
Using Genetic Programming for Source-Level Data Assignment to Dual Memory Banks

Alastair Murray and Björn Franke
Institute for Computing Systems Architecture,
School of Informatics, University of Edinburgh

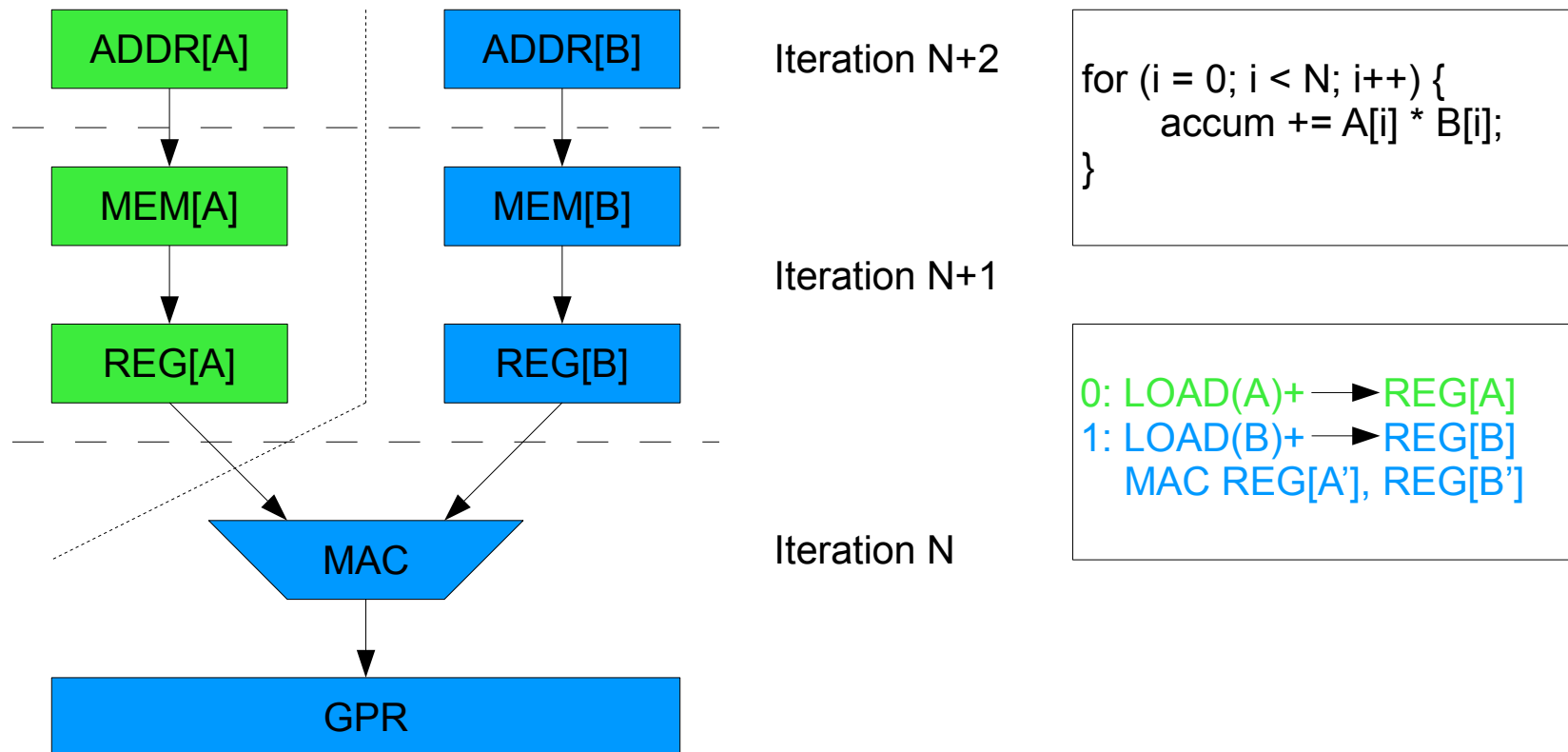
SMART: January 25th, 2009



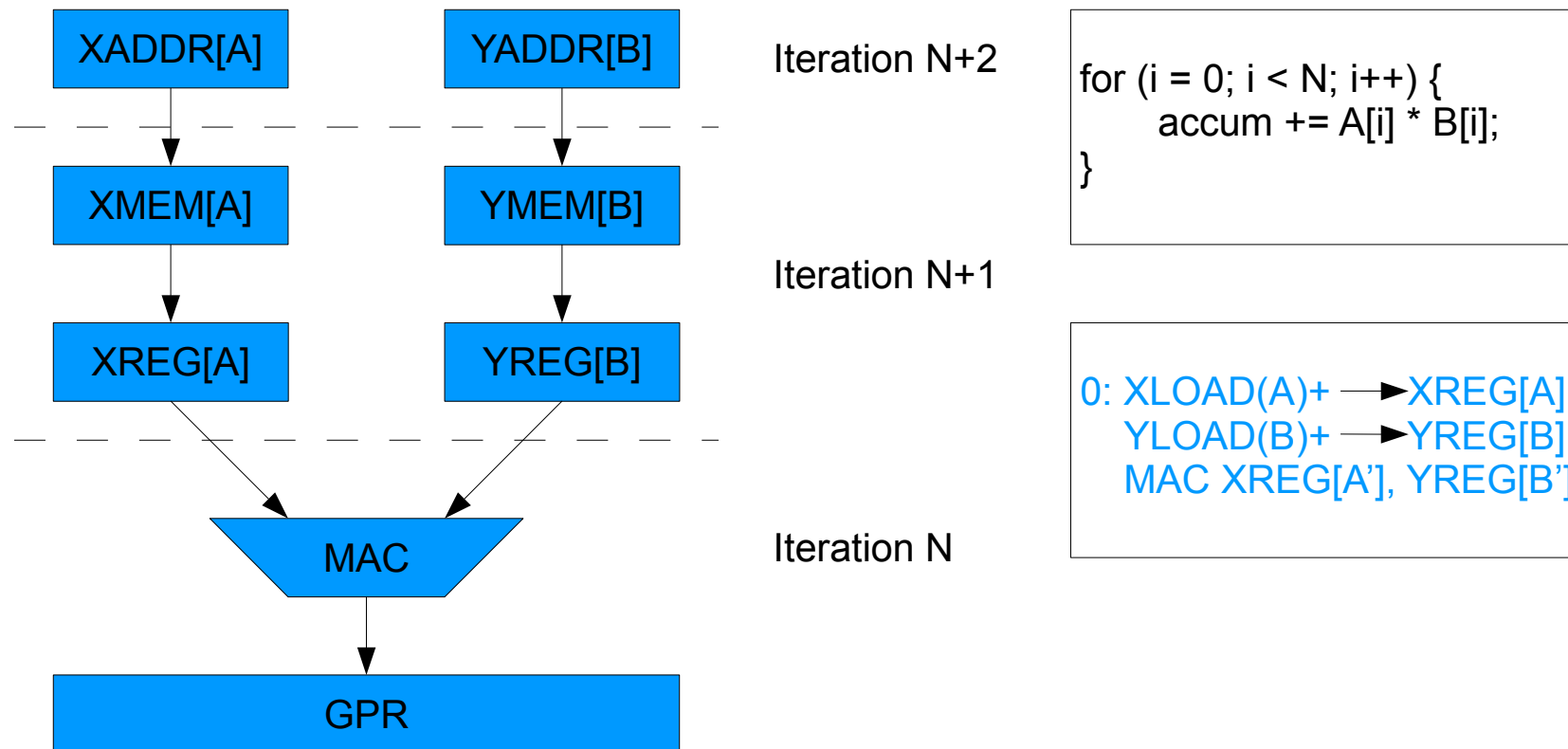
Overview

- Many DSPs have dual on-chip memories to increase bandwidth.
- Automatic methods to exploit these already exist:
 - Aren't used in Industrial DSP compilers.
 - Give little or no control to the programmer.
- Previously created a source-level solution to allow programmer interaction.
- Created some new issues for automatic assignment:
 - Lack of low-level information makes it hard to model the problem.
 - Found that genetic programming helped to overcome this.

