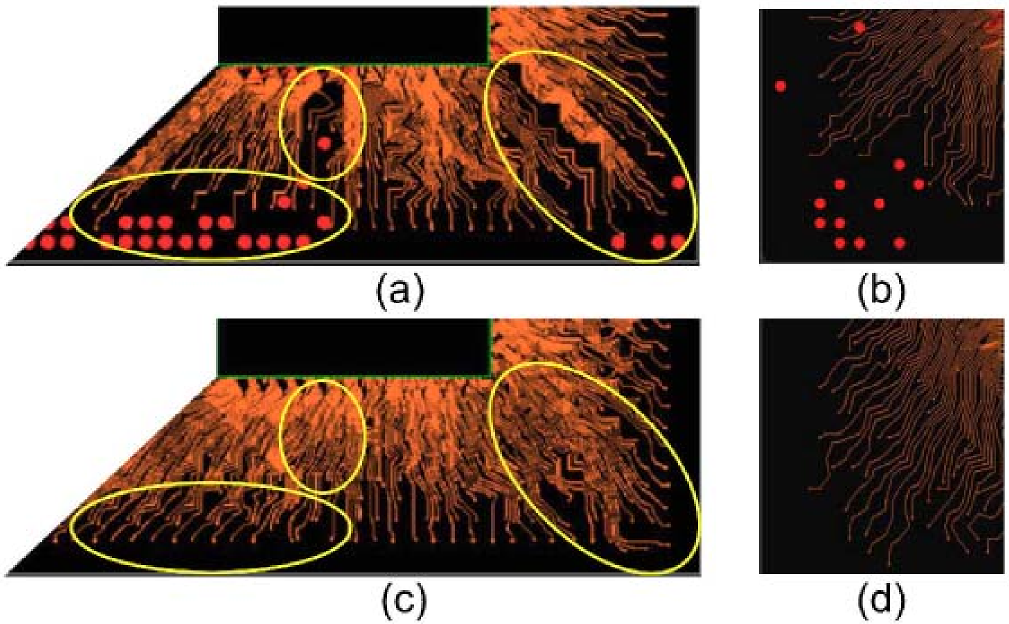


Routing (a) before and (b) after diffusion.

The left figure is a comparison of magnified view of the corner of cases B2 and X3.

(a) (b) is the results from [Liu et al., DAC, 2008]

(c) (d) is the results generated by D-Router



Outline

- Substrate Topological Routing
- Existing Work
- Problem Formulation
- Baseline Algorithms
- Motivating Examples
- Diffusion-Driven Congestion Reduction Algorithm
- Experimental Results
- Conclusions

Conclusions

- On-chip substrate routing for high density packages is challenging
- The existing substrate routing algorithms often result in a large number of unrouted nets that have to be routed manually
- D-Router
  - an effective yet efficient diffusion-driven method
  - improves routability by a simulated diffusion process based on the duality between congestion and concentration
  - Compared with a recently published A*-based algorithm used in a state of the art commercial tool, it reduces the number of unrouted nets by 4.6x, with an average 4.3x runtime reduction

Q&A

Thank you!

Dynamic pushing

- The “dynamic pushing” in [Liu et al., DAC, 2008] only pushes the blocking net wires, and does not “squeeze” through congested area

An example of “dynamic pushing” in routing two nets (a) routed net A blocks the shortest connection of net B, (b) net B pushes net A for the optimal solution.