

A BETTER GLOBAL PROGRESSIVE REGISTER ALLOCATOR

David Ryan Koes and Seth Copen Goldstein

Carnegie Mellon University

{dkoes,seth}@cs.cmu.edu

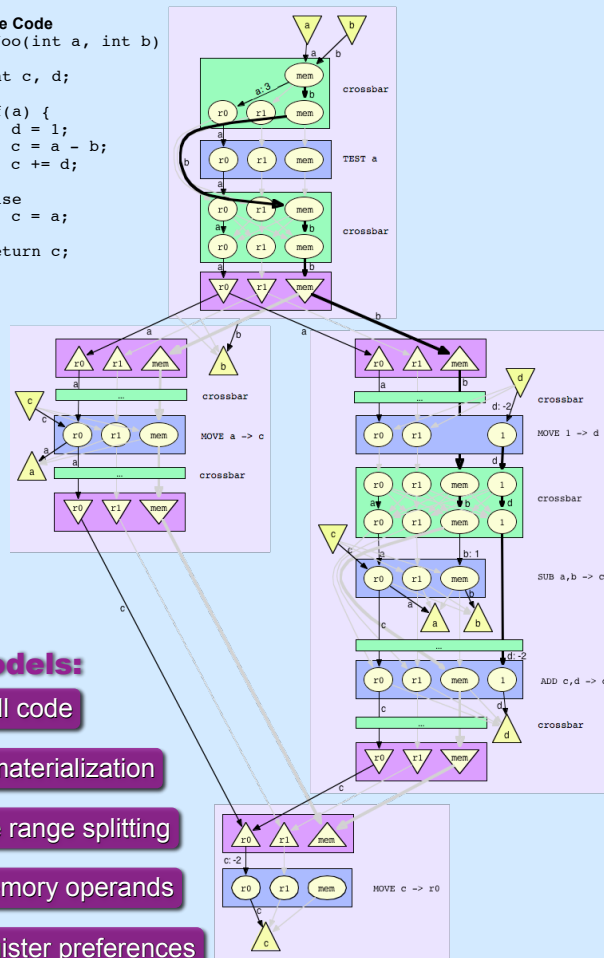
Explicit and Expressive Model: Global MCNF

Source Code

```
int foo(int a, int b)
{
    int c, d;

    if(a) {
        d = 1;
        c = a - b;
        c += d;
    }
    else
        c = a;

    return c;
}
```



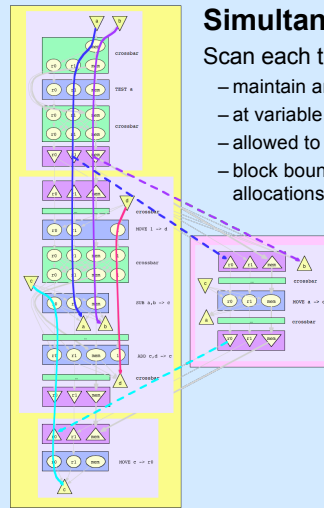
Progressive Solver: Lagrangian Directed Allocator

Quickly finds good allocation
Approaches optimal allocation over time

Simultaneous Allocator

Scan each trace

- maintain an allocation of all live variables
- at variable definition find cheapest allocation
- allowed to evict (reallocate) an allocated variable
- block boundary allocations fixes boundary allocations of other traces

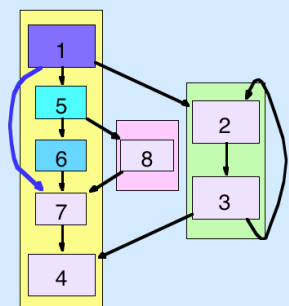


Trace Update

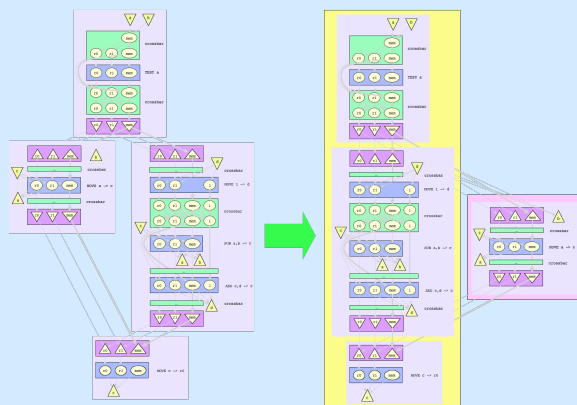
Intra-trace control flow requires shortest path information be updated during trace allocation

easy-update: minimum amount of recomputation (block 6)

full-update: all effected blocks updated (blocks 5 and 6)

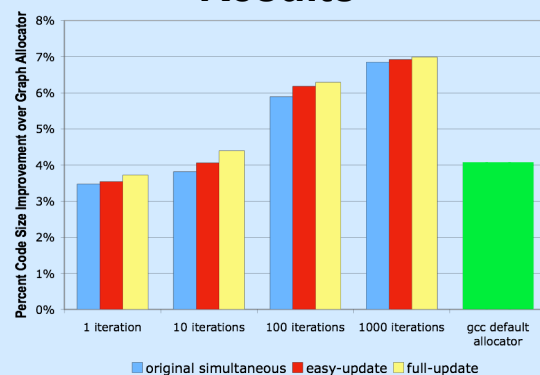


Trace Decomposition



Greedily choose longest traces
Loop headers start new trace

Results



References

D. Koes and S. Goldstein, A Global Progressive Register Allocator. In *Proceedings of the 2006 ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)*. ACM Press, June 2006.
<http://www.cs.cmu.edu/~dkoes/>