Dual Memory Bank Architectures



Dual Memory Bank Architectures



Manual Assignment

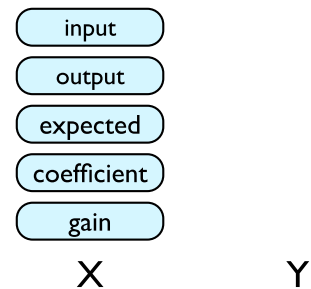
- Assign variables to memory banks via C language extensions.
 - DSP-C
 - Embedded C
- Supported by many DSP compilers.
- Syntax examples:

```
float X data[32];  
int X * Y pointer;
```

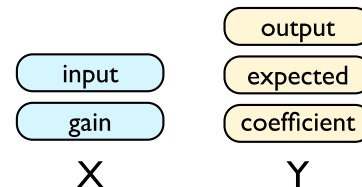
Automatic DSP-C Assignment

- Previous approaches mostly low-level.
- Our approach works at the source-level:
 - Create a C-to-DSP-C source-level transformation tool.
 - Have this tool perform full or partial assignment.
 - Portable to any DSP-C compiler.
- Allows programmers to place some variables manually.
- Leading existing technique – Integer Linear Programming:
 - Good results but issues operating at a high-level.

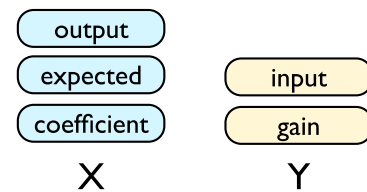
Example Assignment (UTDSP Imsfir)



