Speculative execution

Value prediction

A.D. Temperman

Computer architecture (et4074)
TU Delft, Holland.
January 2000.
Version 2.

Speculative execution:

Example program:

1		1	4
2	becomes	2	5
3		3	6
4		test if prediction is successful	
5		(4)	
6		(5)	
		(6)	

Papers:

- Value Locality and Load Value Prediction [Lipasti, Wilkerson, Shen].
- Exceeding the Dataflow Limit via Value Prediction [Lipasti, Shen].
- The Predictability of Data Values [Sazeides, Smith].

Summary:

- Value locality.
- Value sequences.
- Prediction models:
  - Computational predictors.
  - Context based predictors.
  - Comparing predictors.
  - Test results.

Value locality:
The likelihood of a previously-seen value recurring repeatedly within a storage location.

Questions:
- What are the applications of value locality?
- How can you discover any useful amount of value locality?
- What can be said of the history depth?
- Are there differences in value locality between different kinds of data?
- Why does value locality exist in the first place?
- How can we exploit value locality?

Why does value locality exist:

- Data redundancy.
- Error-checking.
- Program constants.
- Computed branches.
- Virtual function calls.
- Glue code.
- Call-subgraph identities.
- Memory alias resolution.
- Register spill code.

Value sequences:

Classification of value sequences:

- Constant for example 5 5 5 5 5 5 5 ... [delta 0]
- Stride for example 1 2 3 4 5 6 7 8 ... [delta 1]
- Non-stride for example 28 –13 –99 107 23 456 ...

Repeated sequences:

- Repeated stride for example 1 2 3 1 2 3 1 2 3 ...
- Repeated non-stride for example 1 –13 –99 7 1 –13 –99 7 ...

Prediction models:

• *Computational predictors:*
Make a prediction by computing some function of previous values.

• *Context based predictors:*
Learn the value(s) that follow a particular context – a finite ordered sequence of values – and predict one of the values when the same context repeats.

Computational predictors:

• *Last value predictor:*
If the most recent value produced by an instruction is **v** then the prediction for the next value will also be **v**.

• *Stride predictor:*
Adds a delta to the most recent value. This delta is the difference between the two most recent values produced by an instruction.

Context based predictors:

• *Finite Context Method Predictors:*

Rely on mechanisms that predict the next value based on a finite number of preceding values.

An order **k** finite context method predictor uses **k** preceding values.

These predictors are constructed with counters that count the occurrences of a particular value immediately following a certain context (pattern).

The predicted value is the one with the maximum count.

They can't predict non-repeating sequences, but they are very flexible for repeating sequences (not only strides).

Comparing predictors:

• *Constant sequence:*	5 5 5 5 5 5 5 5
Prediction of last value predictor:	*0* 5 5 5 5 5 5 5
Prediction of stride predictor [delta 0]:	*0 0* 5 5 5 5 5 5
Prediction of 2nd order fcm predictor:	*0 0* 5 5 5 5 5 5
• *Non-repeating stride sequences:*	2 4 6 8 10 12 14 16
Prediction of stride predictor [delta 2]:	*0 0* 6 8 10 12 14 16
• *Repeating stride sequences:*	1 2 3 4 1 2 3 4 1 2 3 4
Prediction of stride predictor [delta 1]:	*0 0* 3 4 **5 –2** 3 4 **5 –2** 3 4
Stride predictor with hysteresis:	*0 0* 3 4 **5** 2 3 4 **5** 2 3 4
Prediction of 2nd order fcm predictor:	*0 0 0 0 0 0* 3 4 1 2 3 4
• *Repeating non-stride sequences:*	1 5 3 7 1 5 3 7 1 5 3 7
Prediction of 2nd order fcm predictor:	*0 0 0 0 0 0* 3 7 1 5 3 7

Test results: **[Sazeides and Smith]**

– A large portion of correct predictions is captured by all predictors.

– A smaller portion (but still a significant fraction) of correct predictions is only captured by the finite context method predictor.

– A small number of values are not predicted correctly by any model.

– Stride and last value prediction capture only a very small number of the correct predictions that the finite context method predictor misses.

Conclusion:

A **hybrid** scheme might be useful, consisting of a stride predictor and a finite context method predictor. This enables high prediction accuracies at lower cost.

There is no need for an extra last value predictor.

Further reading:

Modeling program predictability [Sazeides, Smith]:

- The three key components of predictability are: **generation, propagation** and **termination**. Example:

  Sequence: 1 2 3 4 5 6 7 8 1

  Prediction by a stride predictor: *0 0* 3 4 5 6 7 8 **9**

  generation propagation termination