

メニーコア時代に向けた インオーダ・アーキテクチャ再考

三好健文 入江英嗣 松村雄貴 吉永努

電気通信大学大学院情報システム学研究科

概要

- ▶ マルチ/メニーコアプロセッサを構成するコアに適しているのは、インオーダー？アウトオブオーダー？
- ▶ 同じISAで比較してみよう(鬼斬2 / Alpha)
- ▶ メニーコアを想定したメモリ階層を3パターン想定してシミュレーション
- ▶ アウトオブオーダーは速い
- ▶ インオーダーもありかもしれない

マルチ/メニーコアプロセッサへの流れ

- ▶ プロセス微細化によるTr数の増加
- ▶ 1チップに多数のプロセッサを搭載可能に
- ▶ 多数のプロセッサでの並列処理
 - ▶ 処理性能の向上
 - ▶ 消費電力の削減
- ▶ いくつか（沢山）の研究課題
 - ▶ アーキテクチャ, プログラミングモデル,

マルチ/メニーコアプロセッサの例

- ▶ Nehalem(2~8コア), Phenom II(2~6コア),
- ▶ POWER7(2~8コア), ...
- ▶ Cell/B.E.(1PPE + 8SPE),
- ▶ TILE64(64コア), SCC(48コア),
- ▶ Rigel(1024コア), ...

マルチ/メニーコアプロセッサの例

- ▶ Nehalem(2~8コア), Phenom II(2~6コア),
- ▶ POWER7(2~8コア), ...

アウトオブオーダー

- ▶ Cell/B.E.(1PPE + 8SPE),
- ▶ TILE64(64コア), SCC(48コア),
- ▶ Rigel(1024コア), ...