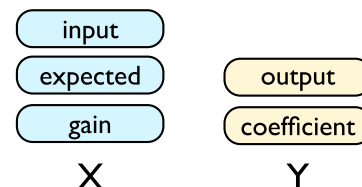
(a) All data in X memory



(b) Bank assignment resulting in best performance

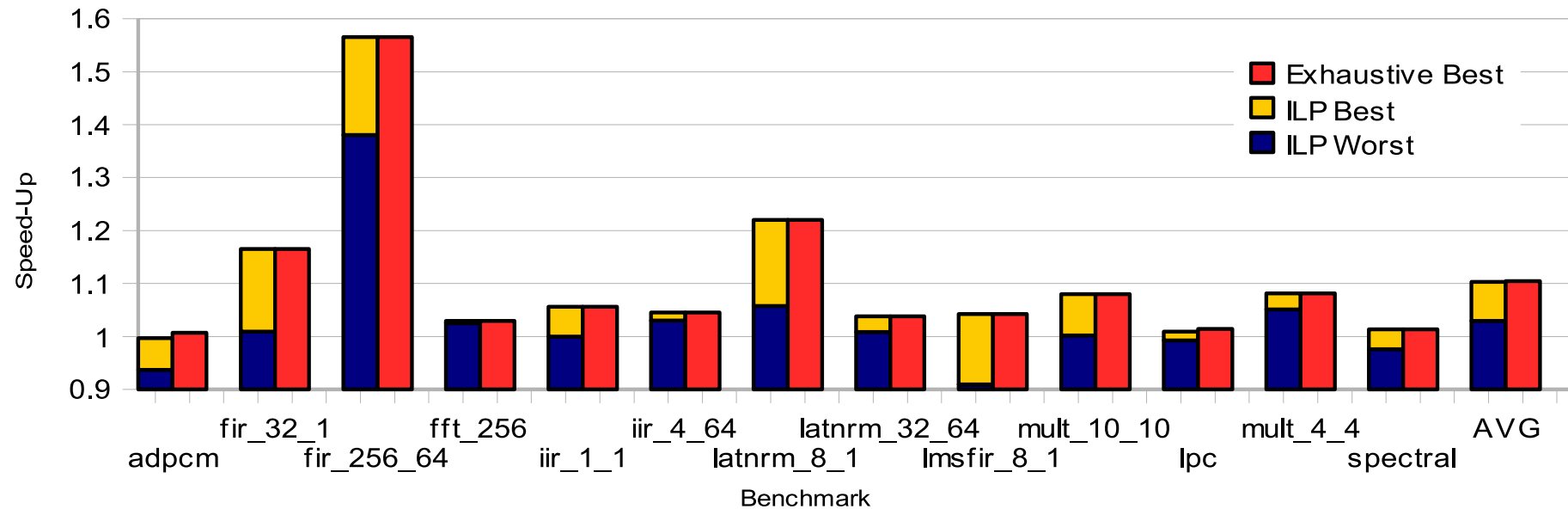


(c) "Inversion" of (b)



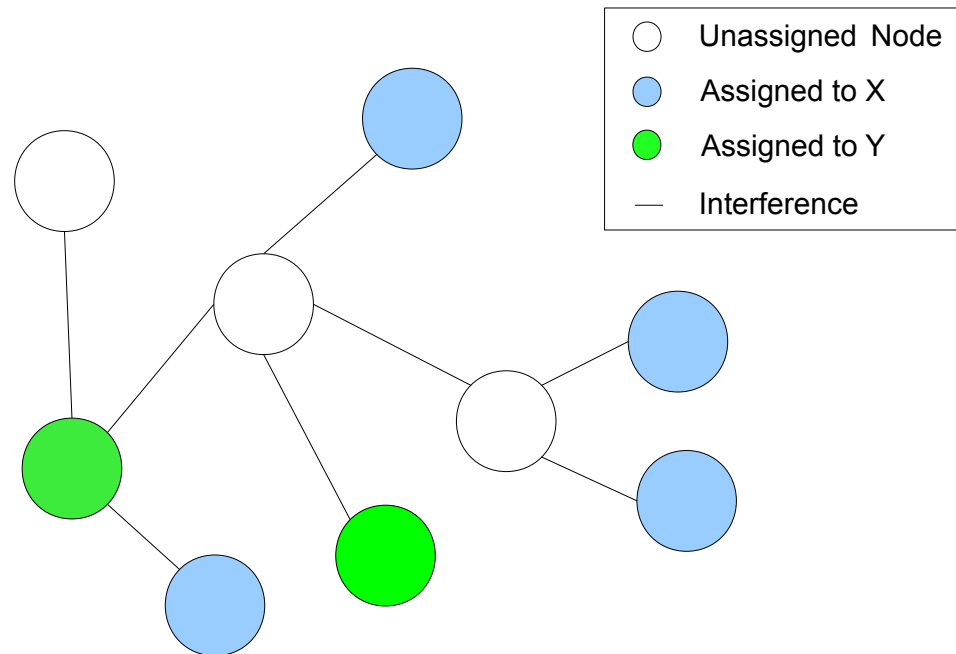
(d) Bank assignment resulting in worst performance

