



Ein Überblick über Placement und Routing in integrierten Schaltungen und Systemen

Jean-Pierre Schwickerath

<http://schwicky.net/projects/2004/asic/>

Technische Universität Darmstadt

Entwurf, Technik und Technologie integrierter Schaltungen und Systeme

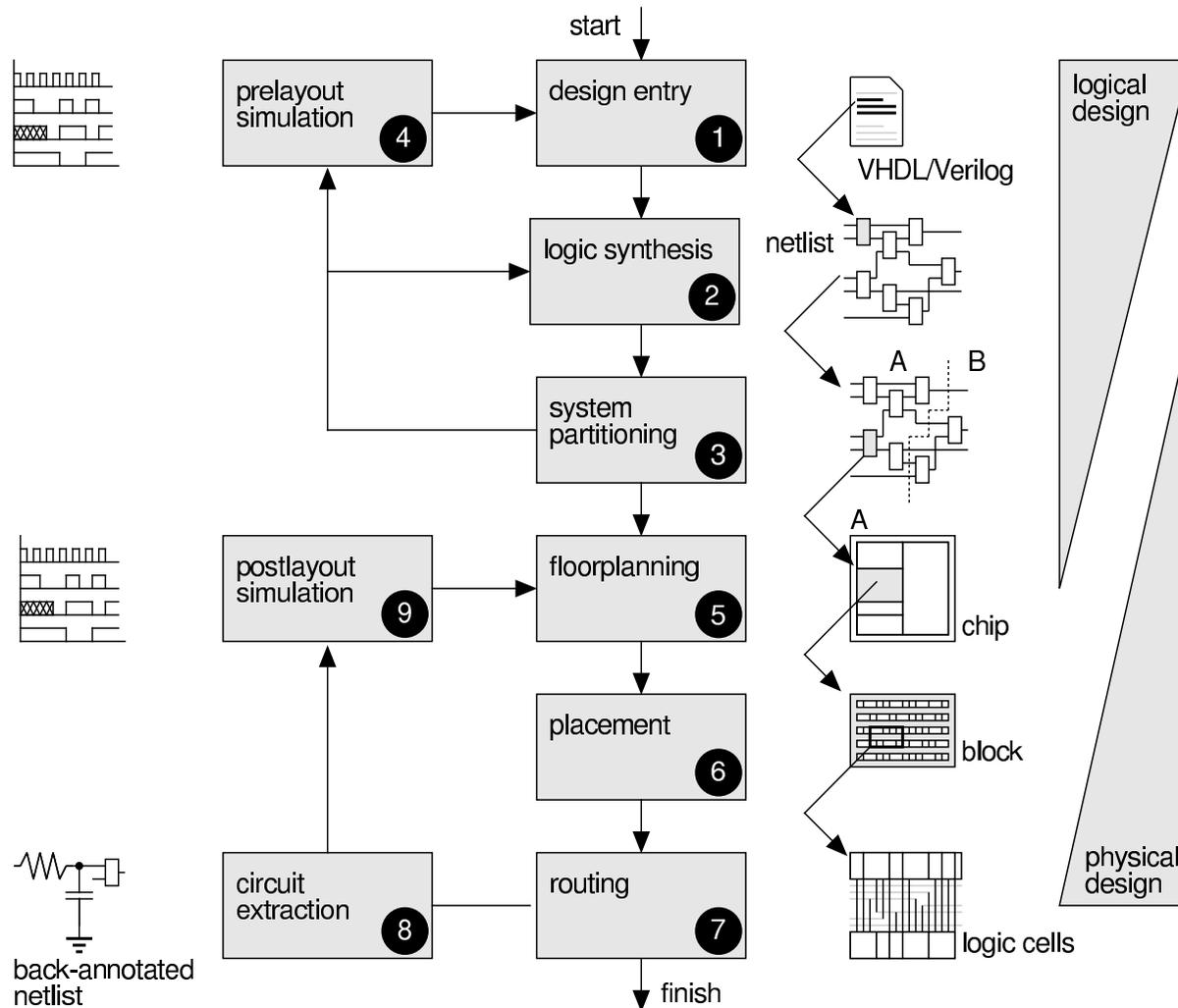
Prof. Rolf Hoffmann, Dipl.-Ing. Wolfgang Heenes



2. Juli 2004



ASIC-Entwurf



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.
 - Vorgaben: Minimierung der Fläche, Länge der Verbindungen. Gleichmässige Hitzeverteilung. Einhalten von Laufzeiten.



