an URI / NEU collaboration

Realizing High IPC Through A Scalable Memory-Latency Tolerant Multipath Microarchitecture

submitted to
Workshop on Memory Access Decoupled Architectures (MEDEA) 2002
in conjunction with PACT 2002

student **David Morano**
advisors **Professor David Kaeli**
Professor Augustus Uht

NUCAR talk 02/07/26

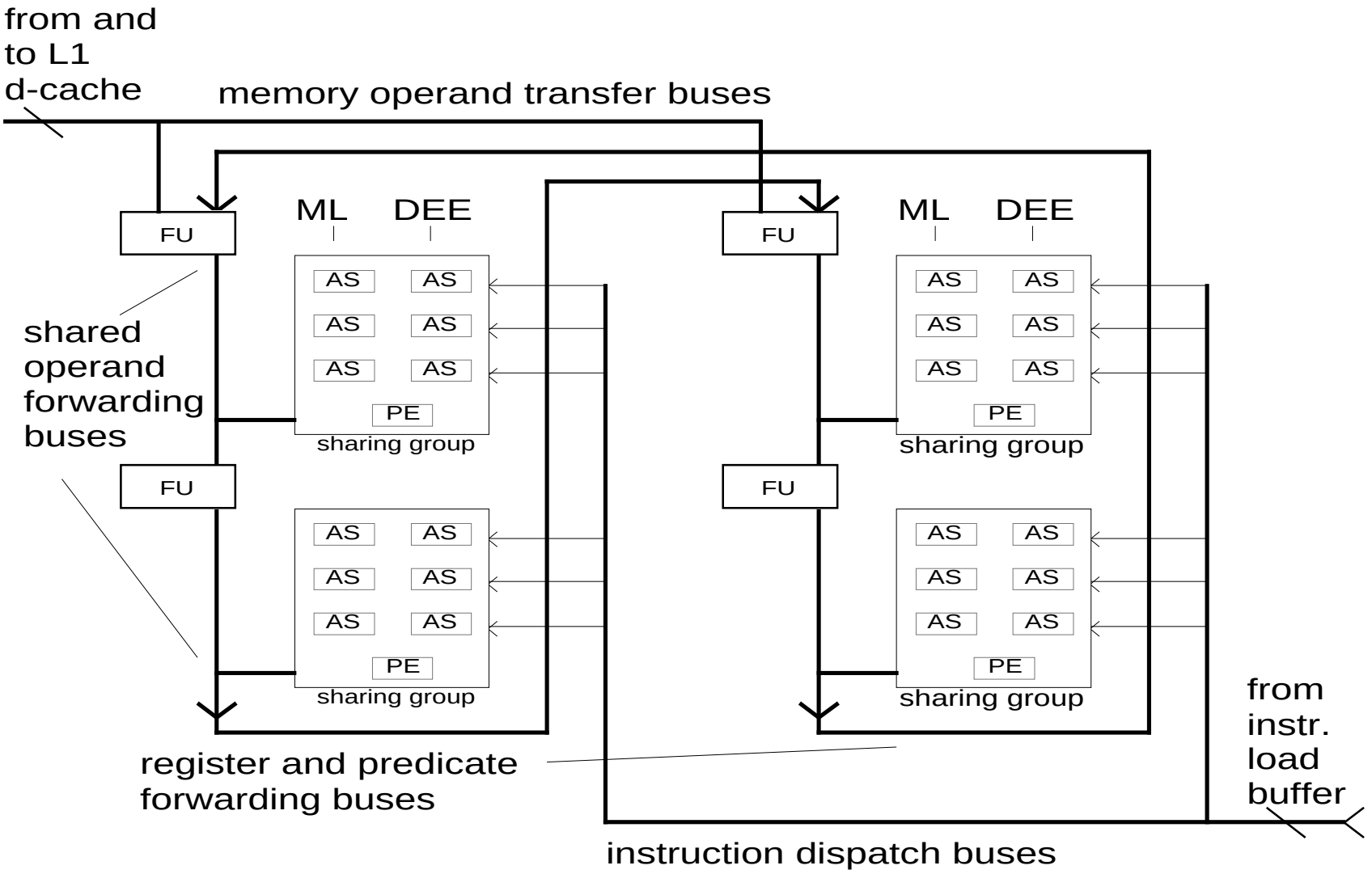
Outline

- **introduction**
- **Levo solution**
- **results**
  - stats
  - L1-I cache latency tolerance
  - L1-D cache latency tolerance
  - L2 cache latency tolerance
  - main-memory latency tolerance
  - IPC results
  - L0 effect
- **summary**