インオーダー

アウトオブオーダー v.s. インオーダー

とっても古い議論

▶ アウトオブオーダー

○ メモリアクセスレイテンシの隠蔽

× 面積オーバヘッド

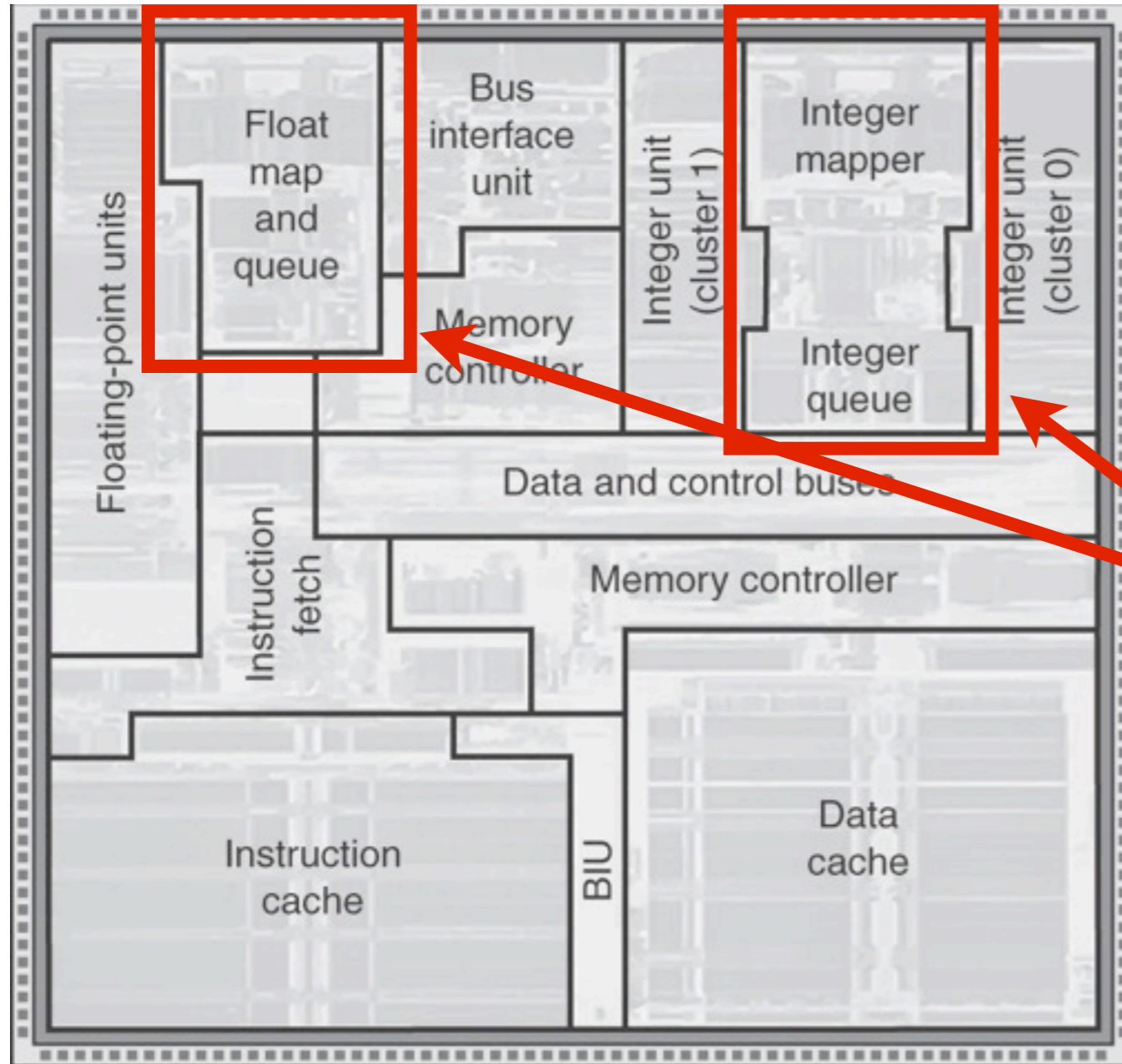
▶ インオーダー

○ シンプル

× メモリアクセスで処理がストール

アウトオブオーダーのHWリソースオーバヘッド

Alpha 21264の場合



OoOを実現するために
必要なHW機構

[12] J.L. Hennessy and D.A. Patterson, Computer Architecture, Fourth Edition: A Quantitative Approach, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.

アウトオブオーダー vs インオーダー

演算処理能力では...アウトオブオーダーに軍配

▶ アウトオブオーダー

○ メモリアクセスレイテンシの隠蔽

× 面積オーバーヘッド

▶ インオーダー

○ シンプル

× メモリアクセスで処理がストール

アウトオブオーダー vs インオーダー

マルチ/メニーコアアーキテクチャという観点では？

- ▶ チップ内に搭載可能なプロセッサ数
- ▶ メモリ階層アーキテクチャ
 - ▶ キャッシュサイズ
 - ▶ キャッシュ構成方式
- ▶ 消費電力
- ▶ 同期や一貫性制御のしやすさ