ILP Results

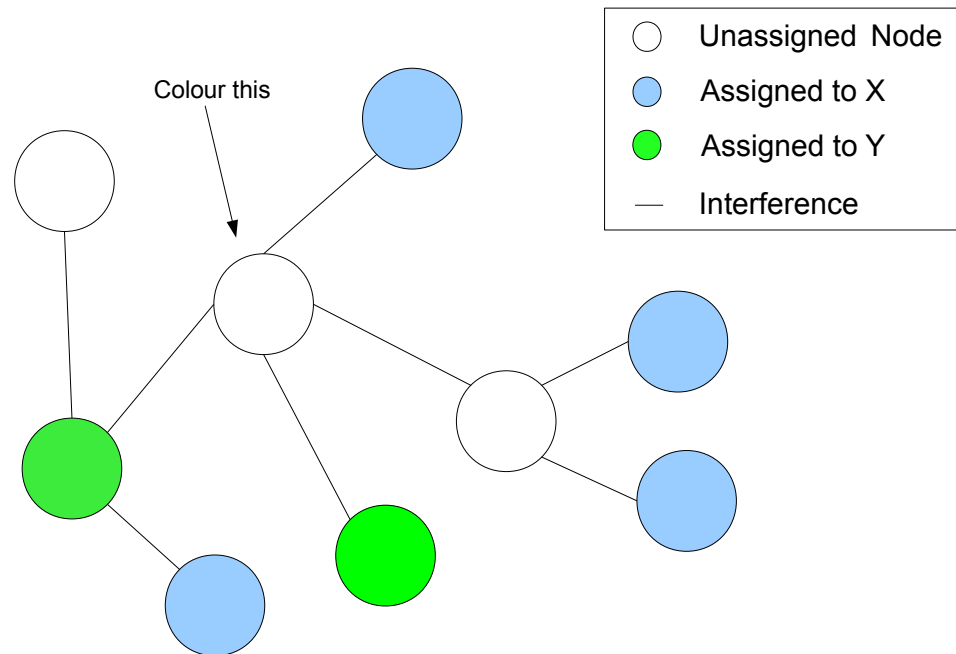


- Timings taken using an Analog Devices TigerSHARC TS-101.
- Could genetic programming colour without this range?