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.
 - Vorgaben: Minimierung der Fläche, Länge der Verbindungen. Gleichmässige Hitzeverteilung. Einhalten von Laufzeiten.
 - Genügend Platz für Routing → Routingregionen.



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.
 - Vorgaben: Minimierung der Fläche, Länge der Verbindungen. Gleichmässige Hitzeverteilung. Einhalten von Laufzeiten.
 - Genügend Platz für Routing → Routingregionen.
- **Routing:** Verbindung zwischen den Pins der Komponenten.



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.
 - Vorgaben: Minimierung der Fläche, Länge der Verbindungen. Gleichmässige Hitzeverteilung. Einhalten von Laufzeiten.
 - Genügend Platz für Routing → Routingregionen.
- **Routing:** Verbindung zwischen den Pins der Komponenten.
 - Vorgaben: Minimierung der Anzahl an Layern, Vias, Störungsquellen, Rauschen.



Definitionen



- **Placement:** Platzierung von Komponenten auf dem Substrat.
 - Vorgaben: Minimierung der Fläche, Länge der Verbindungen. Gleichmässige Hitzeverteilung. Einhalten von Laufzeiten.
 - Genügend Platz für Routing → Routingregionen.
- **Routing:** Verbindung zwischen den Pins der Komponenten.
 - Vorgaben: Minimierung der Anzahl an Layern, Vias, Störungsquellen, Rauschen.
 - Globales Routing und detailliertes Routing.



Konsequenzen



- Beim Placement müsste man schon wissen wie viel Platz das Routing braucht.



Konsequenzen



- Beim Placement müsste man schon wissen wie viel Platz das Routing braucht.
- Priorität: Platzierung von Komponenten mit wichtigen Auswirkungen.



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.
- Grosse, voneinander abhängige *NP*-harte Probleme:



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.
- Grosse, voneinander abhängige *NP*-harte Probleme:
 - Viele voneinander abhängige Kriterien,



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.
- Grosse, voneinander abhängige *NP*-harte Probleme:
 - Viele voneinander abhängige Kriterien,
 - Optimierung unter Berücksichtigung nicht trivialer Bedingungen.



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.
- Grosse, voneinander abhängige *NP*-harte Probleme:
 - Viele voneinander abhängige Kriterien,
 - Optimierung unter Berücksichtigung nicht trivialer Bedingungen.
- Abschätzungen müssen zur Hilfe genommen werden.



Natur der Probleme



- Placement und Routing-Probleme können auf mehreren Ebenen betrachtet werden.
- Grosse, voneinander abhängige *NP*-harte Probleme:
 - Viele voneinander abhängige Kriterien,
 - Optimierung unter Berücksichtigung nicht trivialer Bedingungen.
- Abschätzungen müssen zur Hilfe genommen werden.
- Einsatz von genetischen und stochastischen Verfahren.



Placement



Unter Verwendung

- einer Menge an rechteckigen Zellen mit Pins,
- einer Netlist (Verbindungsliste),
- einer ungefähren Länge W des Chips.



Placement



Unter Verwendung

- einer Menge an rechteckigen Zellen mit Pins,
- einer Netlist (Verbindungsliste),
- einer ungefähren Länge W des Chips.

Berechne

- die absolute Position jeder Zelle,
- die Orientierung und Spiegelung jeder Zelle,
- ein Rechteck B , das die Form des Chips definiert.



Placement (2)



Ziele:

- Minimierung der Fläche von B .



Placement (2)



Ziele:

- Minimierung der Fläche von B .
- Ohne Überlappung.



Placement (2)



Ziele:

- Minimierung der Fläche von B .
- Ohne Überlappung.
- Alle nicht belegten Flächen reichen fürs Routing.



Routing



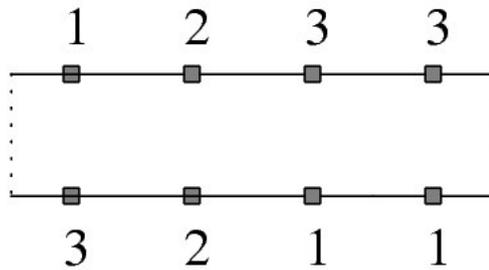
Keine zwei Nets dürfen sich auf einem Layer kreuzen.