アウトオブオーダー vs インオーダー

マルチ/メニーコアアーキテクチャという観点では？

▶ チップ内に搭載可能なプロセッサ数

▶ メモリ階層アーキテクチャ

▶ キャッシュサイズ

▶ キャッシュ構成方式

今回の対象

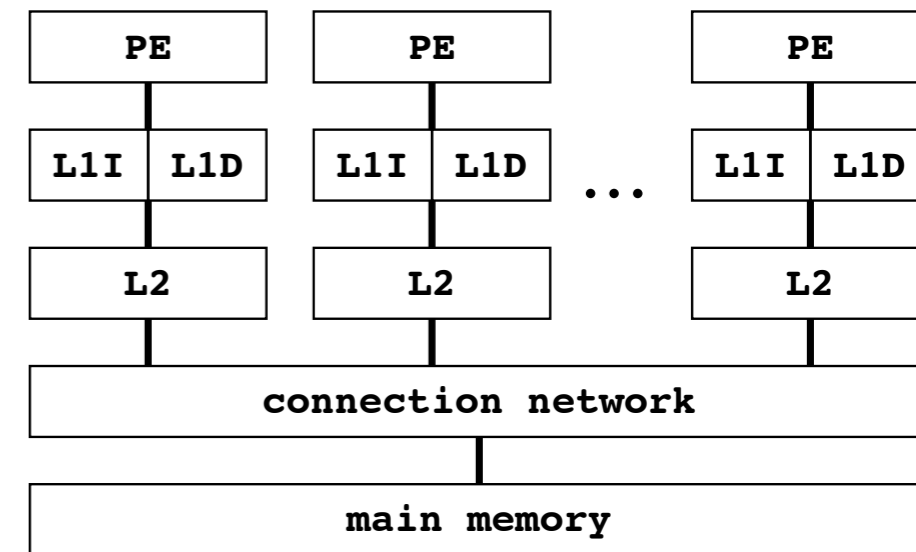
▶ 消費電力

▶ 同期や一貫性制御のしやすさ

シミュレータで評価

鬼斬2 (rev4237) を使ってシミュレーション

- ▶ AlphaなISA→スケジューラだけ変更
- ▶ マルチ/メニーコアプロセッサ中の各コアの処理性能を評価
- ▶ 3パタンのメモリ階層を想定
- ▶ SPEC INT2006/FP2006(ref)



