

编译系统 第八章 代码优化

哈尔滨工业大学 陈鄞



第19讲（ 代码优化_4 ）要点

- 基于数据流分析技术的代码优化

常用的优化方法

- 删除公共子表达式
- 删除无用代码
- 常量合并
- 代码移动
- 强度削弱
- 删除归纳变量

常用的优化方法

➤ 删除公共子表达式

- 如果表达式 $x \text{ op } y$ 先前已被计算过，并且从先前的计算到现在， $x \text{ op } y$ 中变量的值没有改变，那么 $x \text{ op } y$ 的这次出现就称为 **公共子表达式** (*common subexpression*)
- 如果上一次对该表达式的计算位于 **同一个基本块中**，则它是一个 **局部公共子表达式**；否则它是一个 **全局公共子表达式**

常用的优化方法

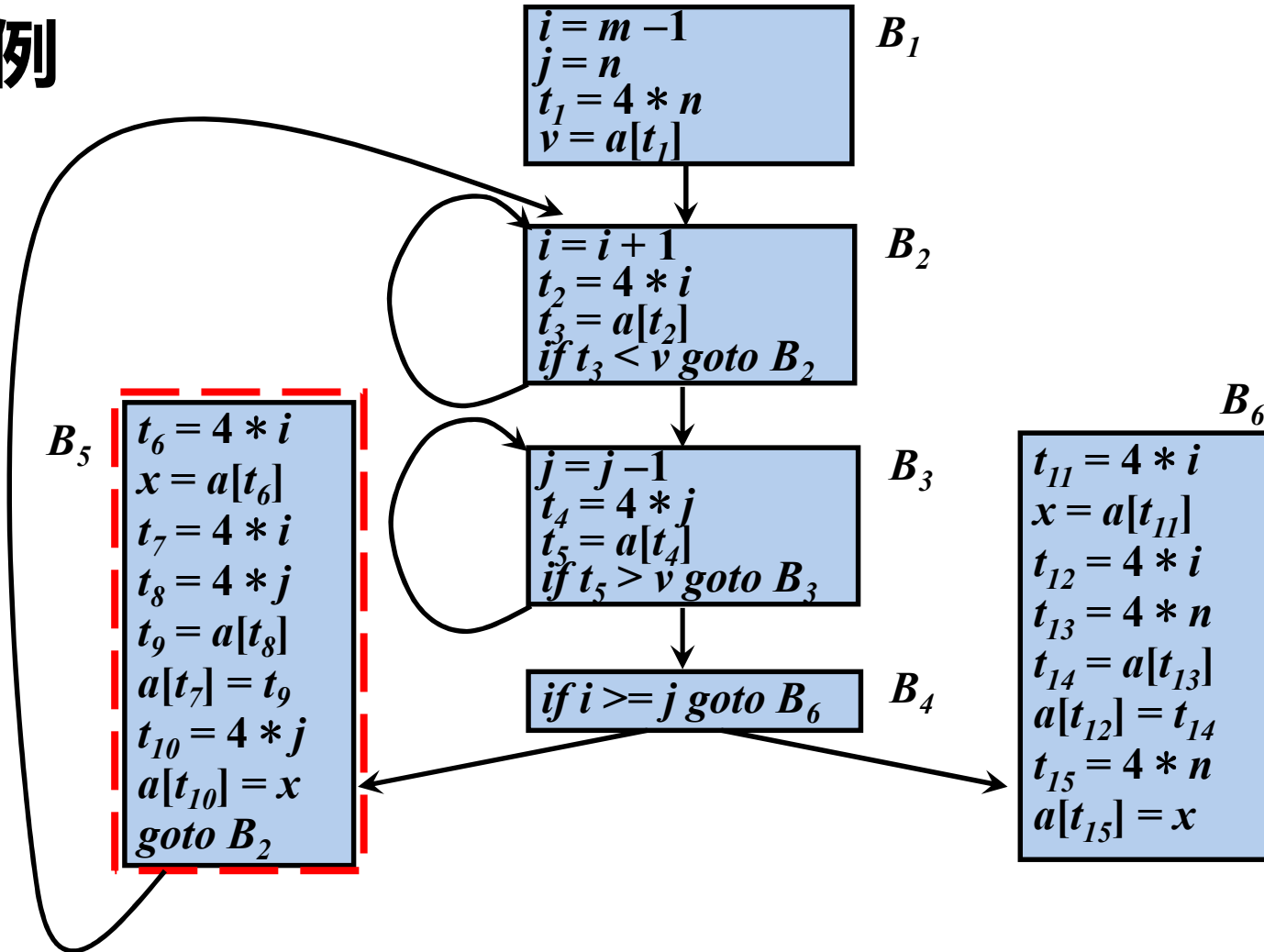
➤ 删除公共子表达式

删除局部公共子表达式

DAG



例



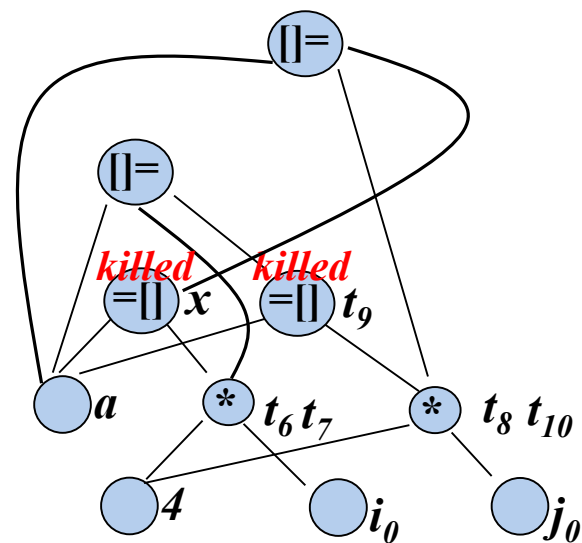
例

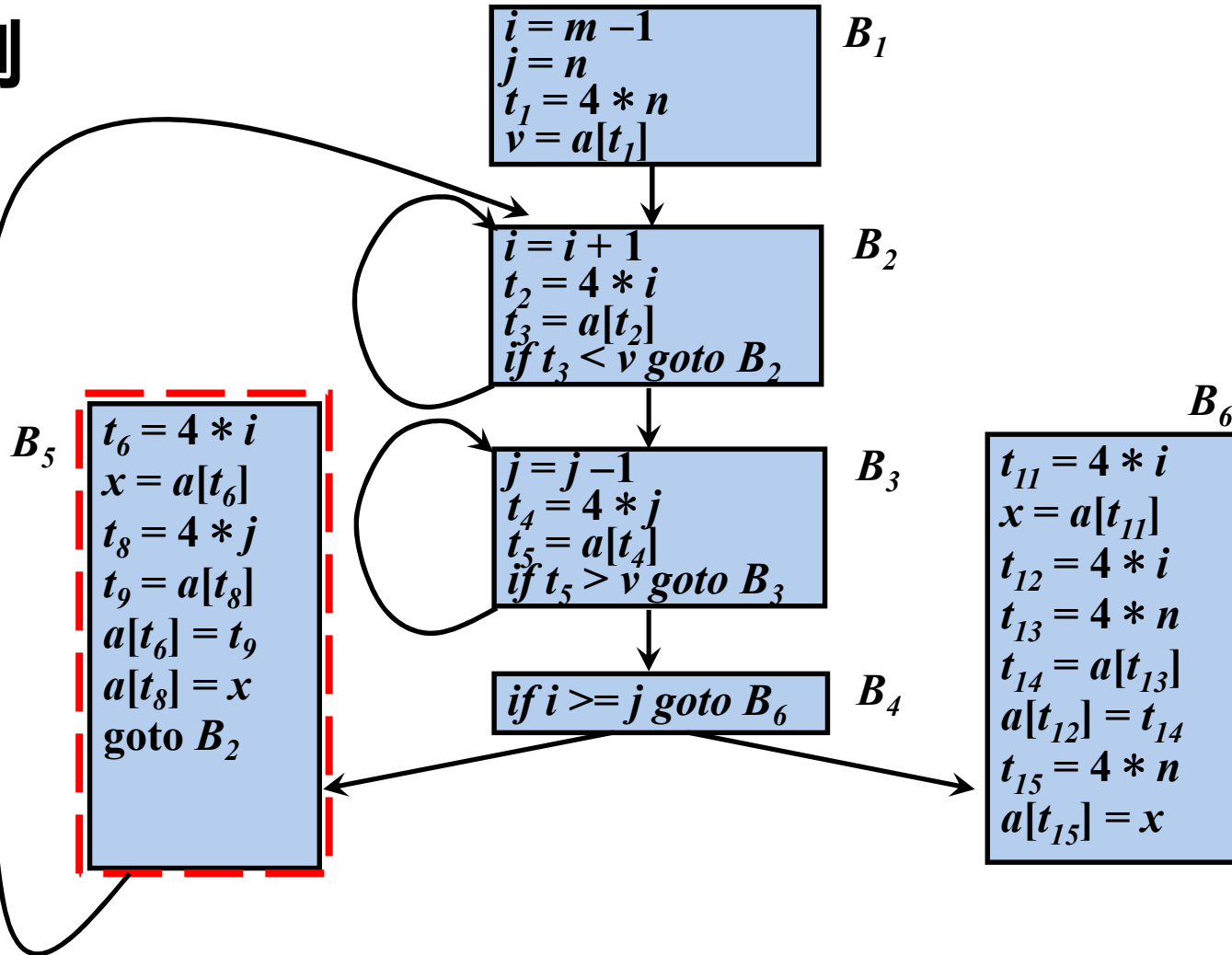
局部公共
子表达式

B_5

$t_6 = 4 * i$	←
$x = a[t_6]$	←
$t_7 = 4 * i$	←
$t_8 = 4 * j$	←
$t_9 = a[t_8]$	←
$a[t_7] = t_9$	←
$t_{10} = 4 * j$	←
$a[t_{10}] = x$	←
$goto B_2$	←