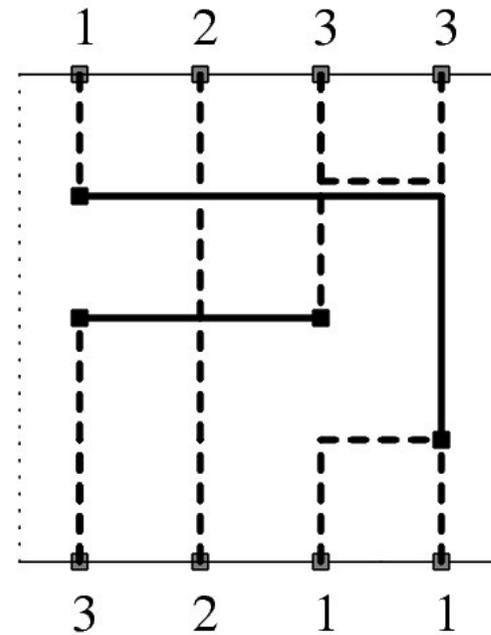
Routing



Keine zwei Nets dürfen sich auf einem Layer kreuzen.
Beispiel: Channel-Routing.



(a)



(b)



Simulated Annealing



- Basiert auf Monte-Carlo-Methode.



Simulated Annealing



- Basiert auf Monte-Carlo-Methode.
- Einfrieren einer Flüssigkeit oder Kristallisierung eines Metalls wird simuliert.



Simulated Annealing



- Basiert auf Monte-Carlo-Methode.
- Einfrieren einer Flüssigkeit oder Kristallisierung eines Metalls wird simuliert.
- Die Substanz muss sich immer beinahe im thermodynamischen Gleichgewicht befinden.



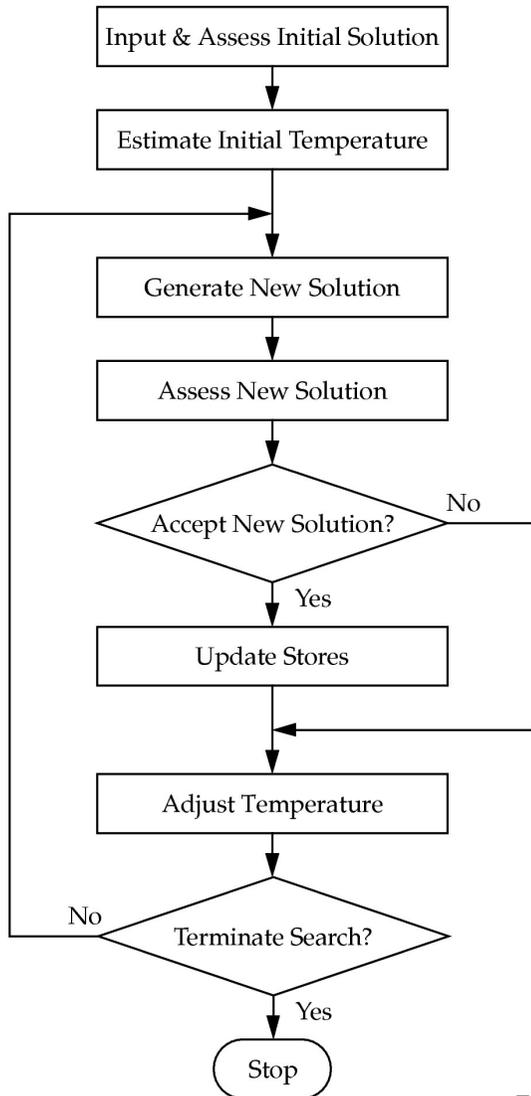
Simulated Annealing



- Basiert auf Monte-Carlo-Methode.
- Einfrieren einer Flüssigkeit oder Kristallisierung eines Metalls wird simuliert.
- Die Substanz muss sich immer beinahe im thermodynamischen Gleichgewicht befinden.
- Von Natur aus sequenziell.