フェッチ発行幅	8
INT演算発行幅	2
FP演算発行幅	2
メモリ演算発行幅	2
アドレス演算発行幅	2

ケーススタディ

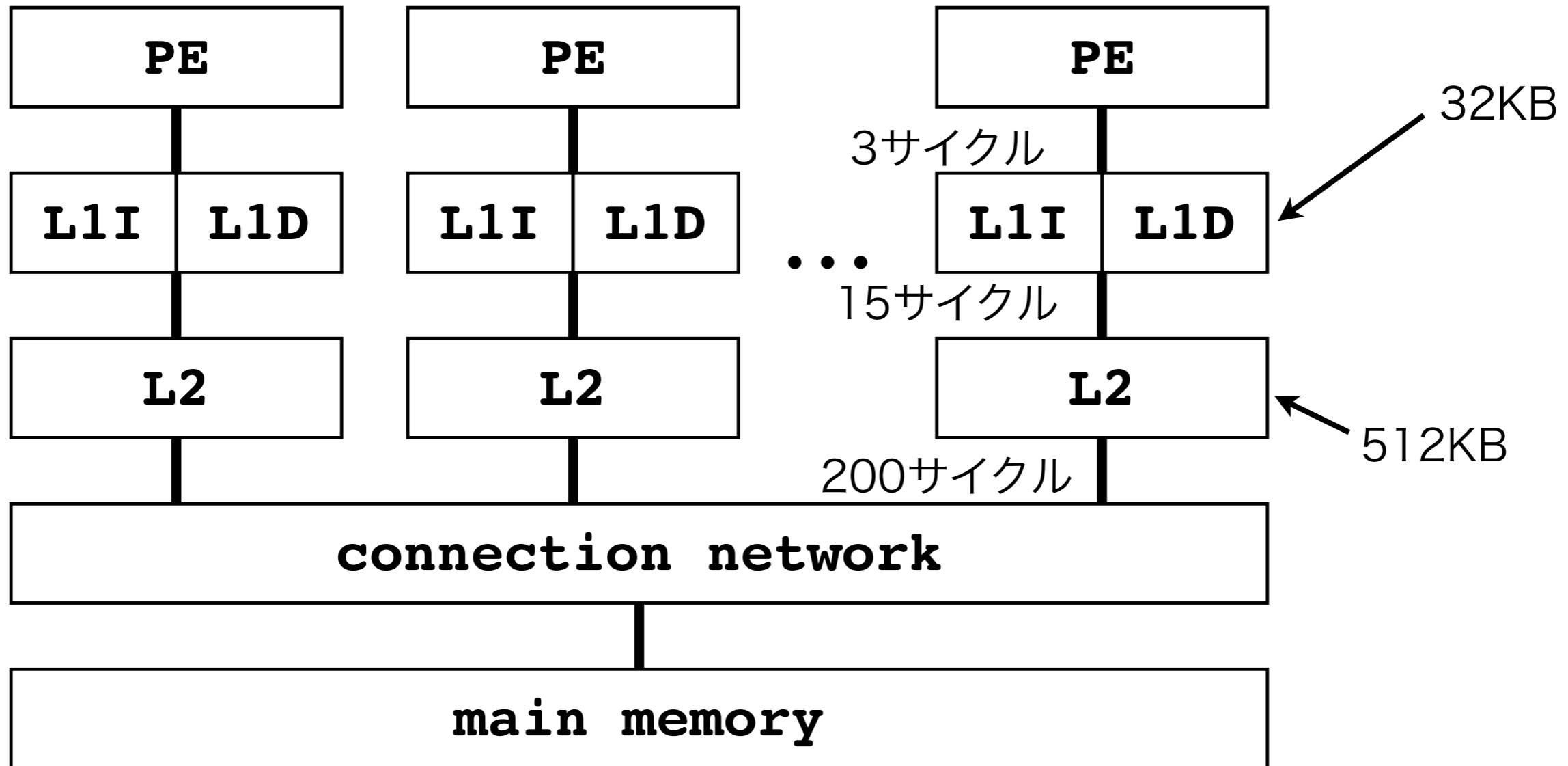
メモリ階層アーキテクチャの異なる

3種類のマルチ/メニーコアプロセッサを想定

	L1 Dキャッシュ	L2 キャッシュ	例
ケーススタディ1	32KB(3cycles)	512KB(15cycles)	Nehalem, POWER7
ケーススタディ2	512B(3cycles)	256KB(6cycles)	Cell/B.E., ...
ケーススタディ3	8KB(3cycles)	16KB(15cycle)	Rigel, ...