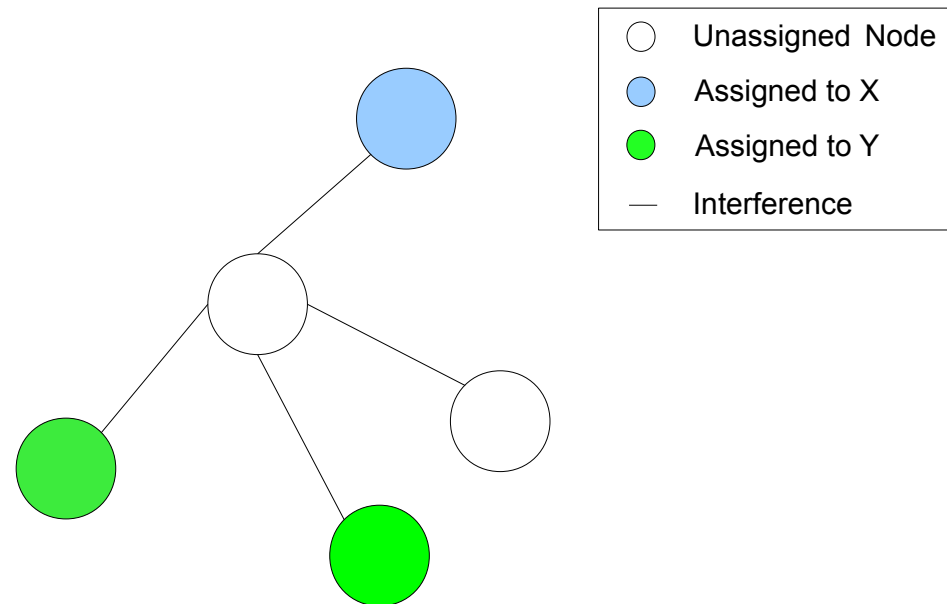
Graph Reduction



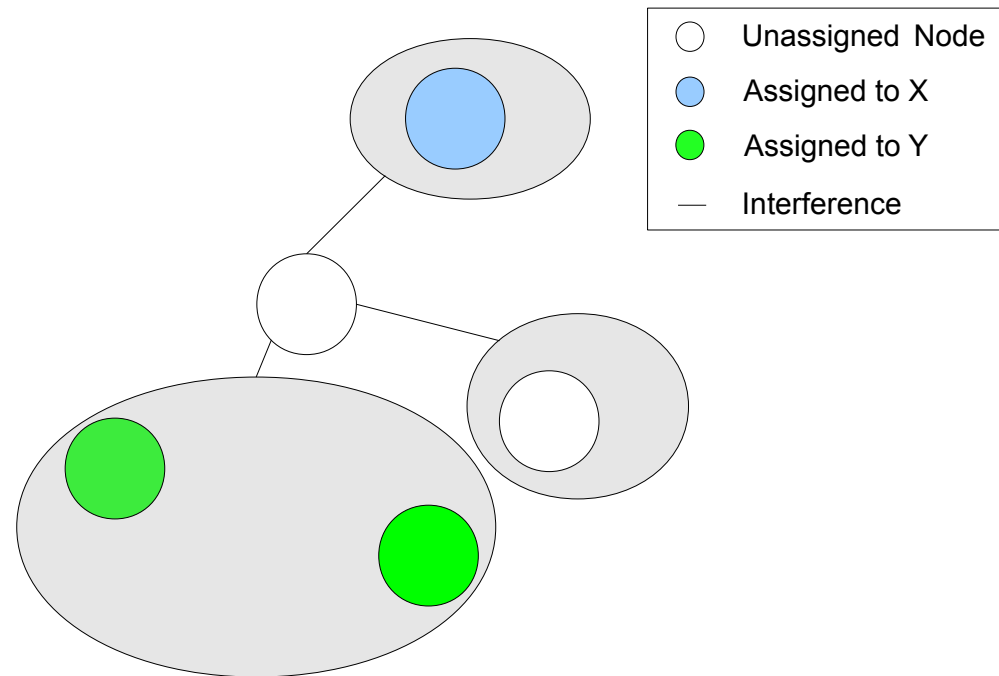
Graph Reduction



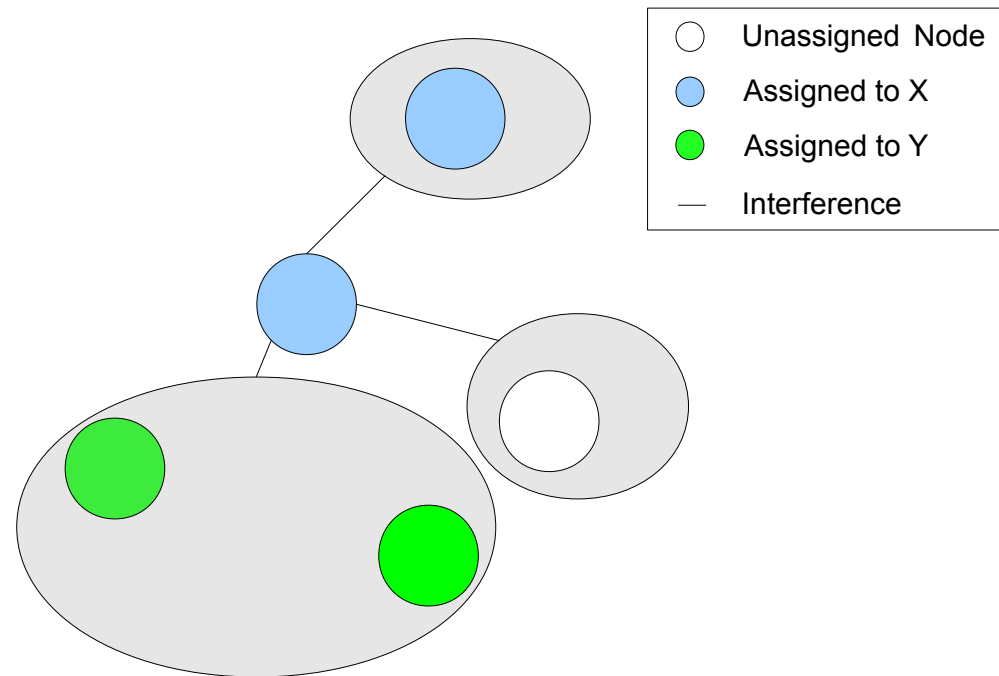
Graph Reduction



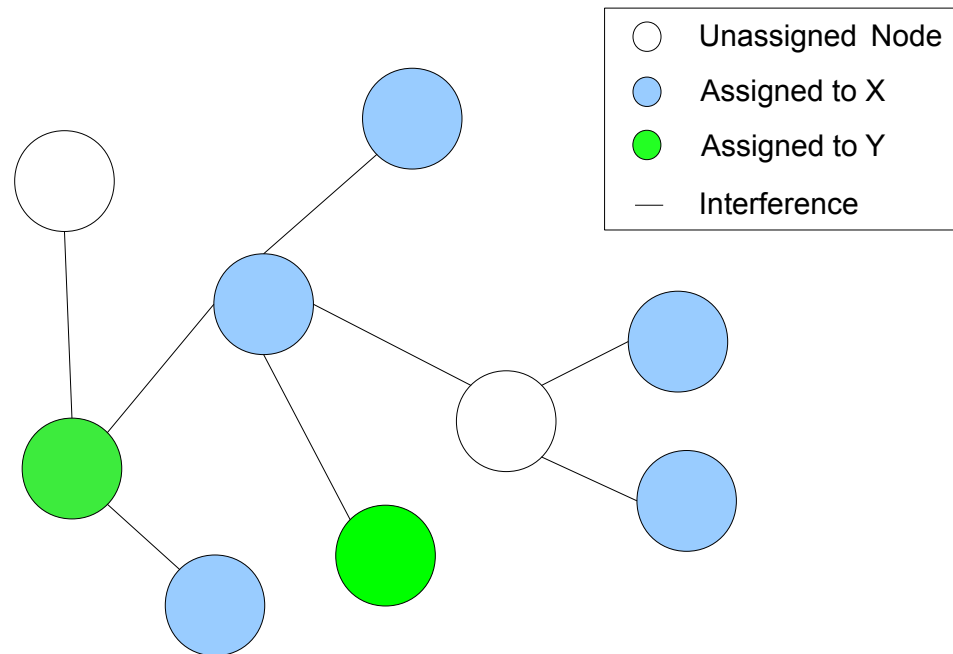
Graph Reduction



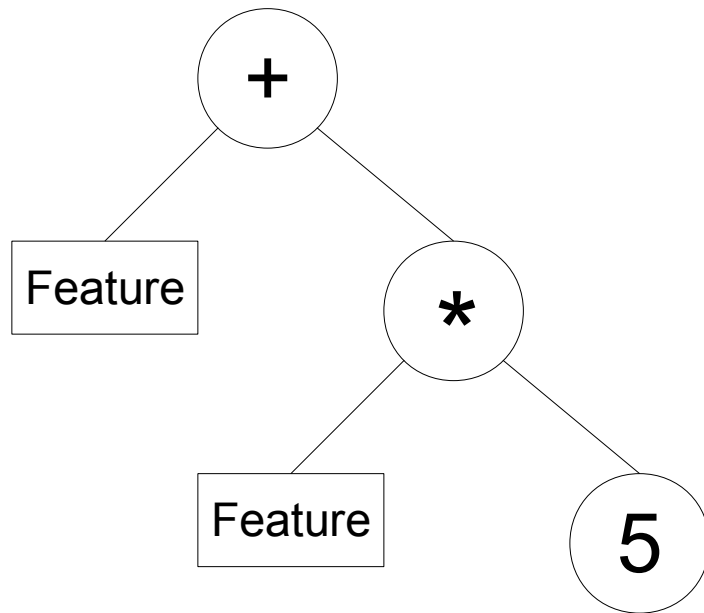
Graph Reduction



Graph Reduction



Heuristics Created



- New heuristic is easy to integrate into existing compiler.
- Heuristics are evaluated by scoring them relative to effectiveness of the colourings they pick.
- Fitness function:
 - $\frac{1}{speedup}$
 - Lower is better.

Program Features

Feature	Description
Parallel Interference	The no. of interferences at an immediate parallel level.
Expression Interference	The no. of interferences at an expression level.
Symbols: Aggregate	No. of aggregate symbols (e.g. structs) in group.
Symbols: Arrays	No. of array symbols in group.
Symbols: Pointers	No. of pointer symbols in group.
Symbols: Scalar	No. of scalar symbols in group.
Type: Integer	No. of integer symbols in group (e.g. an array of ints).
Type: Float	No. of floating point symbols in group.
Type: Complex	No. of non-numerical symbols (e.g. a void pointer).
Size	Total no. of bytes occupied by all variables in this group.