introduction

- **memory is slow !**
  - DRAM is slow
  - most all memory is DRAM (or slower disk)
  - need to hide the latency
- **use caches !**
  - L1 & L2
  - good, but is it good enough ?
- **is something closer than the L1 cache ?**
  - L0 cache (in MFUs)
  - other instructions (segmented buses)
- **Levo does other things as well (DEE, et cetera)**

Levo overview



simulation results (stats)

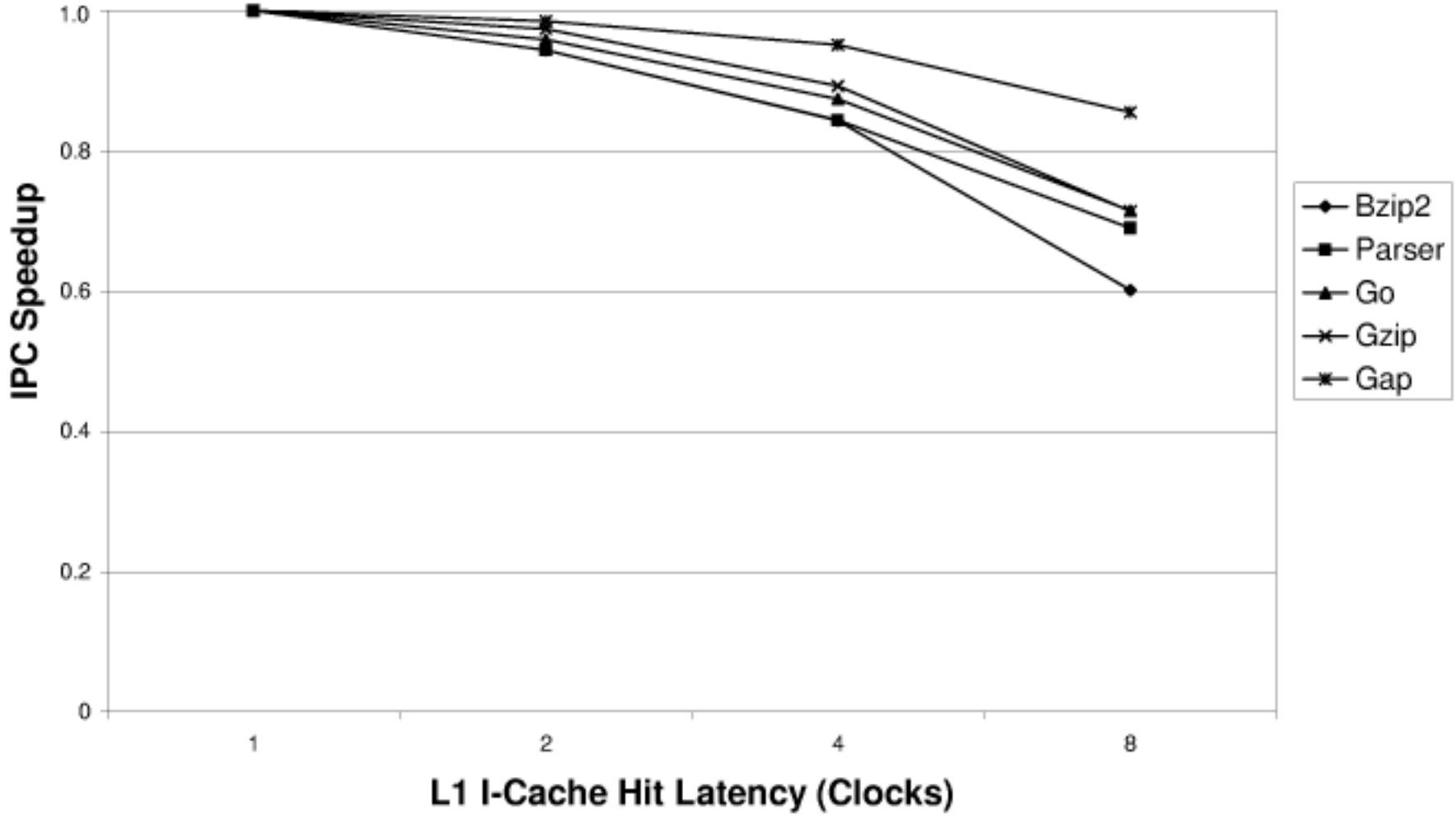
benchmark	bzip2	parser	go	gzip	gap
br. prediction accuracy	90.5%	92.6%	72.1%	85.4%	94.5%
avg. L1-I hit rate	97.2%	96.6%	92.4%	94.7%	89.0%
avg. L1-D hit rate	98.8%	99.0%	98.8%	99.8%	99.3%
avg. L2 hit rate	90.1%	86.0%	96.8%	73.0%	88.5%
dynamic cond. brs.	12.0%	11.0%	12.1%	13.4%	6.5%