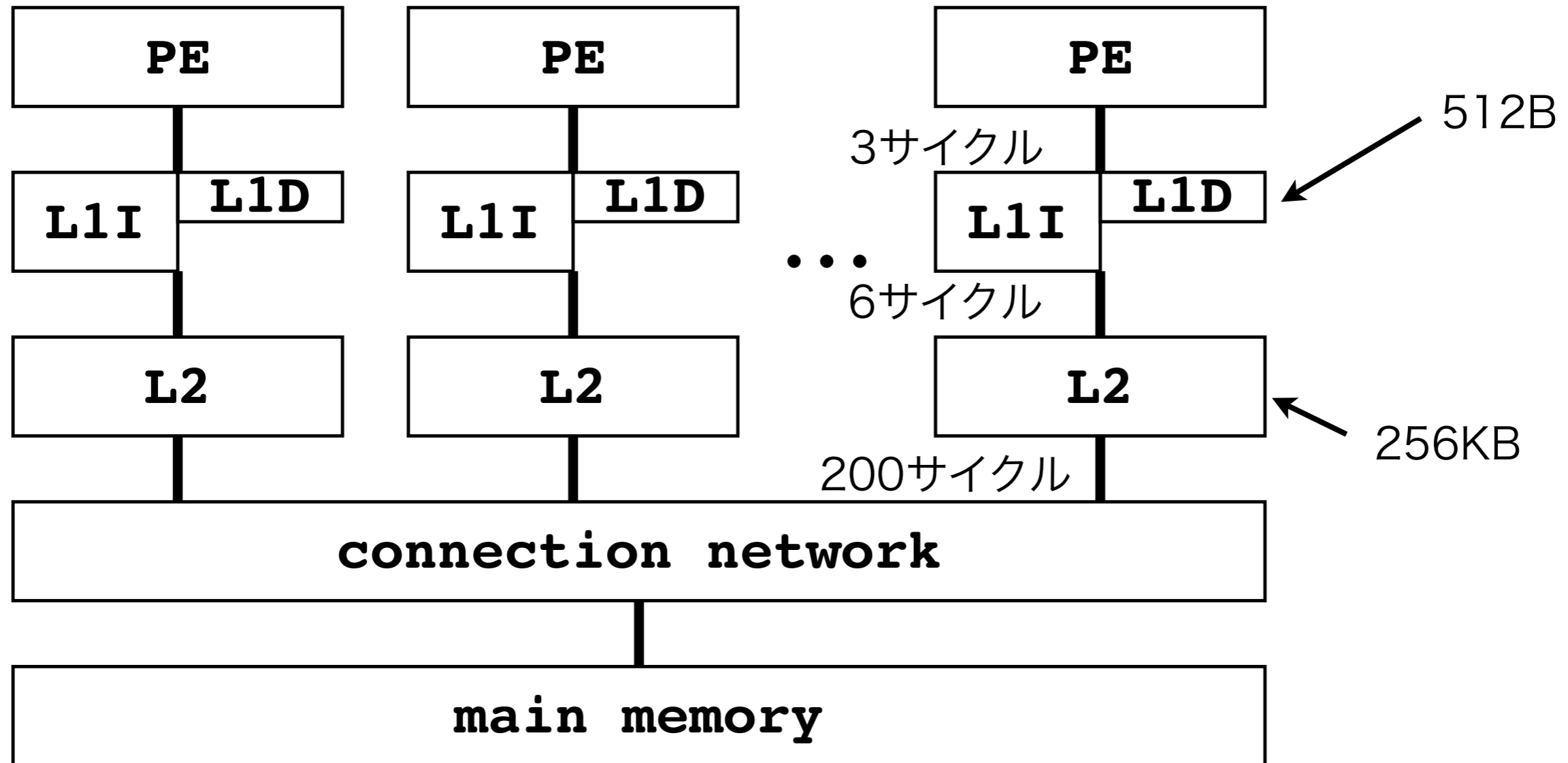
ケーススタディ1

Nehalem, Phenom II, POWER7, ...