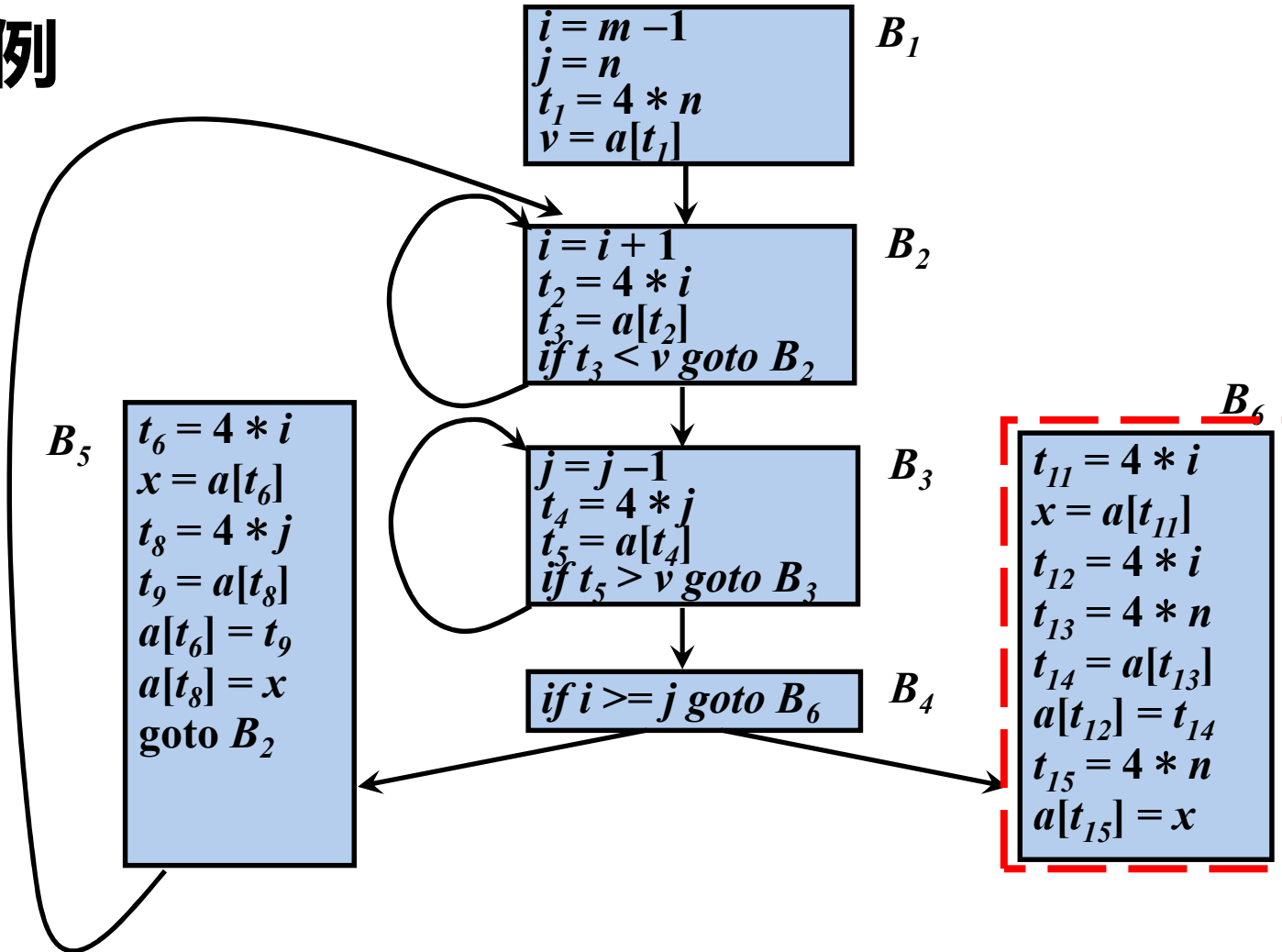
$t_6 = 4 * i$
$x = a[t_6]$
$t_8 = 4 * j$
$t_9 = a[t_8]$
$a[t_6] = t_9$
$a[t_8] = x$
$goto B_2$







例

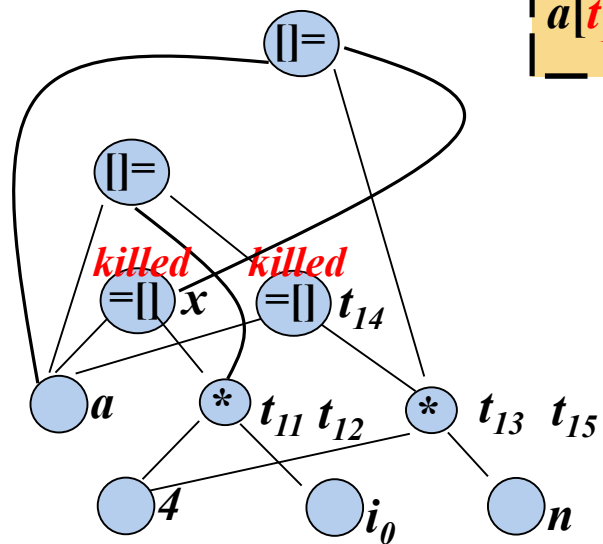


例

```

 $t_{11} = 4 * i$ 
 $x = a[t_{11}]$ 
 $t_{13} = 4 * n$ 
 $t_{14} = a[t_{13}]$ 
 $a[t_{11}] = t_{14}$ 
 $a[t_{13}] = x$ 

```



B_6

```

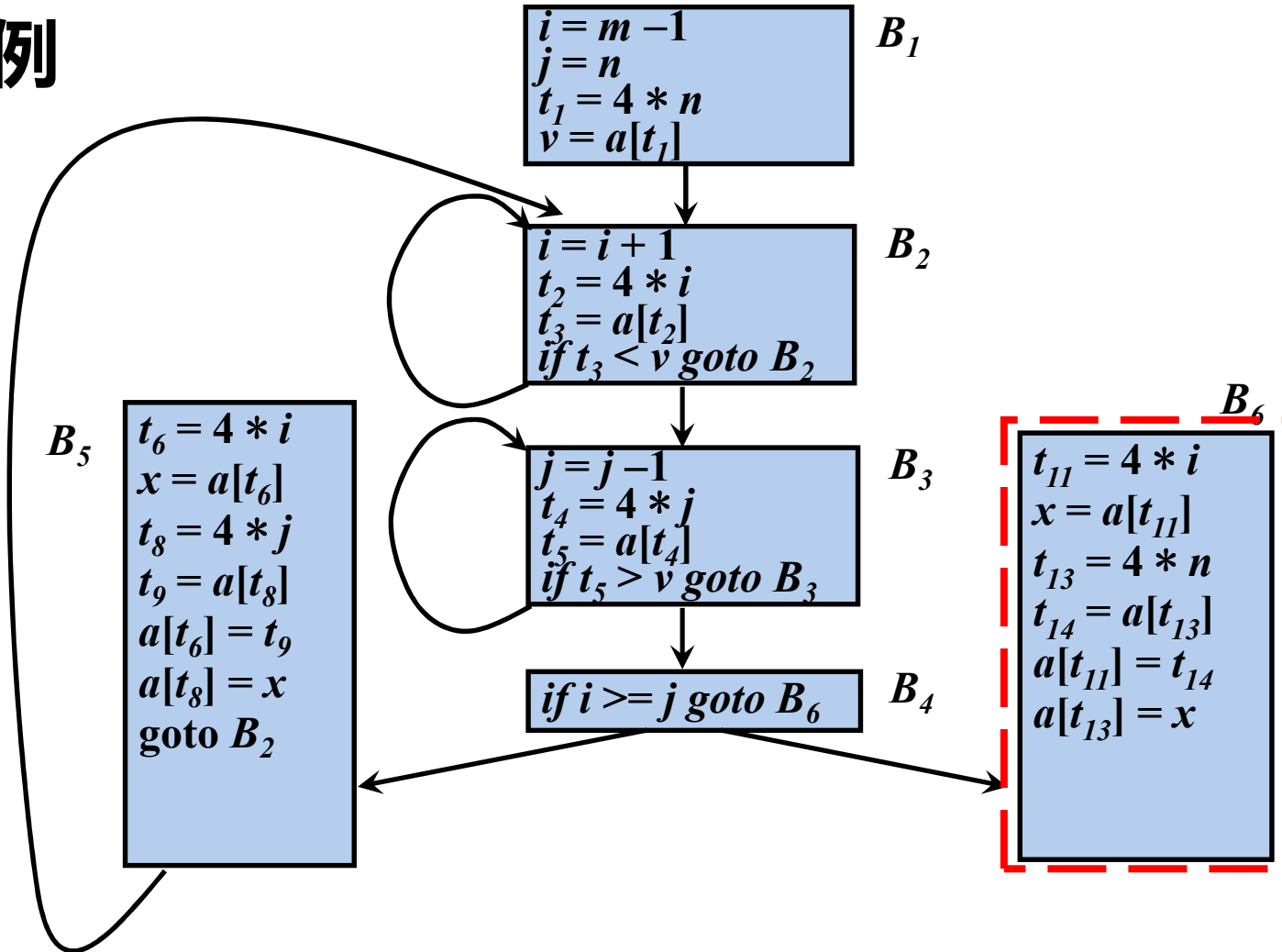
 $t_{11} = 4 * i$ 
 $x = a[t_{11}]$ 
 $t_{12} = 4 * i$ 
 $t_{13} = 4 * n$ 
 $t_{14} = a[t_{13}]$ 
 $a[t_{12}] = t_{14}$ 
 $t_{15} = 4 * n$ 
 $a[t_{15}] = x$ 