IPC results

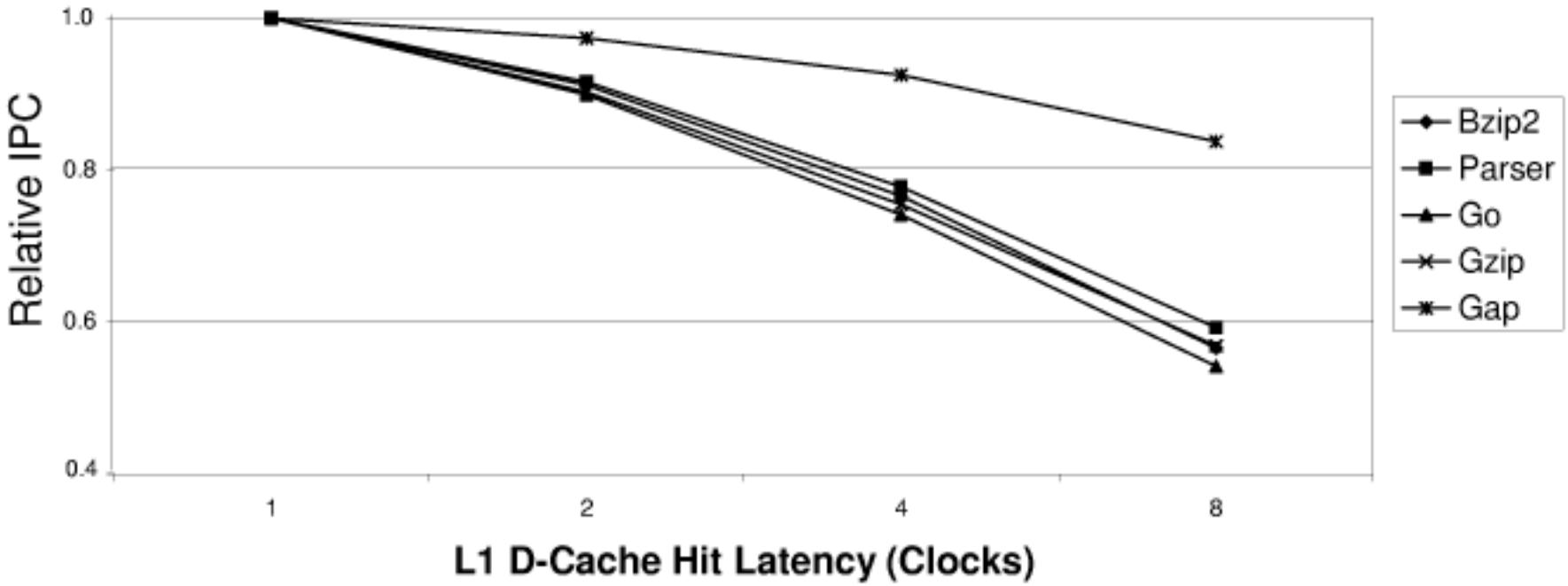
geometry	8-4-8-8	8-8-8-8	16-8-8-8	32-2-16-16	32-4-16-16
bzip2	4.2	5.0	5.8	5.4	5.7
parser	4.3	4.6	5.3	5.0	5.4
go	5.1	5.9	6.7	6.5	6.8
gzip	5.0	6.3	7.0	6.7	7.2
gap	6.0	7.5	7.5	8.9	7.9
HAR-MEAN	4.8	5.7	6.4	6.3	6.5
% speedup over SP	50	46	39	50	41