Simulated Annealing (2)



Genetische Algorithmen



- Suchen mit probabilistischen Übergängen nach vorteilhaften Anpassungen in einer sich immer ändernden Umgebung.



Genetische Algorithmen



- Suchen mit probabilistischen Übergängen nach vorteilhaften Anpassungen in einer sich immer ändernden Umgebung.
- Bevölkerung, Chromosome, Individuen, Evolutionsprozess:



Genetische Algorithmen



- Suchen mit probabilistischen Übergängen nach vorteilhaften Anpassungen in einer sich immer ändernden Umgebung.
- Bevölkerung, Chromosome, Individuen, Evolutionsprozess:
 - Selektion,



Genetische Algorithmen



- Suchen mit probabilistischen Übergängen nach vorteilhaften Anpassungen in einer sich immer ändernden Umgebung.
- Bevölkerung, Chromosome, Individuen, Evolutionsprozess:
 - Selektion,
 - Kreuzung (crossover oder recombination),



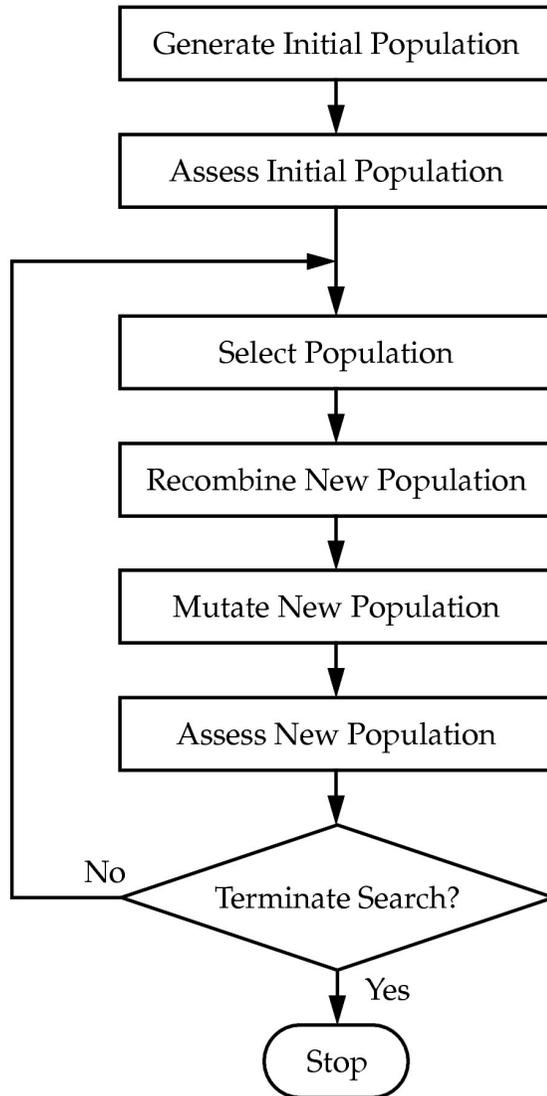
Genetische Algorithmen



- Suchen mit probabilistischen Übergängen nach vorteilhaften Anpassungen in einer sich immer ändernden Umgebung.
- Bevölkerung, Chromosome, Individuen, Evolutionsprozess:
 - Selektion,
 - Kreuzung (crossover oder recombination),
 - Mutation.



Genetische Algorithmen (2)



Genetische Algorithmen (3)



Sind für CAD-Probleme bestens geeignet:

- Relative Performanz verbessert sich mit der Komplexität.



Genetische Algorithmen (3)



Sind für CAD-Probleme bestens geeignet:

- Relative Performanz verbessert sich mit der Komplexität.
- Sehr gut parallelisierbar. Beinahe lineare Skalierung auf MIMD. Rechenintensivste: Kostenfunktion.



Genetische Algorithmen (3)



Sind für CAD-Probleme bestens geeignet:

- Relative Performanz verbessert sich mit der Komplexität.
- Sehr gut parallelisierbar. Beinahe lineare Skalierung auf MIMD. Rechenintensivste: Kostenfunktion.
- Sehr schnell gute Ergebnisse. Bei langer Laufzeit, sehr gute Ergebnisse.