```

局部公共
子表达式



例



常用的优化方法

➤ 删除公共子表达式

删除局部公共子表达式



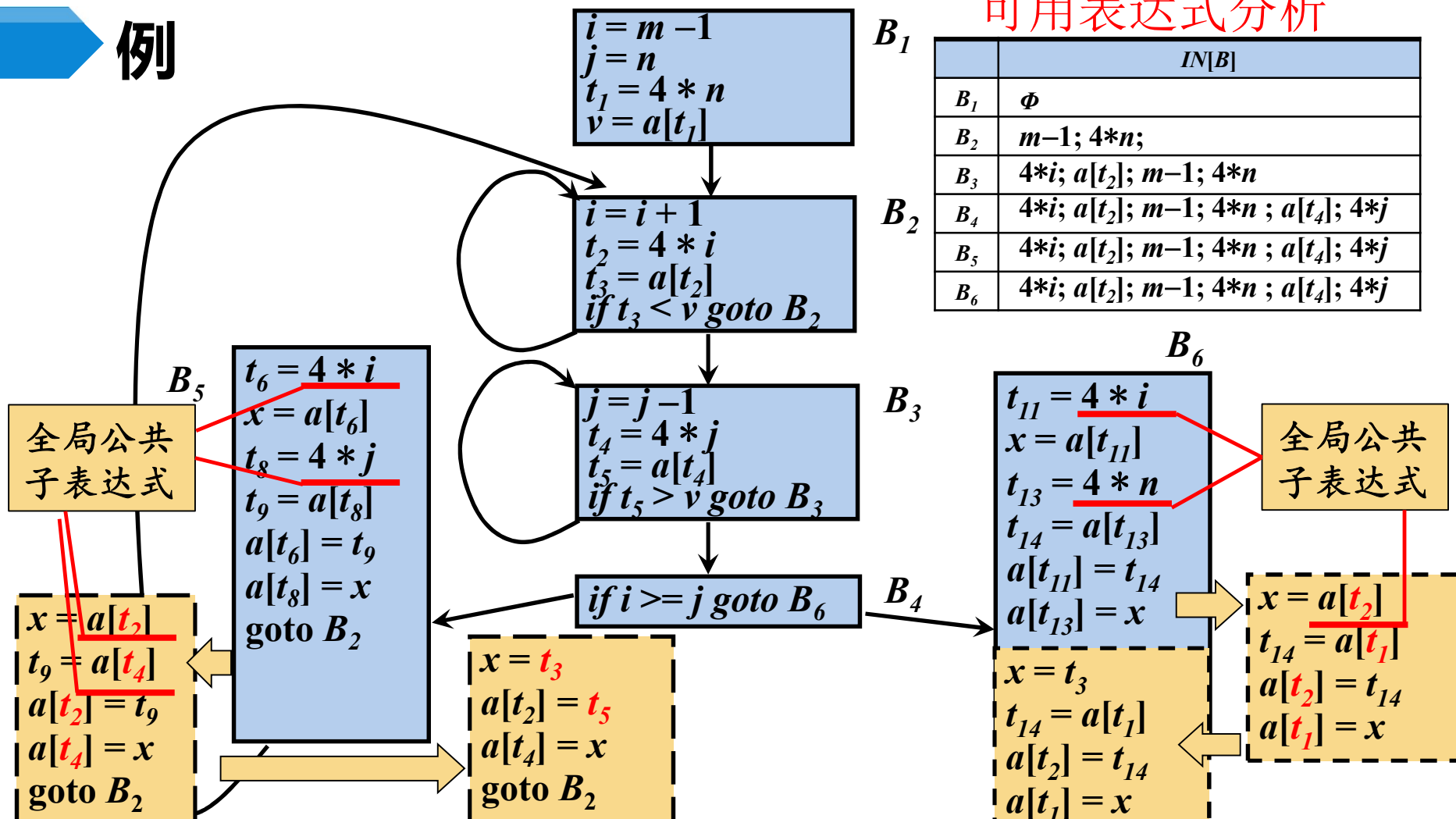
删除全局公共子表达式

DAG

可用表达式分析

例

可用表达式分析



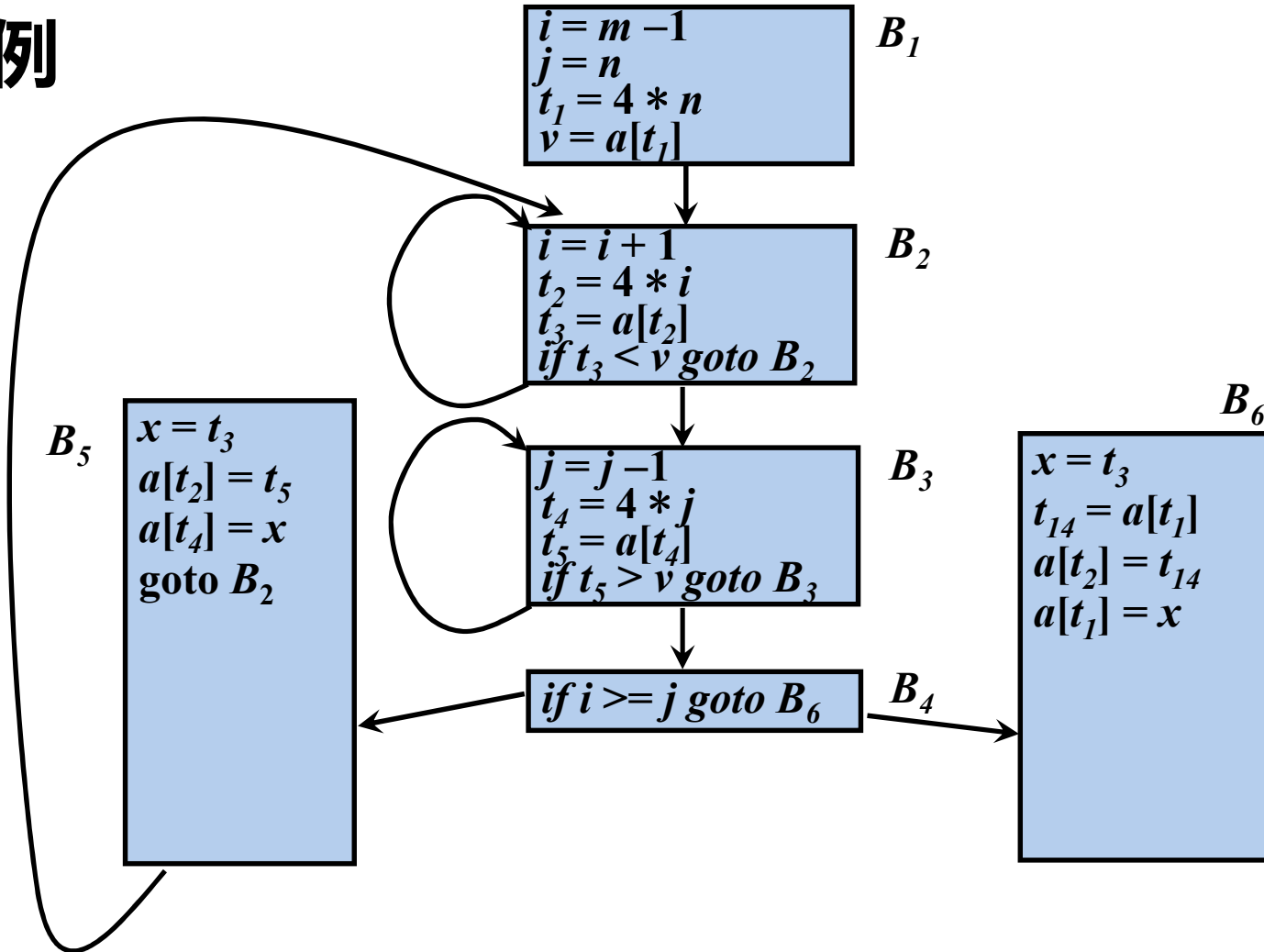
例

可用表达式分析

	$IN[B]^1$	$OUT[B]^1$
B_1	Φ	$m-1; 4*n; a[t_1]$
B_2	$m-1; 4*n; a[t_1]$	$4*i; a[t_2]; m-1; 4*n; a[t_1]$
B_3	$4*i; a[t_2]; m-1; 4*n; a[t_1]$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n; a[t_1]$
B_4	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n; a[t_1]$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n; a[t_1]$
B_5	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n; a[t_1]$	$4*j; 4*i; m-1; 4*n$
B_6	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n; a[t_1]$	$4*j; 4*i; m-1; 4*n$
	$IN[B]^2$	$OUT[B]^2$
B_1	Φ	$m-1; 4*n; a[t_1]$
B_2	$m-1; 4*n;$	$4*i; a[t_2]; m-1; 4*n$
B_3	$4*i; a[t_2]; m-1; 4*n$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$
B_4	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$
B_5	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; 4*i; m-1; 4*n$
B_6	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; 4*i; m-1; 4*n$
	$IN[B]^3$	$OUT[B]^3$
B_1	Φ	$m-1; 4*n; a[t_1]$
B_2	$m-1; 4*n;$	$4*i; a[t_2]; m-1; 4*n$
B_3	$4*i; a[t_2]; m-1; 4*n$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$
B_4	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$
B_5	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; 4*i; m-1; 4*n$
B_6	$4*j; a[t_4]; 4*i; a[t_2]; m-1; 4*n$	$4*j; 4*i; m-1; 4*n$



例



常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

➤ 计算结果永远不会被使用的语句称为**无用代码**(死代码 Dead-Code)

删除局部公共子表达式

DAG

删除全局公共子表达式

可用表达式分析

常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

删除局部公共子表达式

DAG

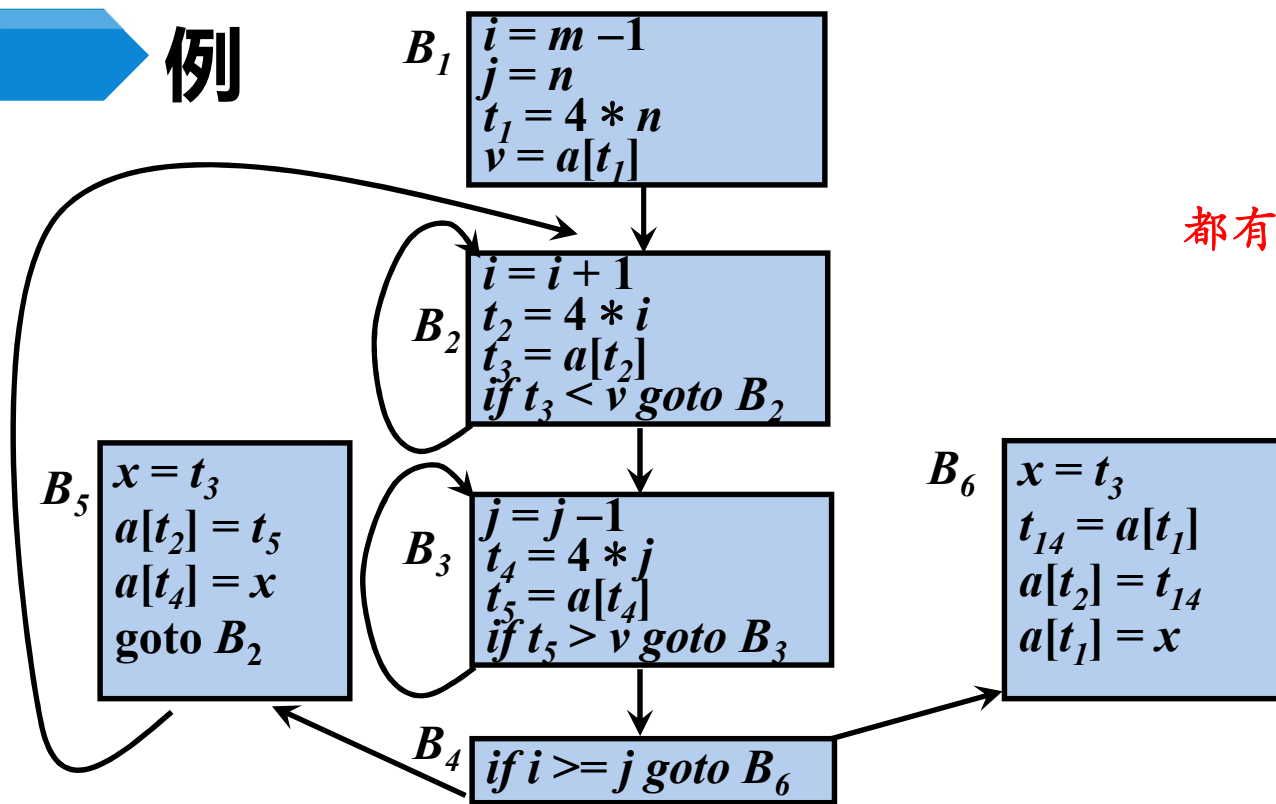
删除全局公共子表达式

可用表达式分析

删除无用代码

活跃变量分析

例



都有用?