L1 1 ck, L2 10 cks, M-M 100 cks

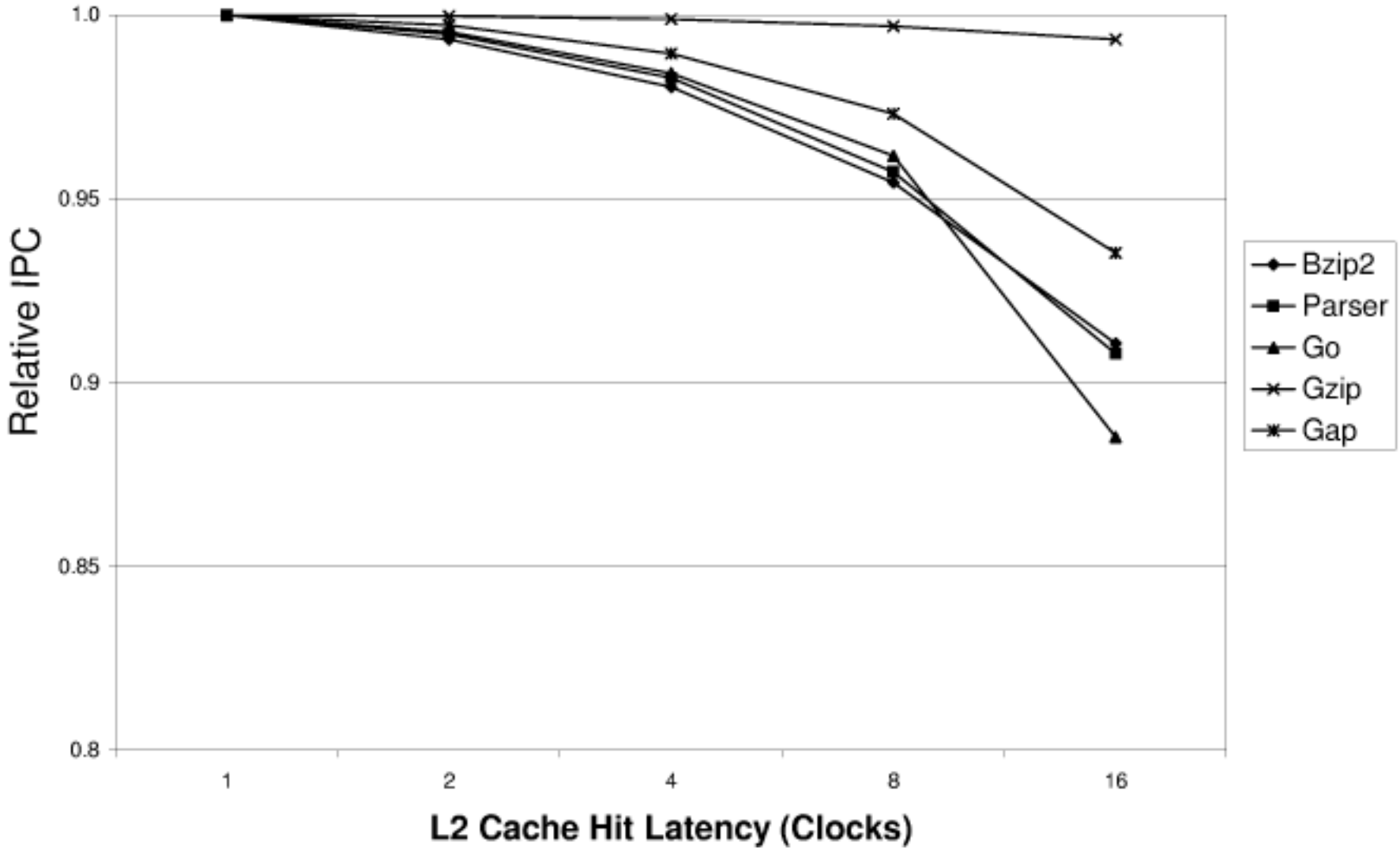
L1-I cache latency tolerance



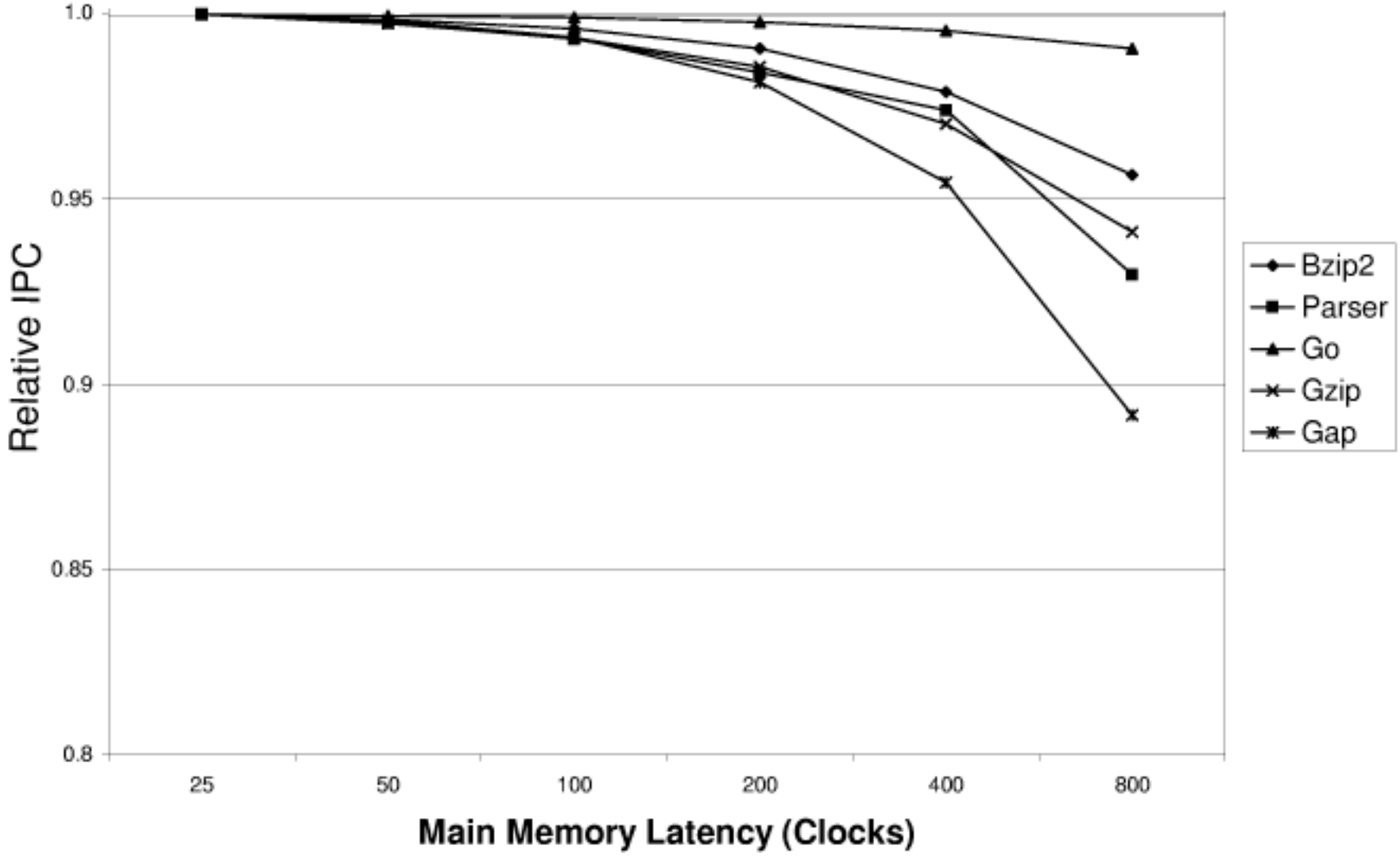
L1-D cache latency tolerance



L2 cache (I/D) latency tolerance



main-memory latency tolerance



effect of L0 caches

	bzip2	parser
% of all loads satified by L0 due to a backwarding request	3.6%	5.2%
% of all loads satified by L0 but w/o any backwarding request	18.2%	28.8%

summary

- **shown a tolerance to substantial main-memory latency**
  - due in part to L1 and L2
  - but also due to L0 and other close instructions
- **"close" resources have been explored by others**
  - register caches
  - "L0" caches
- **"close" resources are integral to the Levo solution**