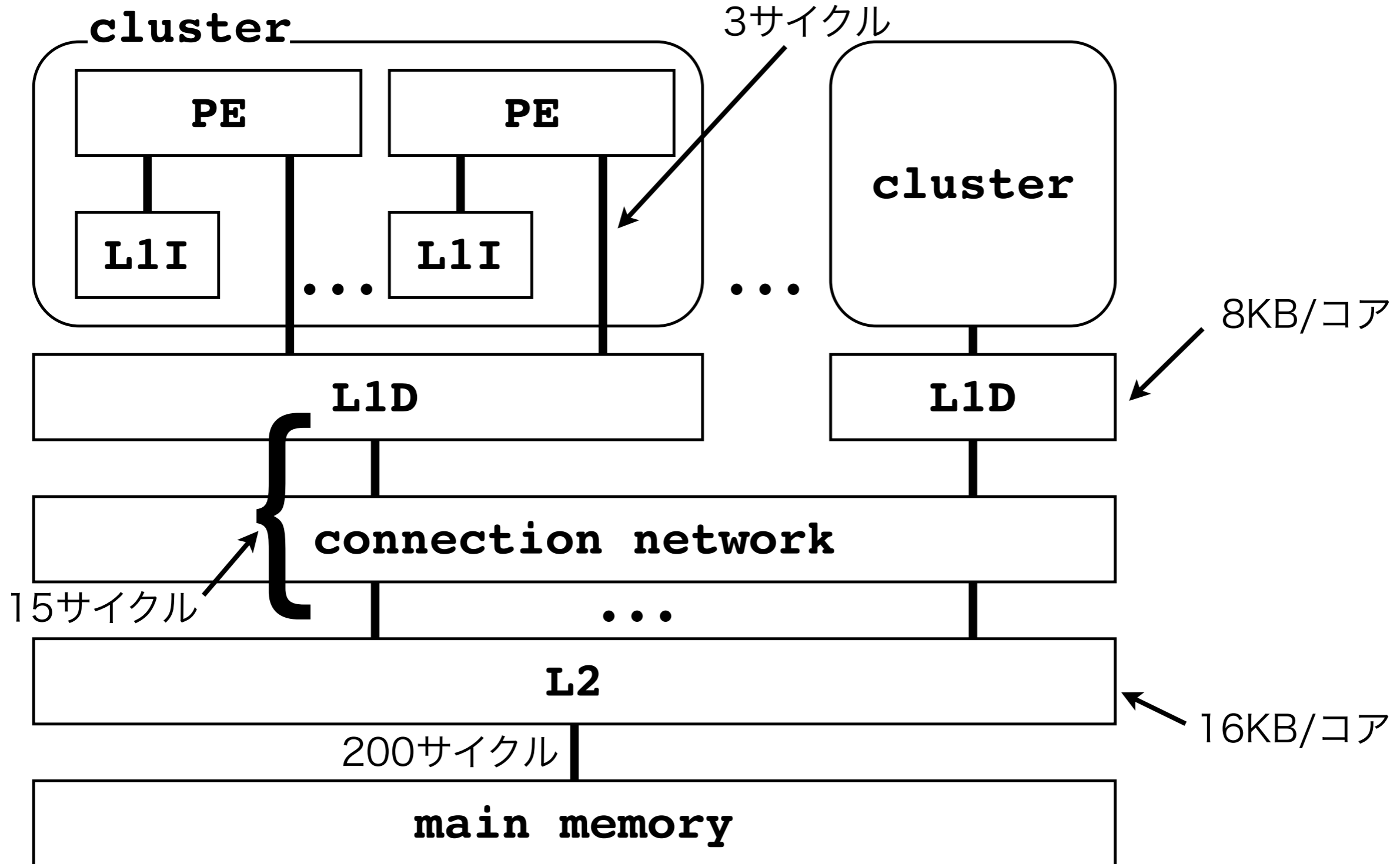
ケーススタディ2

Cell/B.E.,

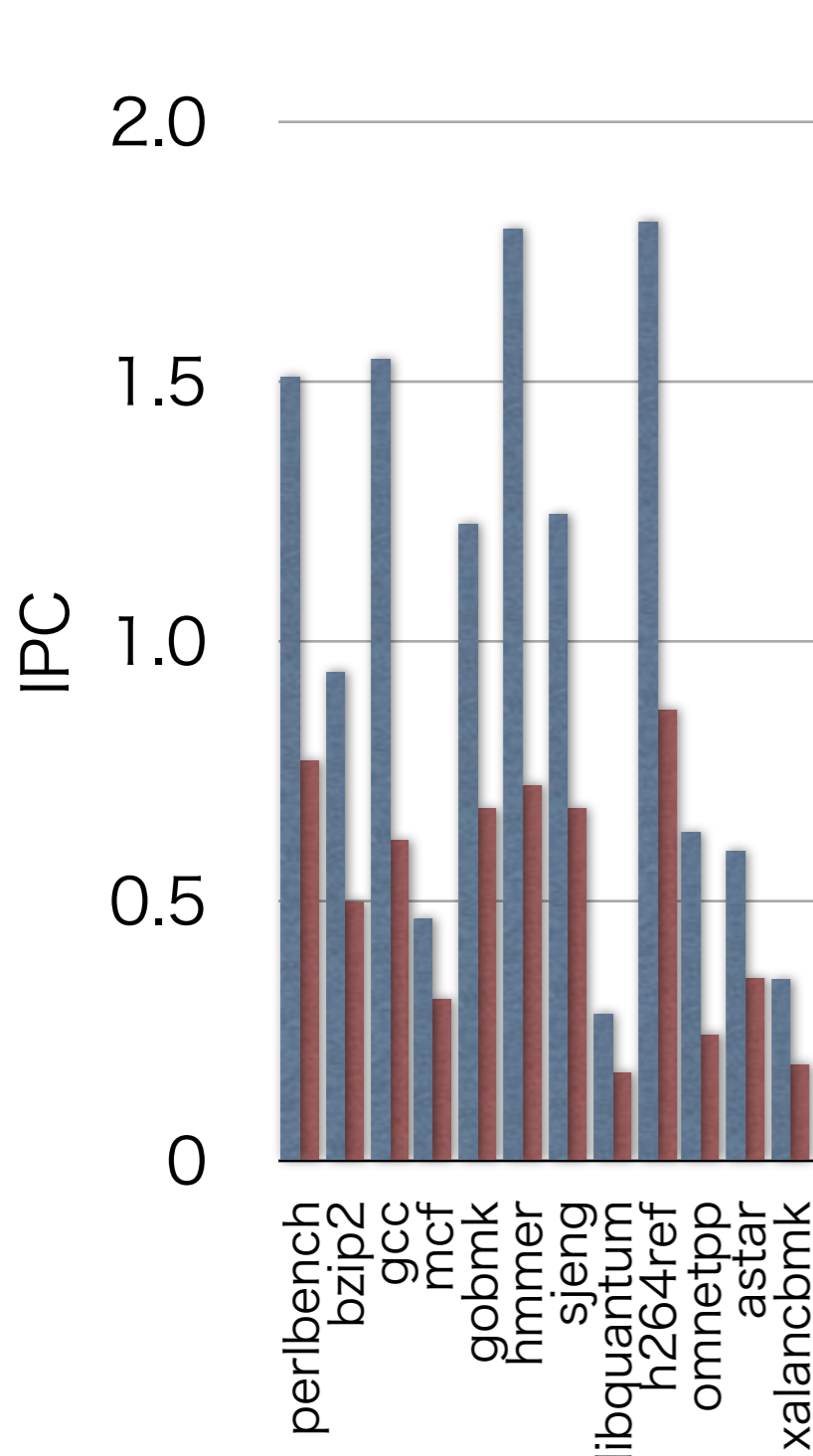