活跃变量分析(du链)

	<i>OUT[B]</i>
<i>B</i> ₆	Φ
<i>B</i> ₅	<i>i</i> ; <i>j</i> ; <i>v</i> ; <i>t</i> ₁ ;
<i>B</i> ₄	<i>i</i> ; <i>j</i> ; <i>v</i> ; <i>t</i> ₁ ; <i>t</i> ₃ ; <i>t</i> ₅ ;
<i>B</i> ₃	<i>i</i> ; <i>j</i> ; <i>v</i> ; <i>t</i> ₁ ; <i>t</i> ₃ ; <i>t</i> ₅
<i>B</i> ₂	<i>i</i> ; <i>j</i> ; <i>v</i> ; <i>t</i> ₁ ; <i>t</i> ₃ ;
<i>B</i> ₁	<i>i</i> ; <i>j</i> ; <i>v</i> ; <i>t</i> ₁ ;

常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

➤ 删除无用复制语句

➤ 对于复制语句 $s: x=y$, 如果在 x 的所有引用点都可以用对 y 的引用代替对 x 的引用(复制传播), 那么可以删除复制语句 $x=y$

删除局部公共子表达式

DAG

删除全局公共子表达式

可用表达式分析

删除无用代码

活跃变量分析

常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

➤ 删除无用复制语句

删除局部公共子表达式

DAG

删除全局公共子表达式

可用表达式分析

删除无用代码

活跃变量分析

复制传播

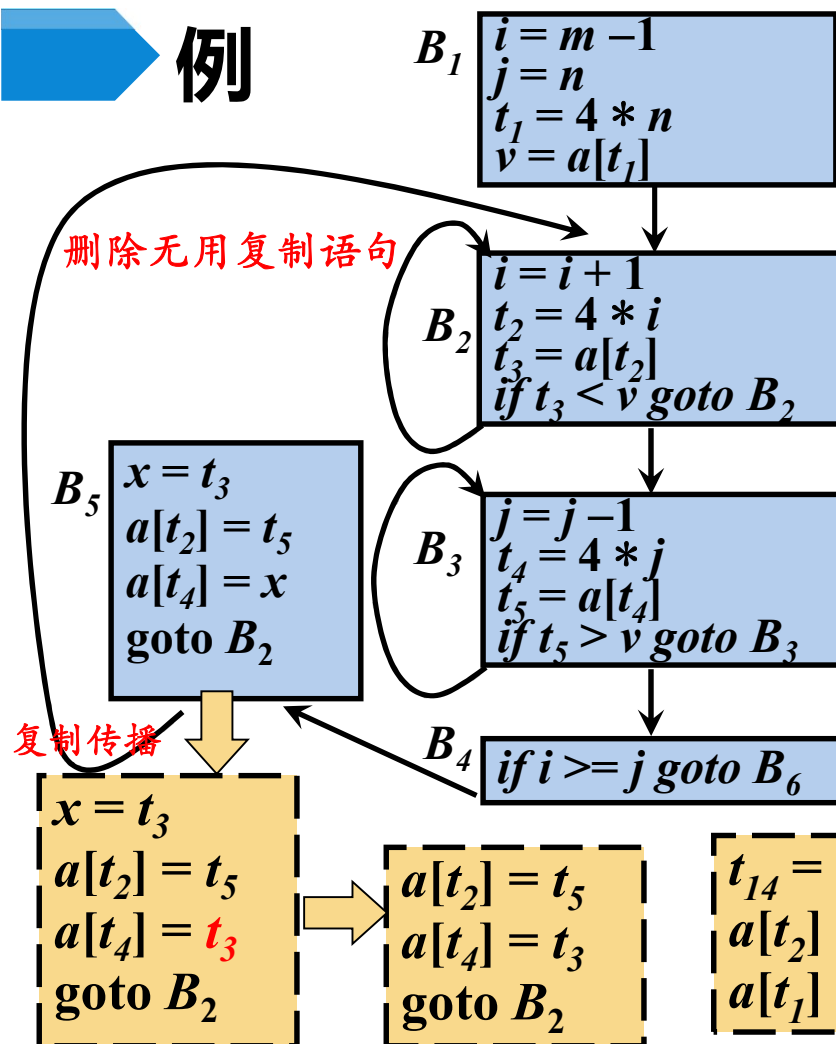
活跃变量分析(du链)