Genetische Algorithmen (3)



Sind für CAD-Probleme bestens geeignet:

- Relative Performanz verbessert sich mit der Komplexität.
- Sehr gut parallelisierbar. Beinahe lineare Skalierung auf MIMD. Rechenintensivste: Kostenfunktion.
- Sehr schnell gute Ergebnisse. Bei langer Laufzeit, sehr gute Ergebnisse.

Aber:

- Irgendwann nur noch sehr kleine Verbesserungen.
- Probleme mit lokalen Optima.



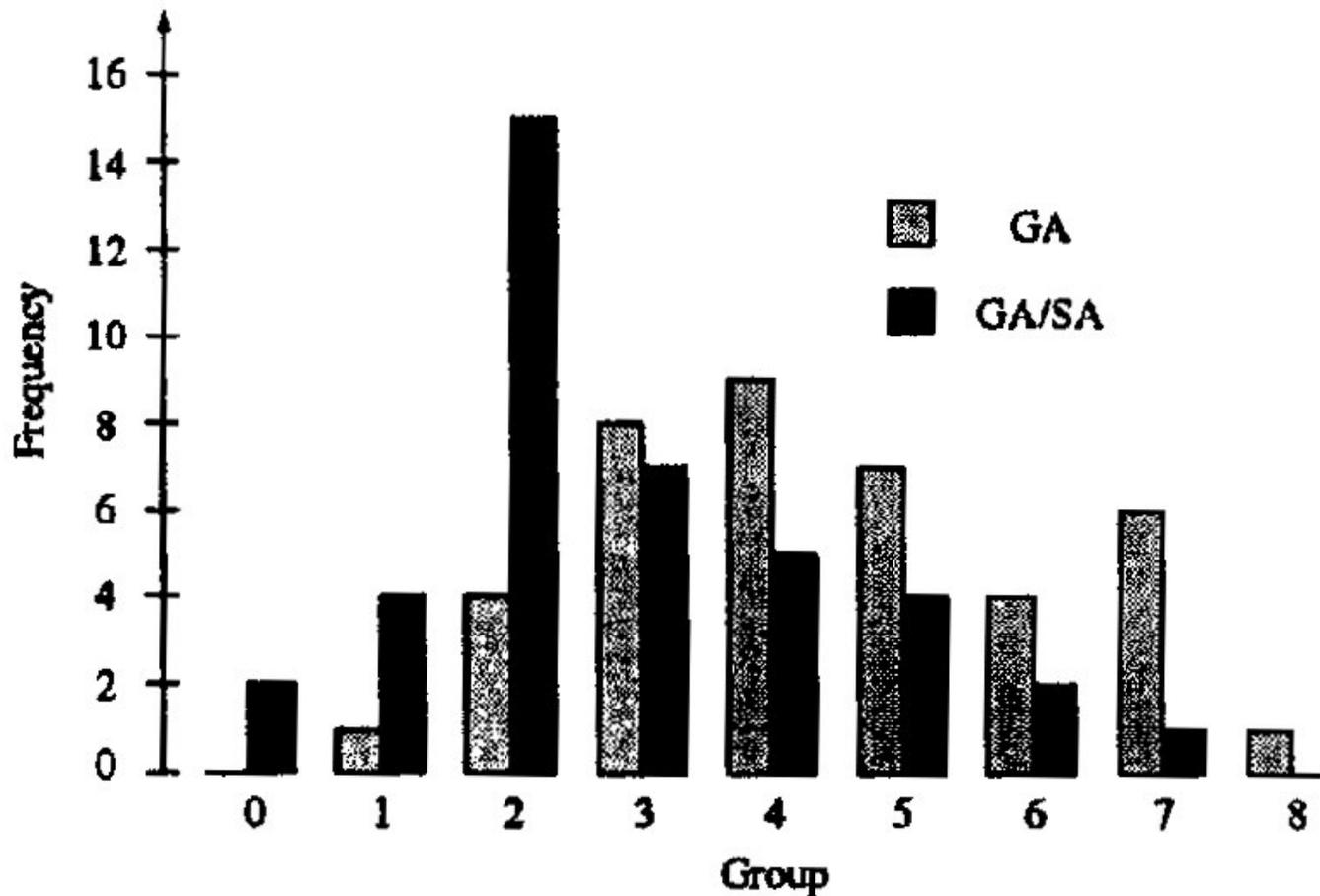
Die Lösung: SAGA

```
generate( $\Pi_c$ );  
 $\forall s \in \Pi_c : T_s = \perp$ ;  
evaluate( $\Pi_c$ );  
 $q = \text{bestOf}(\Pi_c)$ ;  
 $c_R = 0$ ;  
repeat until stopCriteria():  
  if no improvement for  $R$  generations do :  
     $c_R = c_R + 1$ ;  
     $M = \max(\text{round}(\beta^{c_R} M), 1)$ ;  
     $\Pi_c = \text{reduce}(\Pi_c, M)$ ;  
     $p_{mut} = \min(\gamma p_{mut}, 1.0)$ ;  
  end;  
   $\Pi_n = \emptyset$ ;  
  repeat  $M$  times:  
    select  $s \in \Pi_c, t \in \Pi_c$ ;  
     $v = \text{crossover}(s, t)$ ;  
     $T_v = \perp$ ;  
     $\Pi_n = \Pi_n \cup \{v\}$ ;  
  end;  
  evaluate( $\Pi_c \cup \Pi_n$ );  
   $\Pi_c = \text{reduce}(\Pi_c \cup \Pi_n, M)$ ;  
   $\forall s \in \Pi_c : s = \text{SAMutate}(s)$ ;  
   $\forall s \in \Pi_c : \text{with prob. } p_{inv} \text{ do :}$   
     $s = \text{invert}(s)$ ;  
  evaluate( $\Pi_c$ );  
   $q = \text{bestOf}(\Pi_c \cup \{q\})$ ;  
end;  
 $\forall t \in \Pi_c \cup \{q\} : t = \text{optimize}(t)$ ;  
 $r = \text{bestOf}(\Pi_c \cup \{q\})$ ;
```