ケーススタディ3

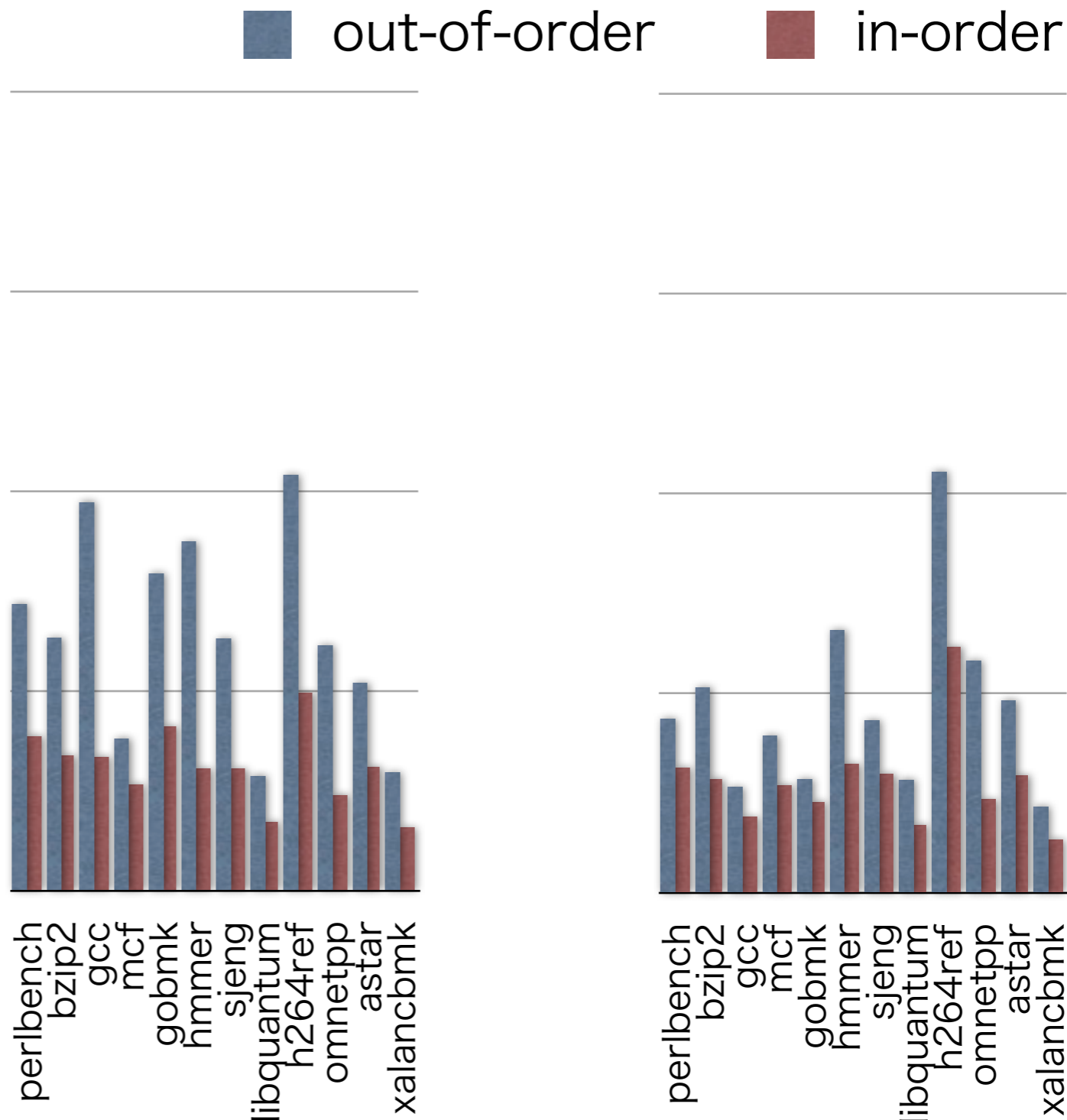
Rigel,...