可用表达式(复制语句)分析

删除无用复制语句

活跃变量分析

例

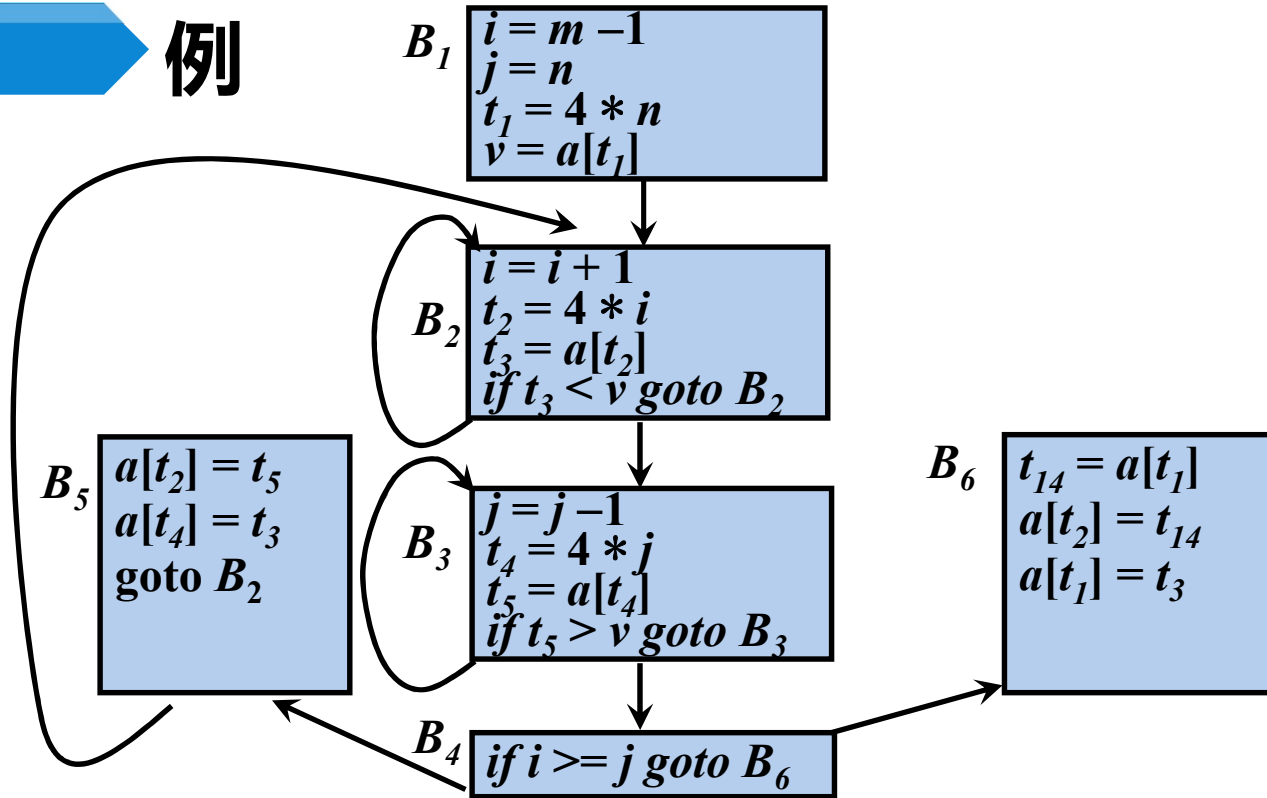


可用表达式（复制语句）分析

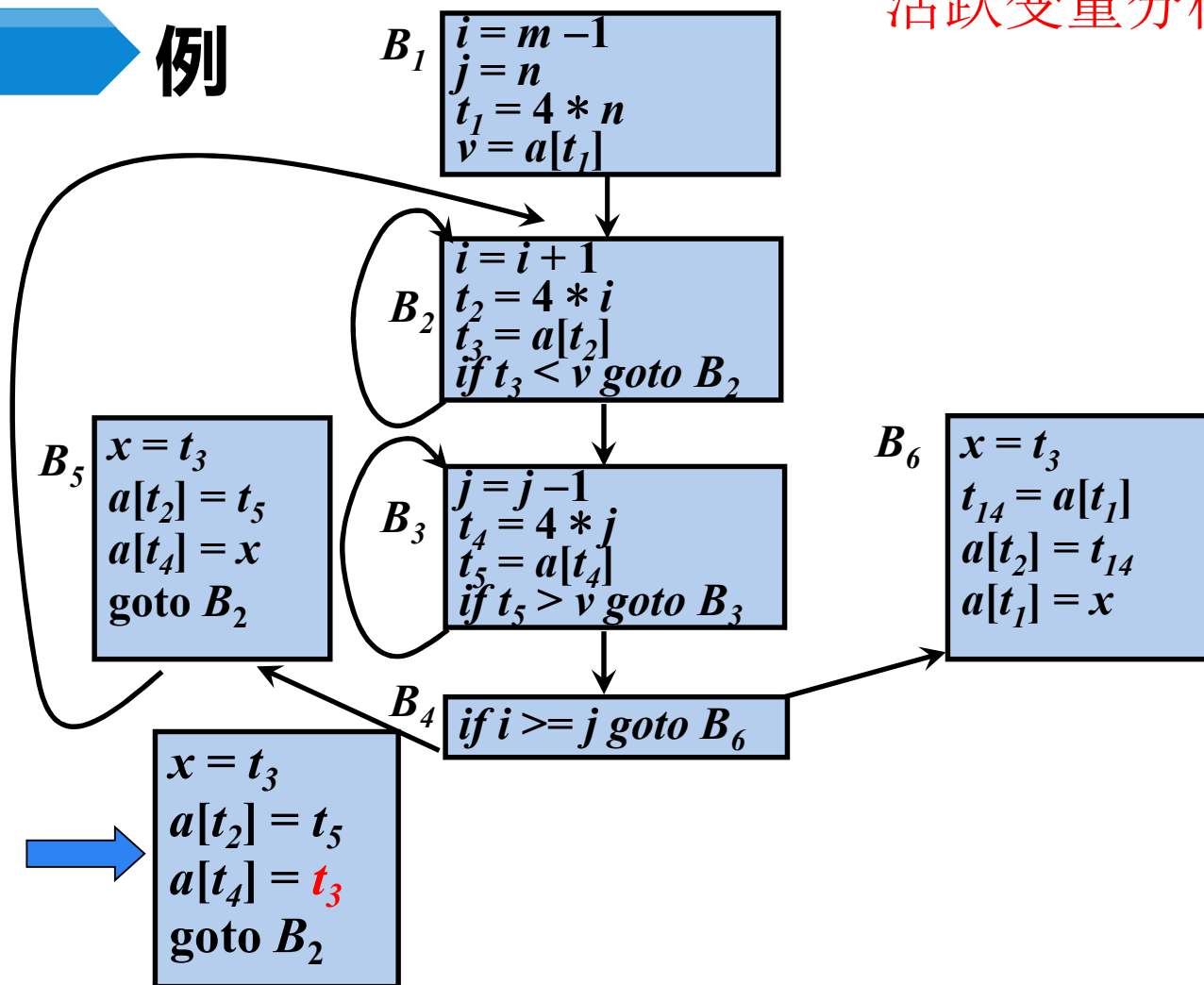
	OUT[B] ⁰	IN[B] ¹	OUT[B] ¹	IN[B] ²	OUT[B] ²	IN[B] ³	OUT[B] ³
B ₁	U	Φ	j=n	Φ	j=n	Φ	j=n
B ₂	U	j=n	j=n	Φ	Φ	Φ	Φ
B ₃	U	j=n	Φ	Φ	Φ	Φ	Φ
B ₄	U	Φ	Φ	Φ	Φ	Φ	Φ
B ₅	U	Φ	x = t ₃	Φ	x = t ₃	Φ	x = t ₃
B ₆	U	Φ	x = t ₃	Φ	x = t ₃	Φ	x = t ₃

活跃变量分析(du链)

	OUT[B]
B ₆	Φ
B ₅	i; j; v; t ₁ ;
B ₄	i; j; v; t ₁ ; t ₃ ; t ₅ ;
B ₃	i; j; v; t ₁ ; t ₃ ; t ₅ ;
B ₂	i; j; v; t ₁ ; t ₃ ;
B ₁	i; j; v; t ₁ ;



例

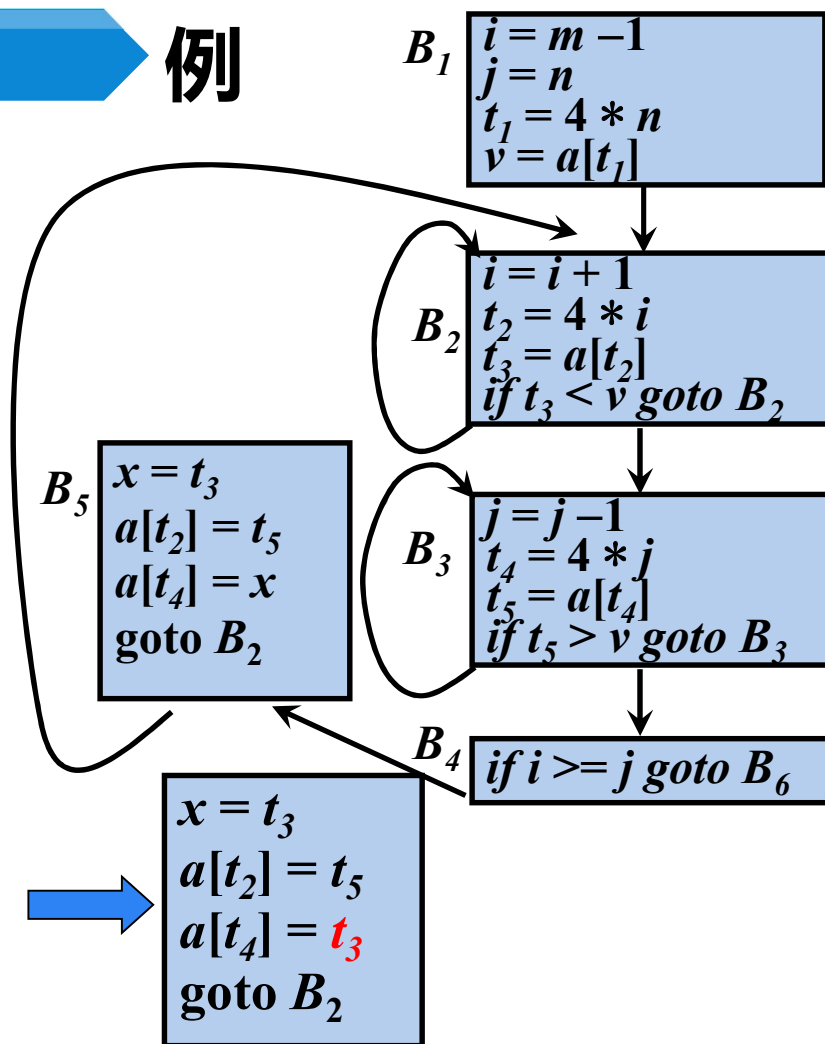


活跃变量分析

	$IN[B]^0$	$OUT[B]^1$	$IN[B]^1$
B_6	Φ	Φ	$t_3; t_1;$
B_5	Φ	Φ	$t_3; t_5;$
B_4	Φ	$t_1; t_3; t_5;$	$t_1; t_3; t_5;$
B_3	Φ	$t_1; t_3; t_5;$	$j; v; t_1; t_3;$
B_2	Φ	$j; v; t_1; t_3;$	$i; j; v; t_1;$
B_1	Φ	$i; j; v; t_1;$	$m; n;$

	$OUT[B]^2$	$IN[B]^2$
B_6	Φ	$t_3; t_1;$
B_5	$i; j; v; t_1;$	$i; j; v; t_1; t_3; t_5;$
B_4	$i; j; v; t_1; t_3; t_5;$	$i; j; v; t_1; t_3; t_5;$
B_3	$i; j; v; t_1; t_3; t_5;$	$i; j; v; t_1; t_3;$
B_2	$i; j; v; t_1; t_3;$	$i; j; v; t_1;$
B_1	$i; j; v; t_1;$	$m; n;$

例



可用复制语句（表达式）分析

	$OUT[B]^0$	$IN[B]^1$	$OUT[B]^1$	$IN[B]^2$	$OUT[B]^2$	$IN[B]^3$	$OUT[B]^3$
B_1	U	Φ	$j=n$	Φ	$j=n$	Φ	$j=n$
B_2	U	$j=n$	$j=n$	Φ	Φ	Φ	Φ
B_3	U	$j=n$	Φ	Φ	Φ	Φ	Φ
B_4	U	Φ	Φ	Φ	Φ	Φ	Φ
B_5	U	Φ	$x = t_3$	Φ	$x = t_3$	Φ	$x = t_3$
B_6	U	Φ	$x = t_3$	Φ	$x = t_3$	Φ	$x = t_3$