SAGA Benchmarks



Xerox Benchmark: Anzahl der Durchläufe im Intervall $[27.15 + 0.3i, 27.15 + 0.3(i + 1)[$.



SAGA Benchmarks (2)

Benchmark	System	Area (mm ²)
Apte	SAGA	53.58
	BB [5]	54.05
	Seattle Silicon [6]	54.77
Xerox	Seattle Silicon [6]	25.79
	BB [5]	26.17
	SAGA	27.15
	BEAR [3]	28.47
	MOSAICO ¹	29.01
	VITAL ¹	31.71
Hp	SAGA	11.81
	Seattle Silicon [6]	11.85
	BB [5]	12.15

Placement im Detail



- Länge der Verbindung werden immer wichtiger:
Verzögerung von Quelle zu Senke.



Placement im Detail



- Länge der Verbindung werden immer wichtiger:
Verzögerung von Quelle zu Senke.
- Verstärkt durch Widerstand und Kapazität.



Placement im Detail



- Länge der Verbindung werden immer wichtiger:
Verzögerung von Quelle zu Senke.
- Verstärkt durch Widerstand und Kapazität.
- Konstruktive Verfahren, meistens mit Min-Cut Algorithmen
durch sukzessive Anwendung von Partitioning.



Placement im Detail

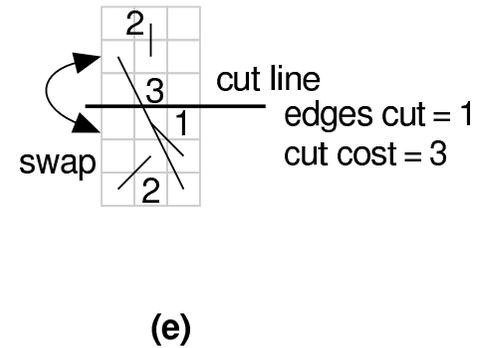
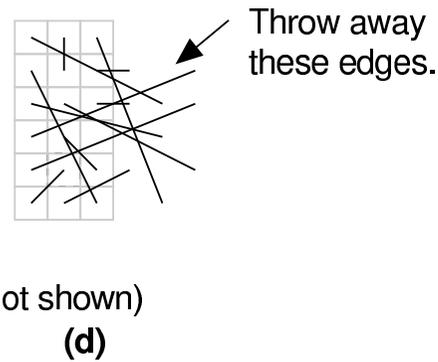
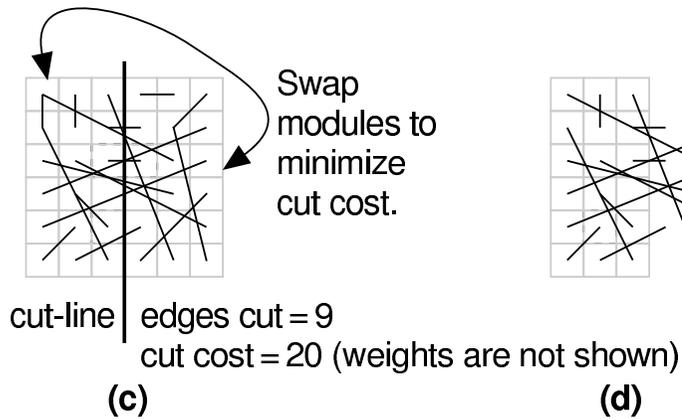
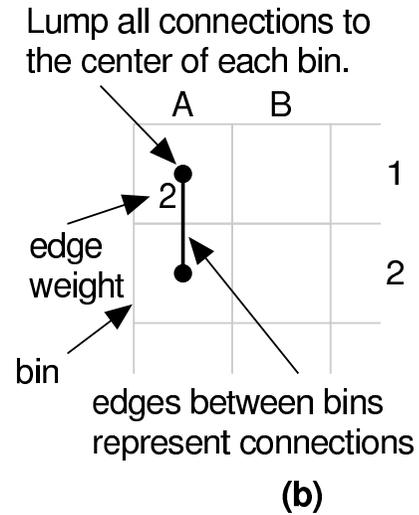
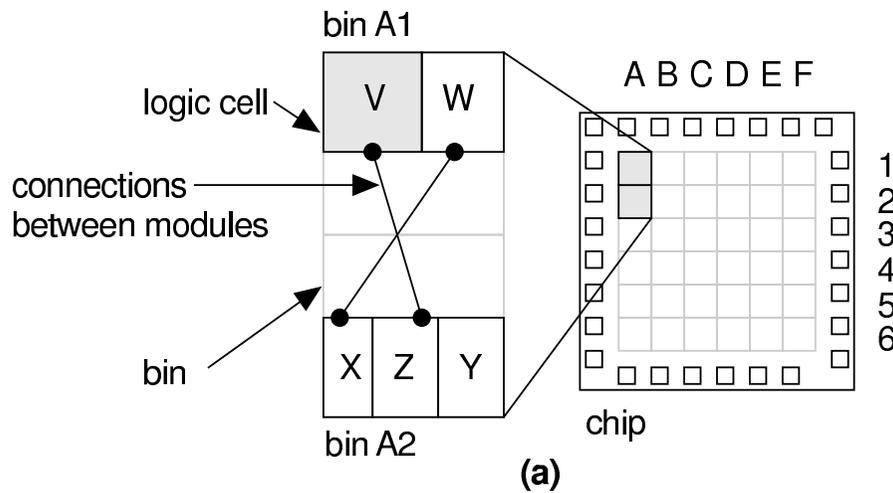


- Länge der Verbindung werden immer wichtiger:
Verzögerung von Quelle zu Senke.
- Verstärkt durch Widerstand und Kapazität.
- Konstruktive Verfahren, meistens mit Min-Cut Algorithmen
durch sukzessive Anwendung von Partitioning.
- Produzieren ein vollständiges Placement.





Min-Cut Placement



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.
- Simulated Annealing:



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.
- Simulated Annealing:
 - Wähle Zelle für den Tausch.



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.
- Simulated Annealing:
 - Wähle Zelle für den Tausch.
 - Berechne Funktion E für das neue Placement.



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.
- Simulated Annealing:
 - Wähle Zelle für den Tausch.
 - Berechne Funktion E für das neue Placement.
 - Anhand des Messkriteriums wird entschieden ob die Zelle getauscht wird.



Placement im Detail (2)



Iterative Verfahren:

- Selektionskriterium, Messkriterium.
- Simulated Annealing:
 - Wähle Zelle für den Tausch.
 - Berechne Funktion E für das neue Placement.
 - Anhand des Messkriteriums wird entschieden ob die Zelle getauscht wird.
 - Senke die Temperatur und wiederhole.