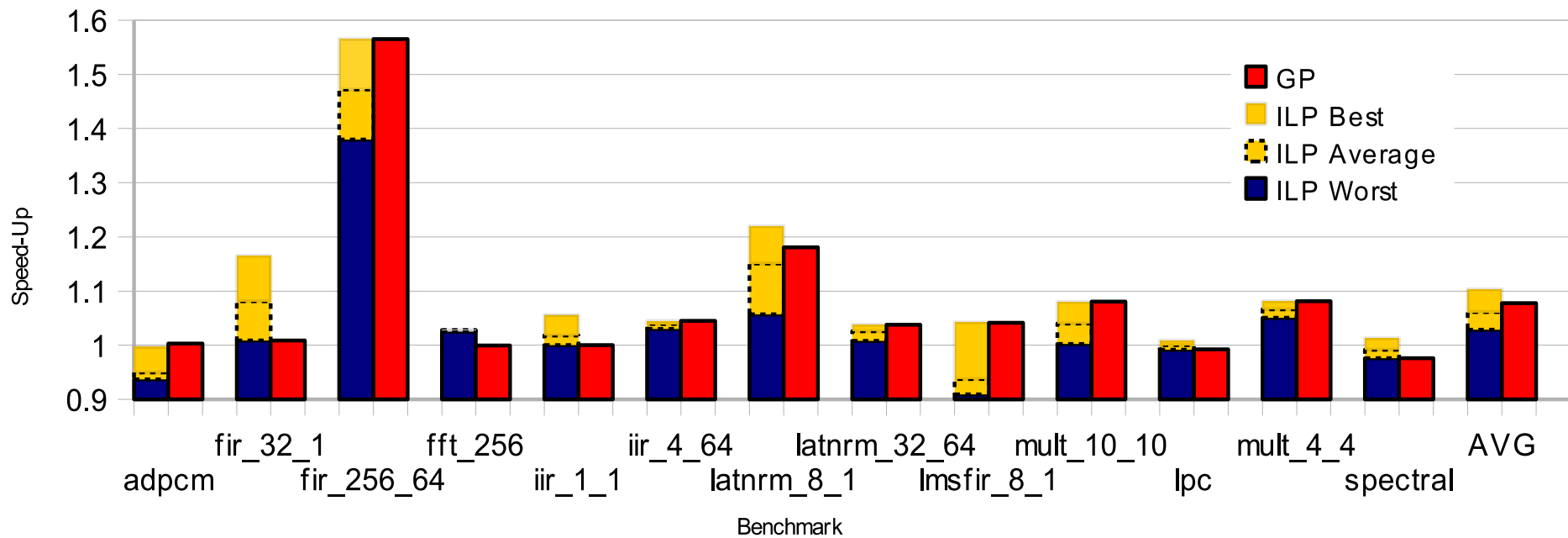
Functions

Function	No. Inputs	Description
Add, Sub	2	$A + B, A - B$
Mul, Div	2	$A * B, A \div B$
Sqrt	1	\sqrt{A} (returns 0.0 for negative inputs)
Abs	1	$ A $
Max, Min	2	$\max A, B, \min A, B$
EQ	4	if (A == B) { C } else { D }
GE	4	if (A >= B) { C } else { D }
GT	4	if (A > B) { C } else { D }
LE	4	if (A <= B) { C } else { D }
LT	4	if (A < B) { C } else { D }
Const	0	random ($0 \leq X < 1000$)

Genetic Programming Setup

- Standard Koza GP (Population 1024, 50 Generations.)
- Size constraint limit built into fitness function.
 - Penalises larger functions.
- GP picks order to colour nodes in:
 - Attempt to colour all nodes ...
 - actually colour node with highest score ...
 - repeat until all nodes coloured.
- Train and evaluate using leave-one-out cross-validation.

Results - ILP vs Genetic Programming



- Timings taken using an Analog Devices TigerSHARC TS-101.

Summary

- Source-level assignment:
 - Gives more control to the programmer.
 - Is easily portable.
- Genetic programming colourer:
 - Achieved an average speedup of 7.8% on UTDSP.
 - Found a better solution than the ILP colour 66% of the time.
 - Scales more effectively than an ILP colourer.
 - Is also easily portable.

Any Questions...?