常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

➤ 常量合并

➤ 如果在编译时刻推导出一个表达式的值是常量，就可以使用该常量来替代这个表达式。该技术被称为**常量合并**



常用的优化方法

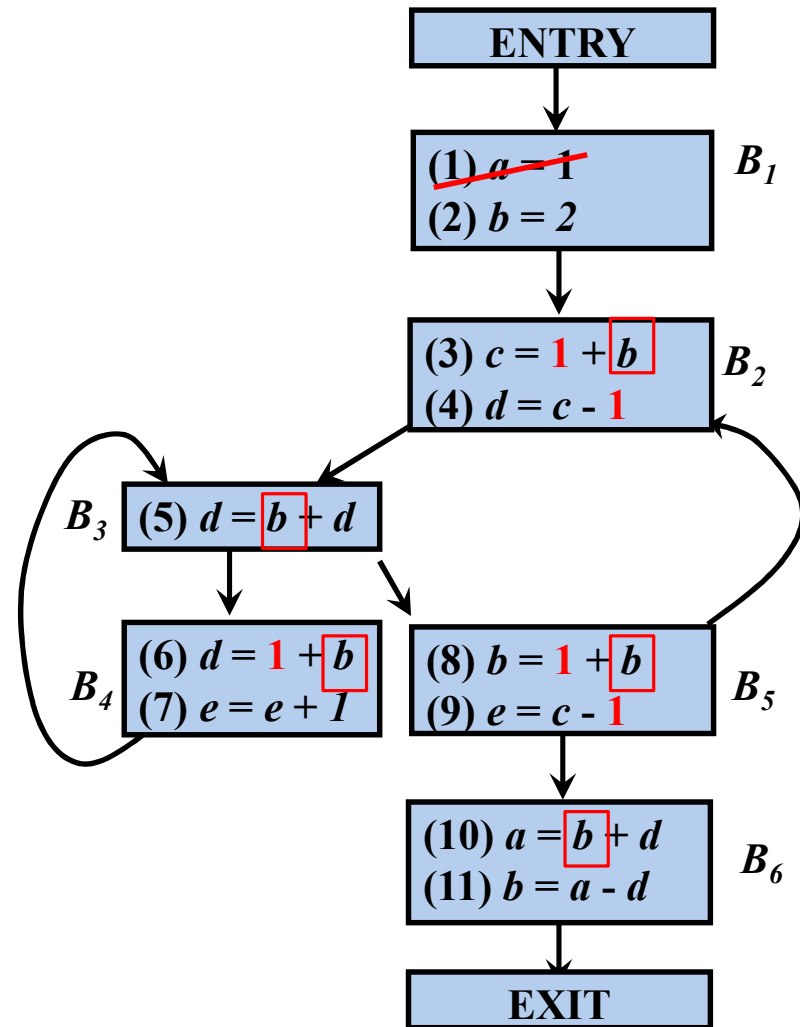
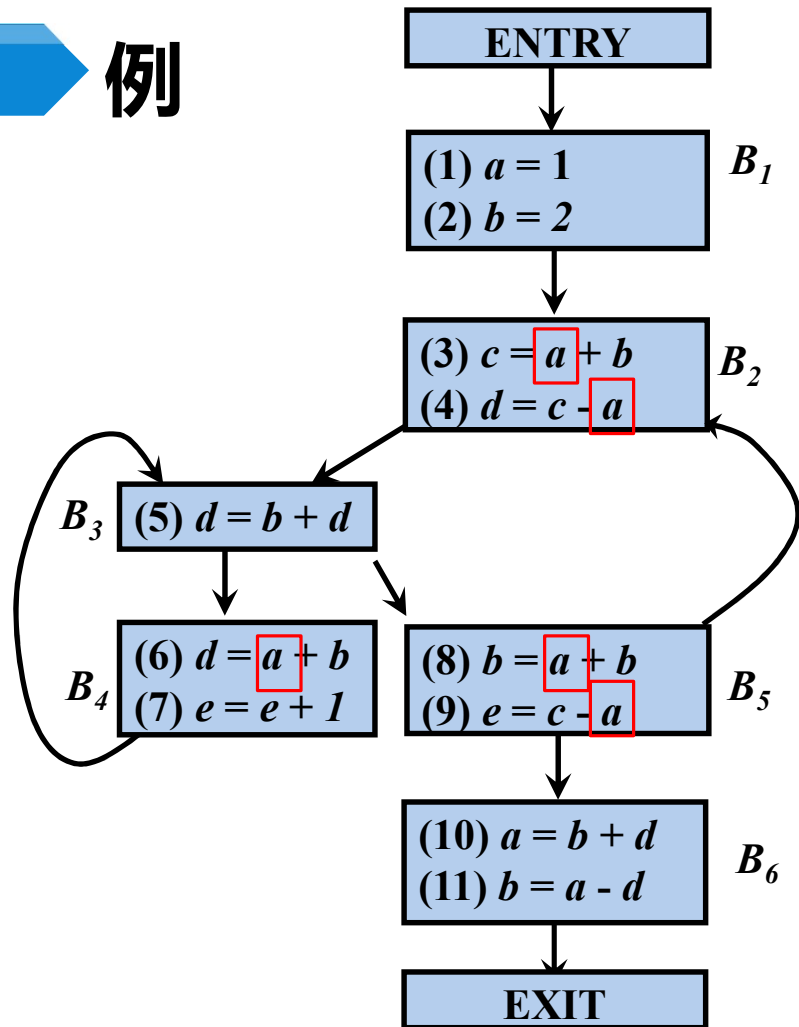
➤ 删除公共子表达式

