On the Complexity of Register Coalescing

F. Bouchez, A. Darte, and F. Rastello

LIP UMR CNRS — Inria — ENS Lyon — UCBL
France

CGO’07, San Jose
Outline

1. What, Why, and How to Coalesce
   - Basic Formulation
   - The Different Approaches

2. A Hard Optimization Problem

3. Conclusion: What should we Implement Now?
Outline

1. What, Why, and How to Coalesce
   - Basic Formulation
   - The Different Approaches

2. A Hard Optimization Problem

3. Conclusion: What should we Implement Now?
Coalescing is

- rename 2 variables into a unique representant
- MOVE $A, B$: an *affinity* between $A$ and $B$
- $A$ and $B$ cannot share the same resource: an *interference* between $A$ and $B$

\[
\begin{align*}
a &= \\
d &= a \\
\cdots &= a \\
c &= \\
d &= b \\
\cdots &= c \\
\cdots &= d
\end{align*}
\]
Coalescing: Coloring the Interference/Affinity Graph

Coalescing is

- rename 2 variables into a unique representant
- MOVE $A, B$: an affinity between $A$ and $B$
- $A$ and $B$ cannot share the same resource: an interference between $A$ and $B$

```
ad = ...
b = ...
... = ad
c = ...
ad = b
... = c
... = ad
```
Coalescing: Coloring the Interference/Affinity Graph

Coalescing is

- rename 2 variables into a unique representant
- MOVE \(A, B\): an *affinity* between \(A\) and \(B\)
- \(A\) and \(B\) cannot share the same resource: an *interference* between \(A\) and \(B\)

![Diagram of coalescing process]

\[
\begin{align*}
ad &= \ldots \\
\ldots &= ad \\
c &= \ldots \\
ad &= b \\
\ldots &= c \\
\ldots &= ad
\end{align*}
\]
Many MOVE instructions due to

- register constraints (function call, 2 address instructions, etc.)
- SSA construction followed by basic SSA destruction

\[
\begin{align*}
A &= \ldots \\
B &= \ldots \\
\text{MOVE } R0, A \\
\text{MOVE } R1, B \\
D &= f(A, B) \\
\text{MOVE } D, R0 \\
C_1 &= \ldots \\
C_2 &= \ldots \\
C &= \phi(C_1, C_2) \\
\text{use } C
\end{align*}
\]
Many \texttt{MOVE} instructions due to

- register constraints (function call, 2 address instructions, etc.)
- SSA construction followed by basic SSA destruction

\begin{align*}
A &= \ldots \\
B &= \ldots \\
\text{MOVE } &R0, A \\
\text{MOVE } &R1, B \\
\text{call } &f \\
\text{MOVE } &D, R0 \\
C_1 &= \ldots \\
C_2 &= \ldots \\
C &= \phi(C_1, C_2) \\
\text{use } &C
\end{align*}
Many MOVE instructions due to

- register constraints (function call, 2 address instructions, etc.)
- SSA construction followed by basic SSA destruction

\[ A = \ldots \]
\[ B = \ldots \]
\[ \text{MOVE } R0, A \]
\[ \text{MOVE } R1, B \]
\[ \text{call } f \]
\[ \text{MOVE } D, R0 \]

\[ C_1 = \ldots \]
\[ C_2 = \ldots \]
\[ \text{MOVE } C, C_1 \]
\[ \text{MOVE } C, C_2 \]

\[ C = \phi(C_1, C_2) \]

use \[ C \]
Many MOVE... to remove

Many MOVE instructions due to

- register constraints (function call, 2 address instructions, etc.)
- SSA construction followed by basic SSA destruction

Our past experience

(aggresive) coalescing during SSA destruction → pre-sched. → reg. alloc. (Iterared) → sched.

- on most benchmarks, a good speedup
- on some of them, slow down!
**Aggressive** coalescing may lead to spilling. Coalescing aware of colorability is **conservative**.
Aggressive coalescing may lead to spilling. Coalescing aware of colorability is conservative.
Aggressive coalescing may lead to spilling. Coalescing aware of colorability is conservative.
Aggressive coalescing may lead to spilling. Coalescing aware of colorability is conservative.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- *k*-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- \(k\)-colorability check is hard, but checking the Greedy-\(k\)-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-$k$-colorability** is easy.

- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...

- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
Coalescing

- What, Why, and How to Coalesce
- The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental** coalescing...
- Incremental is not optimal. **Decoalescing** is better.

![Diagram of coalescing process]
Coalescing

What, Why, and How to Coalesce

The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.

- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...

- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
Coalescing

- What, Why, and How to Coalesce
- The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental** coalescing...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
Coalescing

What, Why, and How to Coalesce

The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the \textit{Greedy-$k$-colorability} is easy.
- Still, finding the optimal subset of affinities is hard. We do \textit{Incremental} coalescing...
- Incremental is not optimal. \textit{Decoalescing} is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do Incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental** coalescing...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental** coalescing...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do incremental coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
$k$-colorability check is hard, but checking the **Greedy-$k$-colorability** is easy.

Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...

Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental** coalescing...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- *k*-colorability check is hard, but checking the [Greedy-*k*-colorability](#) is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-k-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the **Greedy-$k$-colorability** is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-k-colorablility* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-k-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-k-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental coalescing*...
- Incremental is not optimal. *Decoalescing* is better.
Coalescing

- What, Why, and How to Coalesce
- The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the Greedy-$k$-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing...**
- Incremental is not optimal. **Decoalescing** is better.

![Graph Diagram]

- Not greedy-3-colorable
...and the speedup

- *k*-colorability check is hard, but checking the *Greedy-k-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
Coalescing

- What, Why, and How to Coalesce
- The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.
...and the speedup

- $k$-colorability check is hard, but checking the \textit{Greedy-$k$-colorability} is easy.
- Still, finding the optimal subset of affinities is hard. We do \textit{Incremental coalescing}...
- Incremental is not optimal. \textit{Decoalescing} is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-k-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do **Incremental coalescing**...
- Incremental is not optimal. **Decoalescing** is better.

![Diagram of graph with nodes a to jkl and edges connecting them. Some nodes are labeled with letters a to m and some with i, jkl, e, f, g, h, d, c. There is a note that says "greedy-3-colorable".]
**Coalescing**

- What, Why, and How to Coalesce
- The Different Approaches

...and the speedup

- $k$-colorability check is hard, but checking the *Greedy*-k-colorability is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the \textit{Greedy-$k$-colorability} is easy.
- Still, finding the optimal subset of affinities is hard. We do \textit{Incremental} coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the \textit{Greedy-}$k$-\textcolor{red}{\textit{colorability}} is easy.
- Still, finding the optimal subset of affinities is hard. We do \textit{Incremental} coalescing...
- Incremental is not optimal. Decoalescing is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-*$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.

---

**Coalescing**

- What, Why, and How to Coalesce
- The Different Approaches

---

**Aggressive coalescing**

- **Conservative coalescing**
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.

![Diagram showing different coalescing approaches: Aggressive, Conservative, and Incremental conservative coalescing.](image)
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-k-colorablility* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. *Decoalescing* is better.

---

**Aggressive coalescing**

**Conservative coalescing**

**Incremental conservative coalescing**

**Decoalescing**
...and the speedup

- $k$-colorability check is hard, but checking the *Greedy-$k$-colorability* is easy.
- Still, finding the optimal subset of affinities is hard. We do *Incremental* coalescing...
- Incremental is not optimal. Decoalescing is better.
Outline

1. What, Why, and How to Coalesce
   - Basic Formulation
   - The Different Approaches

2. A Hard Optimization Problem

3. Conclusion: What should we Implement Now?
Coalescing is Hard

\( G \): greedy-\( k \)-colorable interference graph,

**Aggressive coalescing** NP-complete, even with \( k = 3 \).

**Conservative coalescing** NP-complete even if \( k = 3 \) and only affinities can be merged.

**Incremental conservative coalescing (Briggs, George)**

NP-complete if \( G \) is arbitrary.

**Open** if \( G \) is greedy-\( k \)-colorable.

**Optimistic coalescing (Park & Moon) = conservative de-coalescing**

NP-complete even if \( k = 4 \).
A little hope: Chordal Graphs

- Interference graph of SSA programs
- $k$-colorability easy on chordal graphs
- $\text{MAXLIVE} = w(G) = \chi(G)$
- $k$-chordal $\subset$ greedy-$k$-colorable $\subset$ $k$-colorable
A little hope: Chordal Graphs

- Interference graph of SSA programs
- $k$-colorability easy on chordal graphs
- $\text{MAXLIVE} = w(G) = \chi(G)$
- $k$-chordal $\subset$ greedy-$k$-colorable $\subset$ $k$-colorable
Coalescing

A Hard Optimization Problem

A little hope: Chordal Graphs

- Interference graph of SSA programs
- \( k \)-colorability easy on chordal graphs
- \( \text{MAXLIVE} = w(G) = \chi(G) \)
- \( k \)-chordal \( \subset \) greedy-\( k \)-colorable \( \subset \) \( k \)-colorable
A little hope: Chordal Graphs