Genetisches Routing

```
create initial population ( $\mathcal{P}_c$ )
fitness_calculation ( $\mathcal{P}_c$ )
 $p_{best} = \text{best\_individual} (\mathcal{P}_c)$ 
for generation = 1 until max_generation
     $\mathcal{P}_n = \emptyset$ 
    for offspring = 1 until max_descendant
         $p_\alpha = \text{selection} (\mathcal{P}_c)$ 
         $p_\beta = \text{selection} (\mathcal{P}_c)$ 
         $\mathcal{P}_n = \mathcal{P}_n \cup \text{crossover} (p_\alpha, p_\beta)$ 
    endfor
    fitness_calculation ( $\mathcal{P}_n$ )
     $\mathcal{P}_c = \text{reduction} (\mathcal{P}_c \cup \mathcal{P}_n)$ 
     $p_{best} = \text{best\_individual} (p_{best} \cup \mathcal{P}_c)$ 
    mutation ( $\mathcal{P}_c$ )
    fitness_calculation ( $\mathcal{P}_c$ )
endfor
optimize ( $p_{best}$ )
```

Genetisches Routing (2)



- Bevölkerung: \mathcal{P}_c .



Genetisches Routing (2)



- Bevölkerung: \mathcal{P}_c .
- Lösung: Individuum (p_{best}).



Genetisches Routing (2)



- Bevölkerung: \mathcal{P}_c .
- Lösung: Individuum (p_{best}).
- Qualität: Fitness.



Genetisches Routing (2)



- Bevölkerung: \mathcal{P}_c .
- Lösung: Individuum (p_{best}).
- Qualität: Fitness.
- Selektion, Crossover, Mutation.



Genetisches Routing (2)



- Bevölkerung: \mathcal{P}_c .
- Lösung: Individuum (p_{best}).
- Qualität: Fitness.
- Selektion, Crossover, Mutation.

--- Connection on layer 1 (pin layer)

— Connection on layer 2

--- Extension line on layer 1

— Extension line on layer 2

■ Via

□ Layer change of extension lines



Benchmarks: Channel-Routing

Benchmark	System	Col.	Rows	Netlength	Vias
Yoshimura-Kuh channel	Yosh.-Kuh [34]	12	5	75	21
	Weaver [19]	12	4	67	12
	Monreale [11]	12	4	72	11
	Our work	12	4	70	11
Joo6_12	Weaver [19]	12	4	79	14
	Packer [12]	12	4	82	18
	Monreale [11]	12	4	84	13
	Our work	12	4	79	14
Joo6_13	Greedy [28]	18	8	194	38
	Weaver [19]	18	7	169	29
	Silk [23]	18	6	171	28
	Packer [12]	18	6	167	25
	Our work	18	6	165	25
Joo6_16	Weaver [19]	11	8	131	23
	Weaver ^a [19]	11	7	121	21
	Monreale [11]	11	7	120	19
	Our work	11	6	116	15
Burstein's difficult channel	Mighty [32]	13 ^b	4	83	8
	Packer [12]	12	4	82	10
	Monreale [11]	12	4	82	10
	Our work	12	4	82	8

^a interactively

^b additional column in the middle of the channel

Zusammenfassung



- Genetische und Stochastische Verfahren sind sehr gut kombinierbar.



Zusammenfassung



- Genetische und Stochastische Verfahren sind sehr gut kombinierbar.
- Liefern qualitativ sehr gute Ergebnisse.



Zusammenfassung



- Genetische und Stochastische Verfahren sind sehr gut kombinierbar.
- Liefern qualitativ sehr gute Ergebnisse.
- Optimal für Parallelisierbarkeit.



Zusammenfassung



- Genetische und Stochastische Verfahren sind sehr gut kombinierbar.
- Liefern qualitativ sehr gute Ergebnisse.
- Optimal für Parallelisierbarkeit.
- Untersuchungen haben gezeigt, dass aktuelle Werkzeuge noch sehr verbesserungswürdig sind.



Zusammenfassung



- Genetische und Stochastische Verfahren sind sehr gut kombinierbar.
- Liefern qualitativ sehr gute Ergebnisse.
- Optimal für Parallelisierbarkeit.
- Untersuchungen haben gezeigt, dass aktuelle Werkzeuge noch sehr verbesserungswürdig sind.

Vielen Dank für die Aufmerksamkeit!