➤ 删除无用代码

➤ 常量合并

➤ 如果在编译时刻推导出一个表达式的值是常量，就可以使用该常量来替代这个表达式。该技术被称为**常量合并**





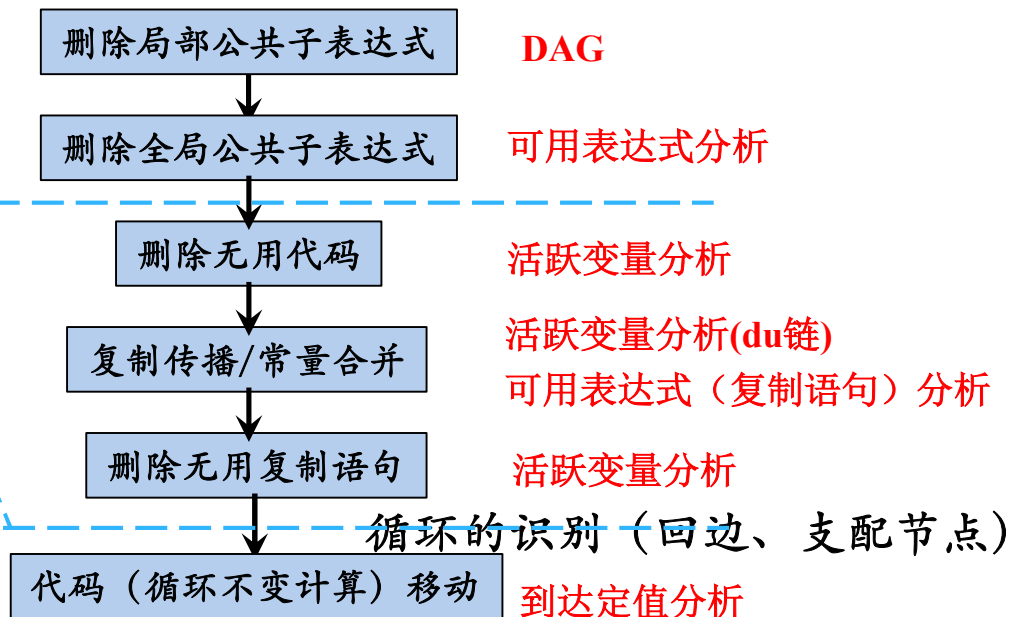
常用的优化方法

➤ 删除公共子表达式

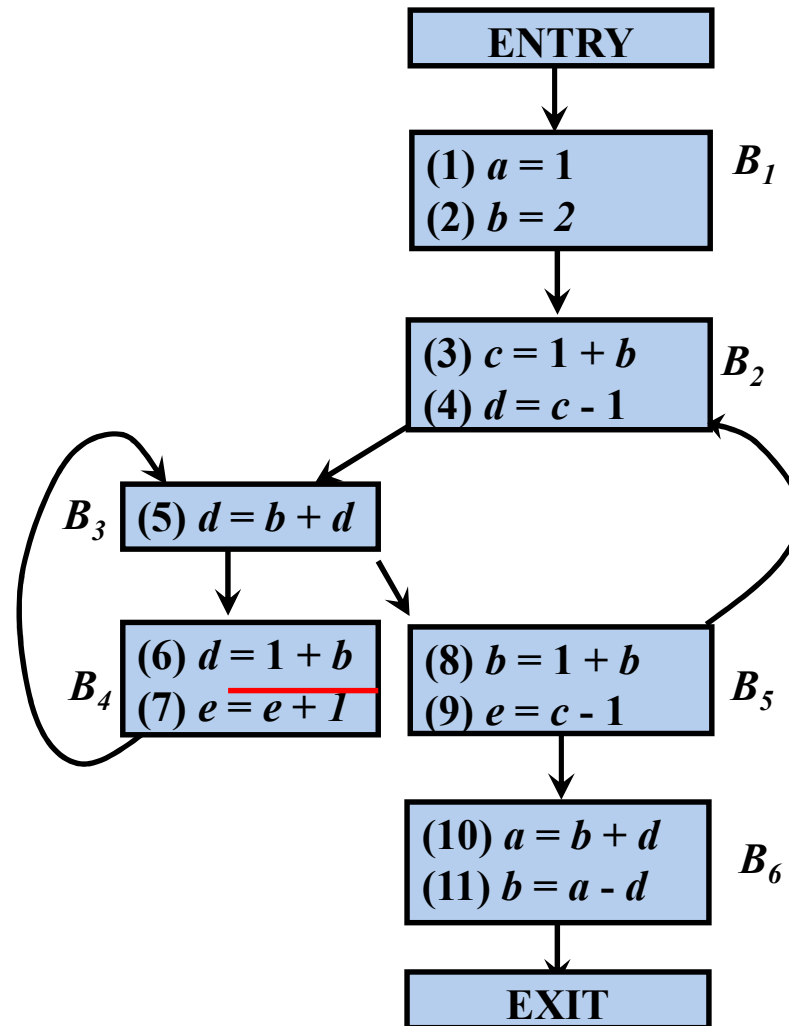
➤ 删除无用代码

➤ 常量合并

➤ 代码移动



例



常用的优化方法

➤ 删除公共子表达式

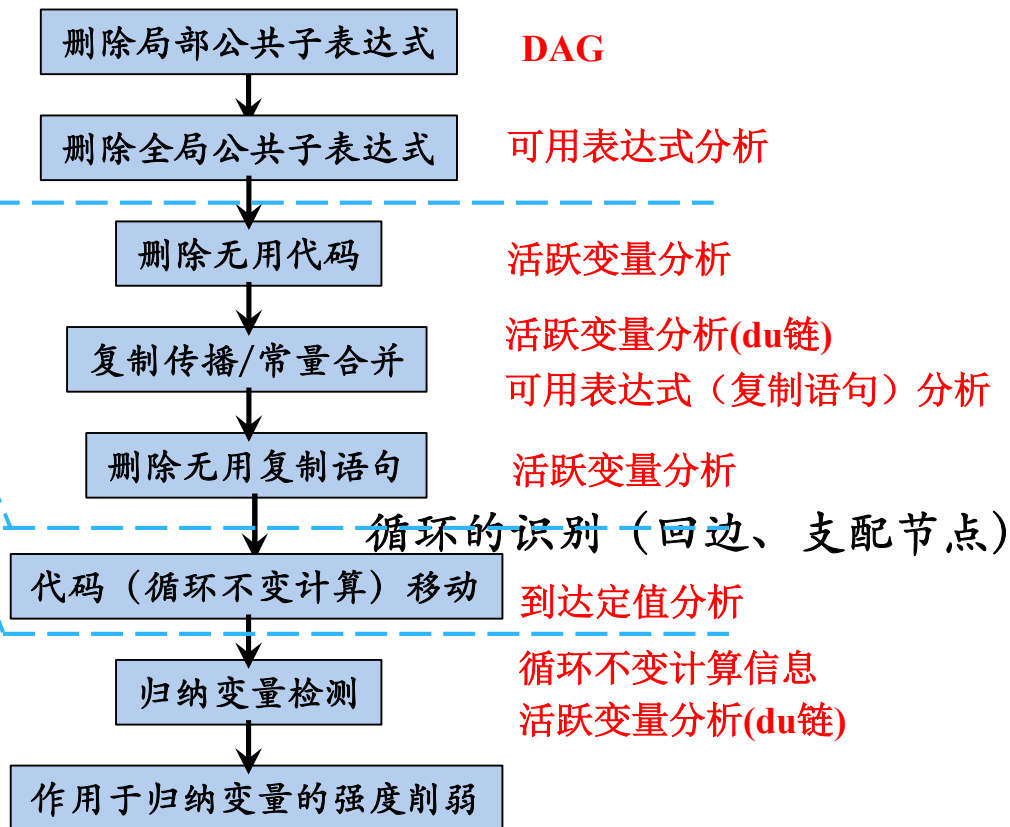
➤ 删除无用代码

➤ 常量合并

➤ 代码移动

➤ 强度削弱

➤ 用较快的操作代替较慢的操作，如用加代替乘



例：归纳变量的检测

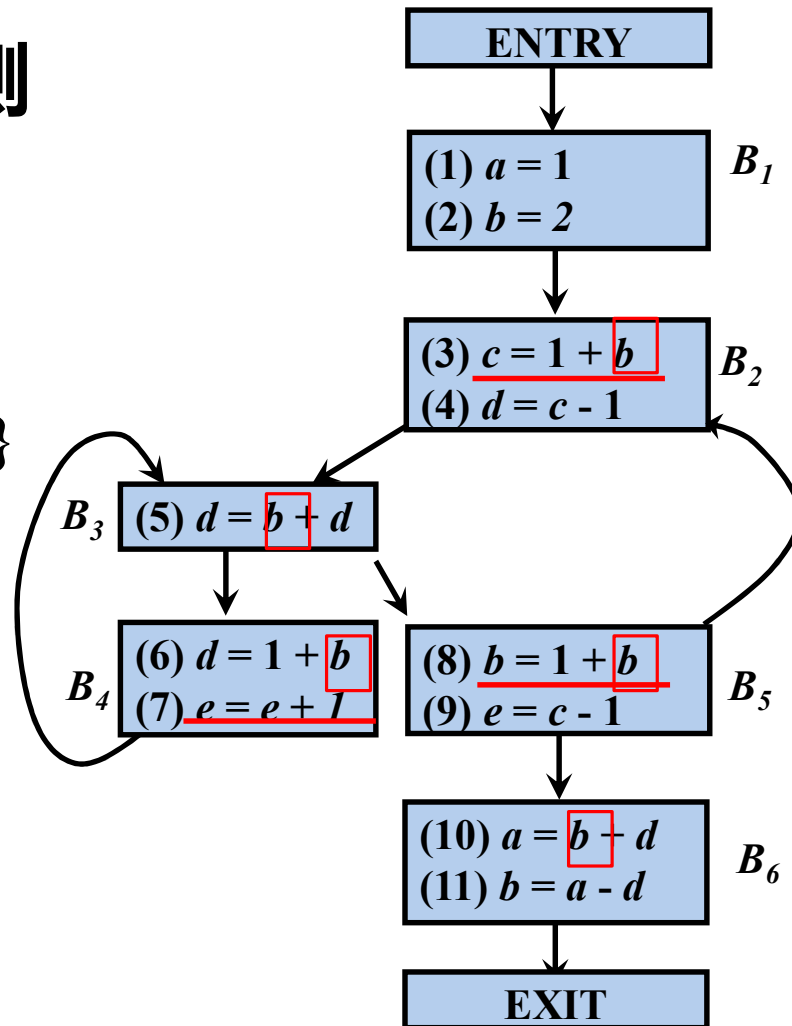
➤ 循环 $\{B_3, B_4\}$

➤ e

➤ 循环 $\{B_2, B_3, B_4, B_5\}$

➤ b

➤ c



例：归纳变量的检测

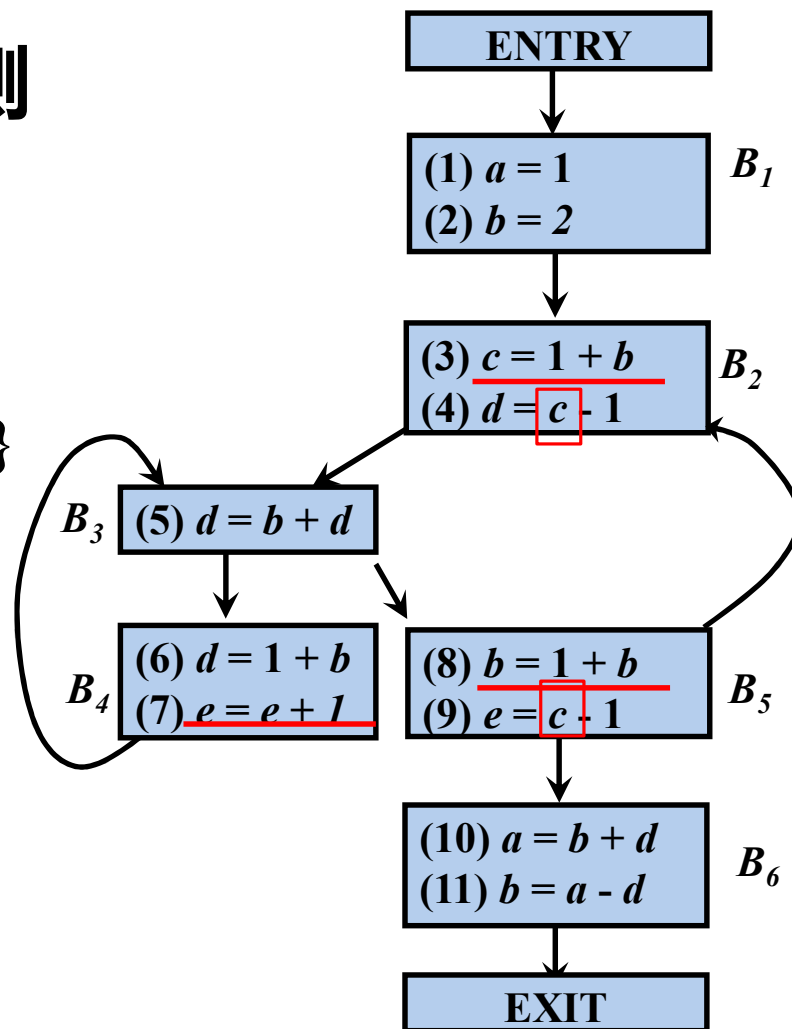
➤ 循环 $\{B_3, B_4\}$

➤ e

➤ 循环 $\{B_2, B_3, B_4, B_5\}$

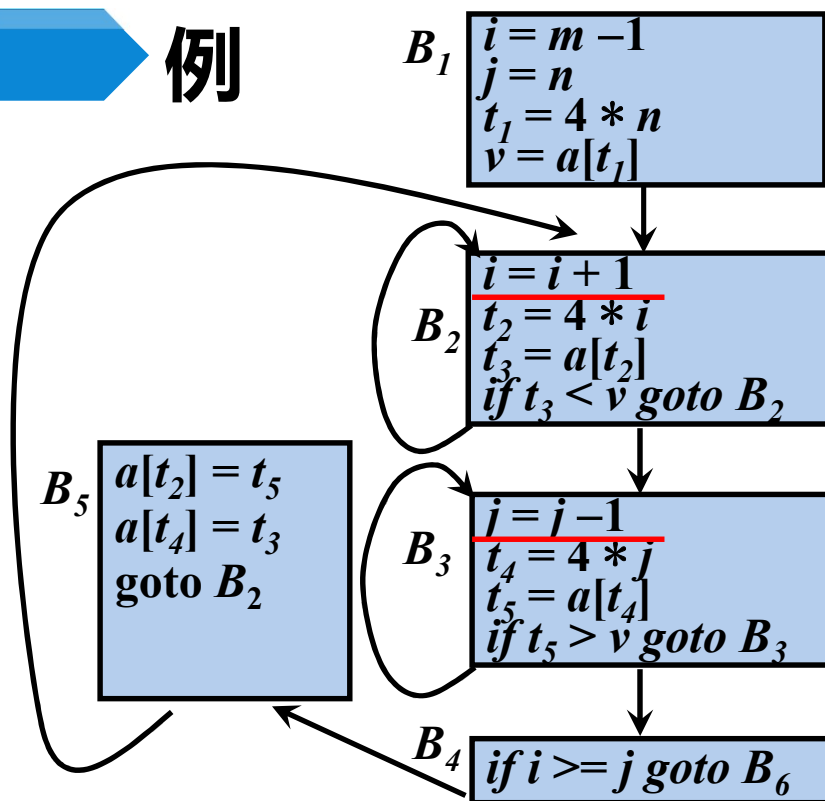
➤ b

➤ c