- Interference graph of SSA programs
- $k$-colorability easy on chordal graphs
- $\text{MAXLIVE} = w(G) = \chi(G)$
- $k$-chordal $\subset$ greedy-$k$-colorable $\subset$ $k$-colorable
A little hope: Chordal Graphs

- Interference graph of SSA programs
- $k$-colorability easy on chordal graphs
- $\text{MAXLIVE} = w(G) = \chi(G)$
- $k$-chordal $\subset$ greedy-$k$-colorable $\subset$ $k$-colorable
A little hope: Chordal Graphs

- Interference graph of SSA programs
- $k$-colorability easy on chordal graphs
- $\text{MAXLIVE} = w(G) = \chi(G)$
- $k$-chordal $\subset$ greedy-$k$-colorable $\subset$ $k$-colorable
Coalescing

A Hard Optimization Problem

... but Coalescing is still Hard

\[ G: k \text{-chordal interference graph.} \]

**Aggressive coalescing** NP-complete.

**Conservative coalescing** NP-complete.

**Incremental conservative coalescing (Briggs, George)** Polynomial!

**Optimistic coalescing (Park & Moon) = conservative de-coalescing** NP-complete even if \( k = 4 \).
"Multiple-move" incremental

One move to coalesce

$k$-col
$k$-greedy
$k$-chordal

$k$-col
$k$-greedy
$k$-chordal

simple test
Coalescing

A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$\mathbf{NP}$

simple test
“Multiple-move” incremental

One move to coalesce

$k$-col
$k$-greedy
$k$-chordal

$k$-col
$k$-greedy
$k$-chordal

\[ \text{NP} \]

simple test
Coalescing
A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col
$k$-greedy
$k$-chordal

$k$-col
$k$-greedy
$k$-chordal

NP
?

P

simple test
Coalescing

A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col

$k$-greedy

$k$-chordal

$\mathbb{NP}$

$\mathbb{P}$

simple test

$\text{Yes/No?}$
Coalescing

A Hard Optimization Problem

"Multiple-move" incremental

One move to coalesce

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$NP$?

$P$

$?$

simple test

yes/no? NO
Coalescing
A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col
$k$-greedy
$k$-chordal

$k$-col
$k$-greedy
$k$-chordal

simple test

$k$-col
$k$-greedy
$k$-chordal

$\mathbb{NP}$

$?$

$\mathbb{P}$

yes/no?

$k$-col
$k$-greedy
$k$-chordal

$k$-col
$k$-greedy
$k$-chordal

yes/no? NO
Coalescing

A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

simple test

$\text{NP}$

$\text{P}$

yes/no? $\text{NO}$
Coalescing

A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

$k$-col

$k$-greedy

$k$-chordal

One move to coalesce

NP

??

P

simple test

yes/no? YES

yes/no? NO
Coalescing

A Hard Optimization Problem

“Multiple-move” incremental

One move to coalesce

$k$-col\,\,$k$-greedy\,\,$k$-chordal

$k$-col\,\,$k$-greedy\,\,$k$-chordal

simple test

Additional coalescings

yes/no? YES

yes/no? NO

NP

?=NP

P

$k$-chordal\,\,$k$-greedy\,\,$k$-col

$k$-chordal\,\,$k$-greedy\,\,$k$-col

yes/no?

NO
Coalescing

Conclusion: What should we Implement Now?

Outline

1. What, Why, and How to Coalesce
   - Basic Formulation
   - The Different Approaches

2. A Hard Optimization Problem

3. Conclusion: What should we Implement Now?
Coalescing

Conclusion: What should we Implement Now?

Some measurements

Aggressive+decoalescing scheme:
- optimizing the aggressive part is important
- decoalescing (optimistic) can still be improved

Incremental scheme:
- conservative rules (Briggs, George) are far from the optimal
Some measurements

**Aggressive+decoalescing scheme:**
- optimizing the aggressive part is important
- decoalescing (optimistic) can still be improved

**Incremental scheme:**
- conservative rules (Briggs, George) are far from the optimal

![Diagram showing weight of moves for Decoalescing]
Coalescing

Conclusion: What should we Implement Now?

Some measurements

Aggressive+decoalescing scheme:
- optimizing the aggressive part is important
- decoalescing (optimistic) can still be improved

Incremental scheme:
- conservative rules (Briggs, George) are far from the optimal
Most problems are NP-complete ⇒ heuristics!

Aggressive+decoalescing scheme:
- Aggressive coalescing is an important issue!
- Still gap for improving decoalescing;

Incremental scheme:
- A large gap for incremental;
- Promising approach: multiple-move incremental on $k$-greedy.