例

归纳变量的检测



i 是基本归纳变量

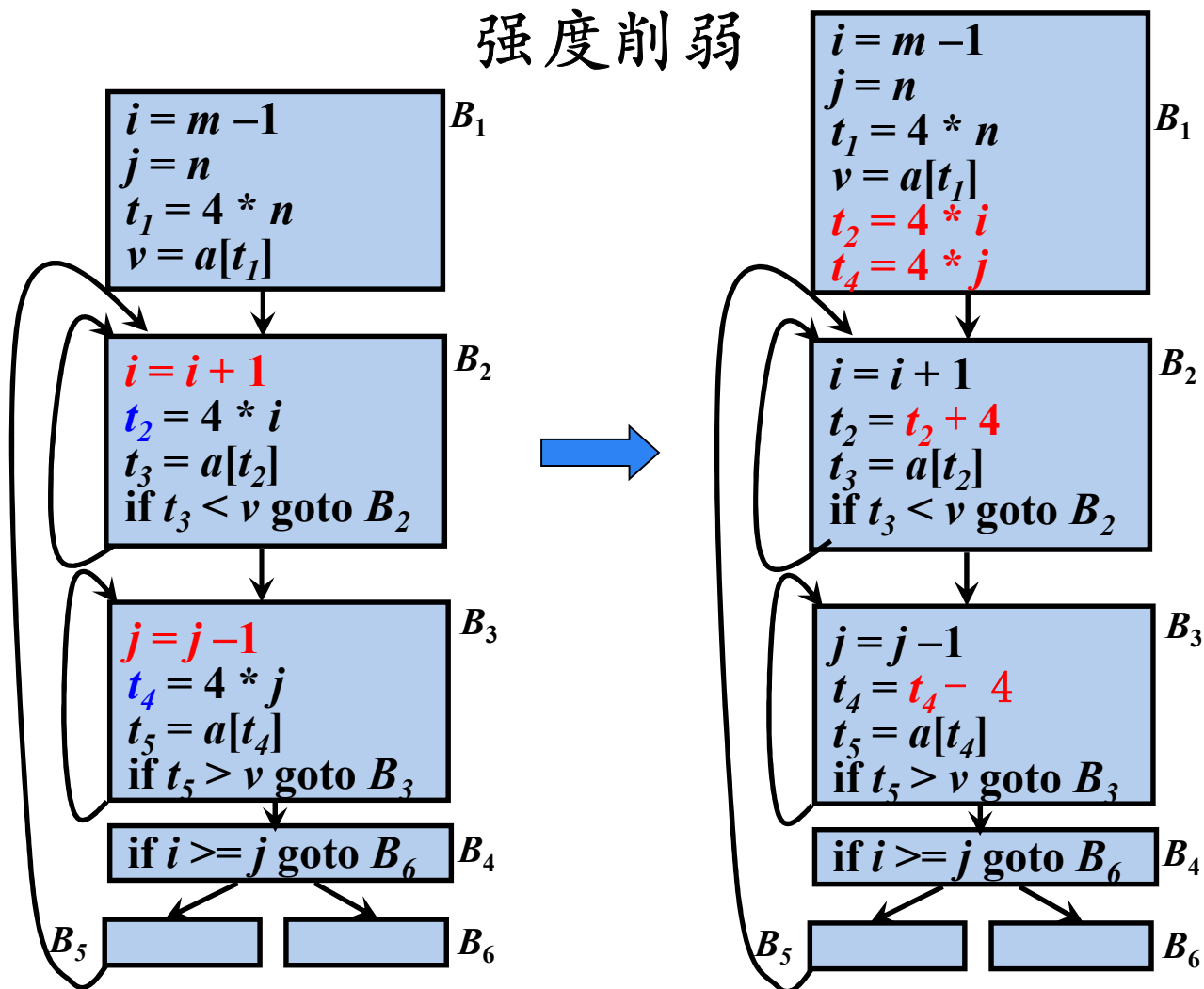
t_2 是 i 族归纳变量, 可以表示为 $(i, 4, 0)$

j 是基本归纳变量

t_4 是 j 族归纳变量, 可以表示为 $(j, 4, 0)$



强度削弱



常用的优化方法

➤ 删除公共子表达式

➤ 删除无用代码

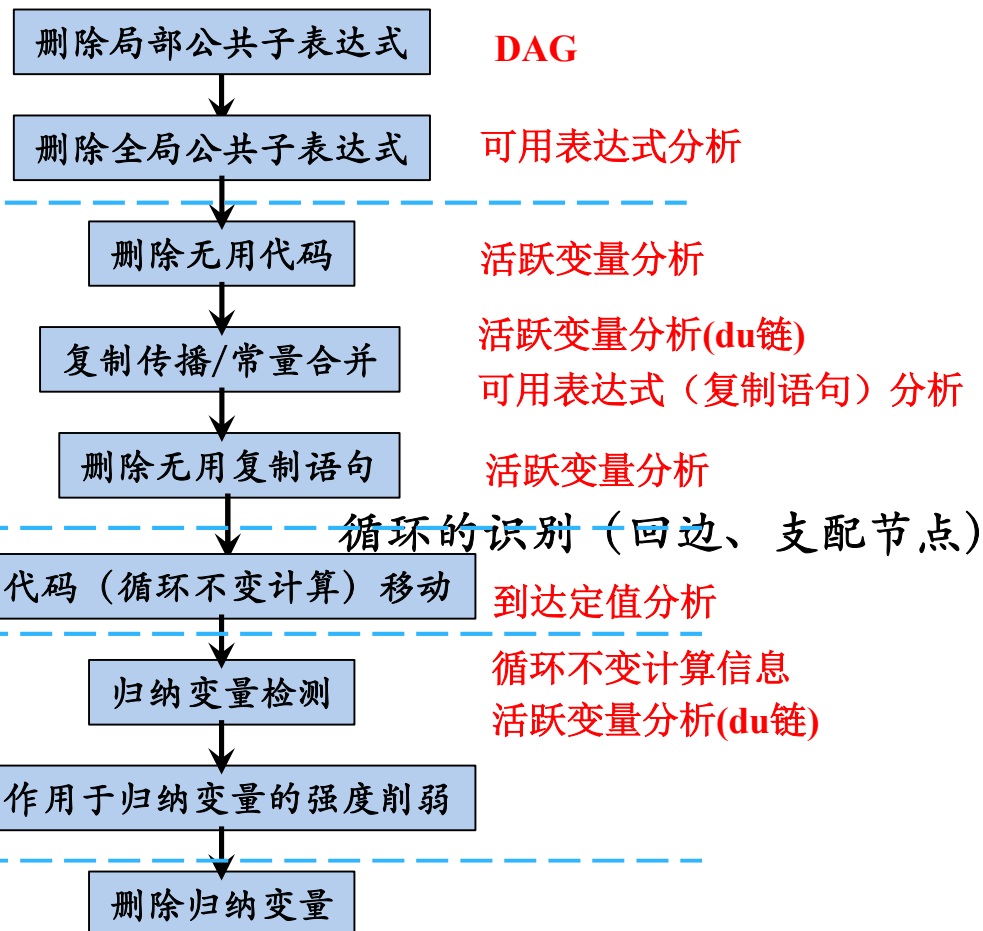
➤ 常量合并

➤ 代码移动

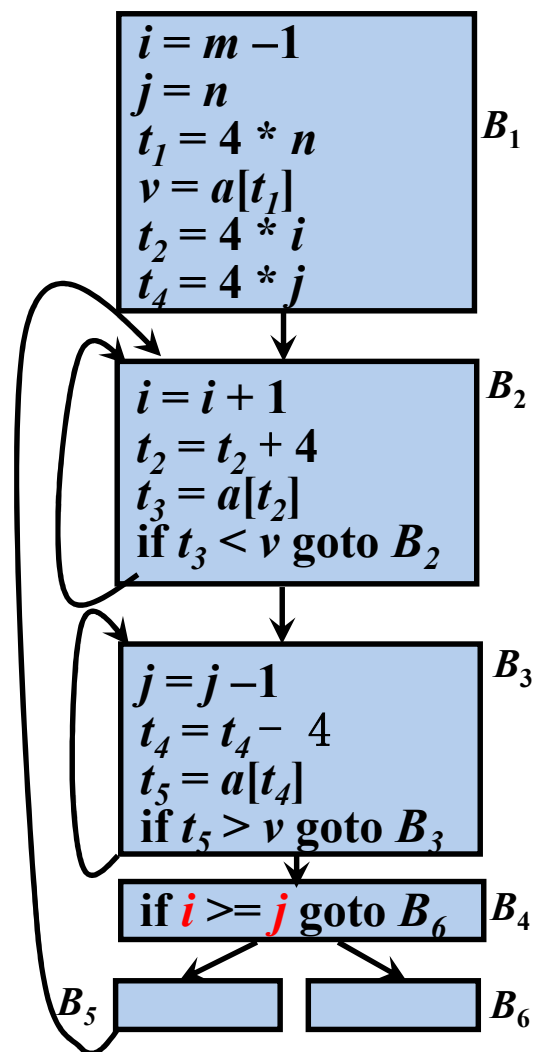
➤ 强度削弱

➤ 删除归纳变量

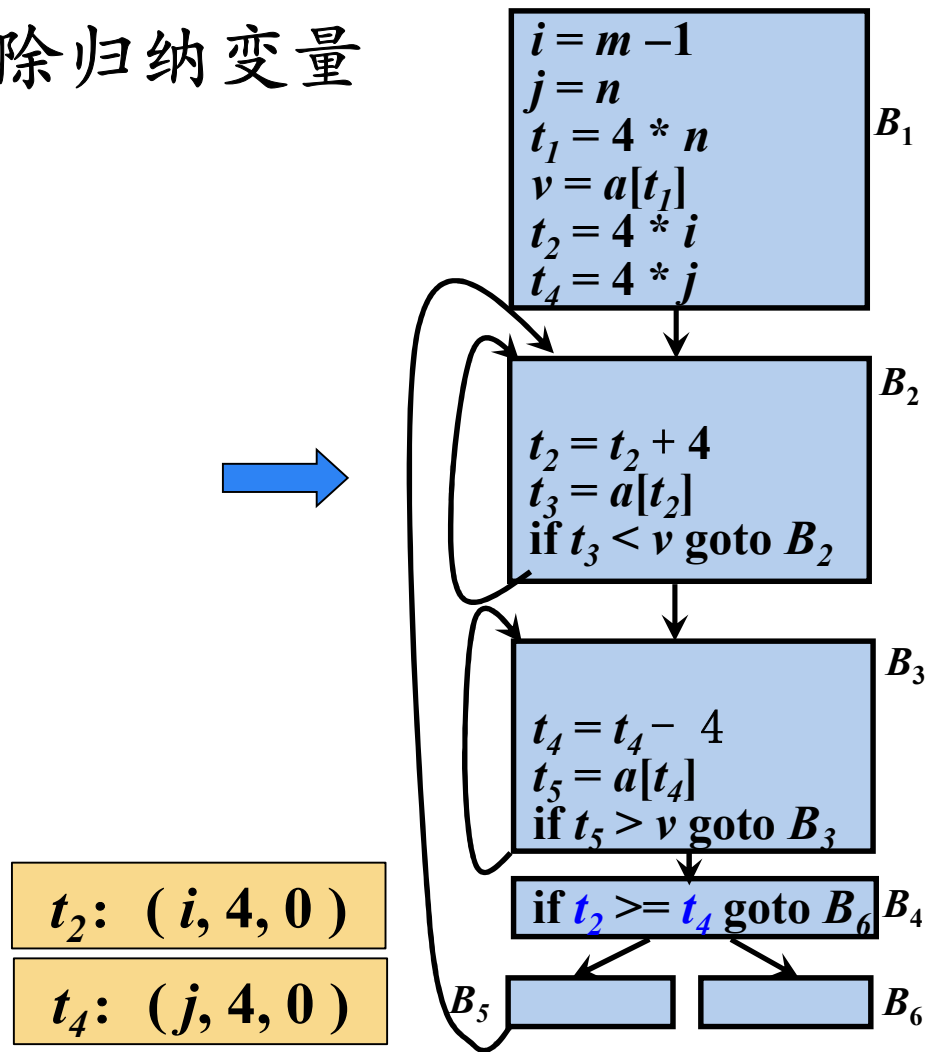
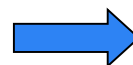
➤ 强度削弱后，有些归纳变量的作用只是用于测试



例



删除归纳变量



回边 (*Back Edges*)

➤ 如果存在从结点 n 到 d 的有向边 $n \rightarrow d$, 且 $d \text{ dom } n$, 那么这条边称为回边

~~✗ ➤ 假定流图中存在两个结点 d 和 n 满足 $d \text{ dom } n$ 。如果存在从结点 n 到 d 的有向边 $n \rightarrow d$, 那么这条边称为回边~~



结束