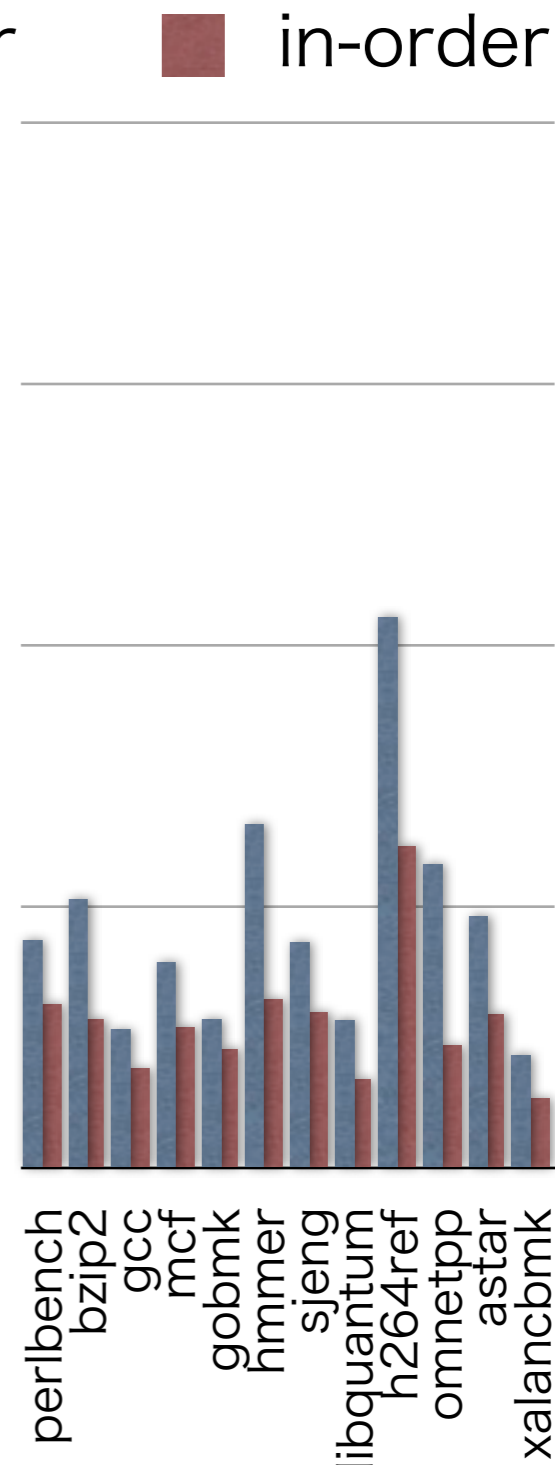
IPC/INT2006



ケーススタディ1



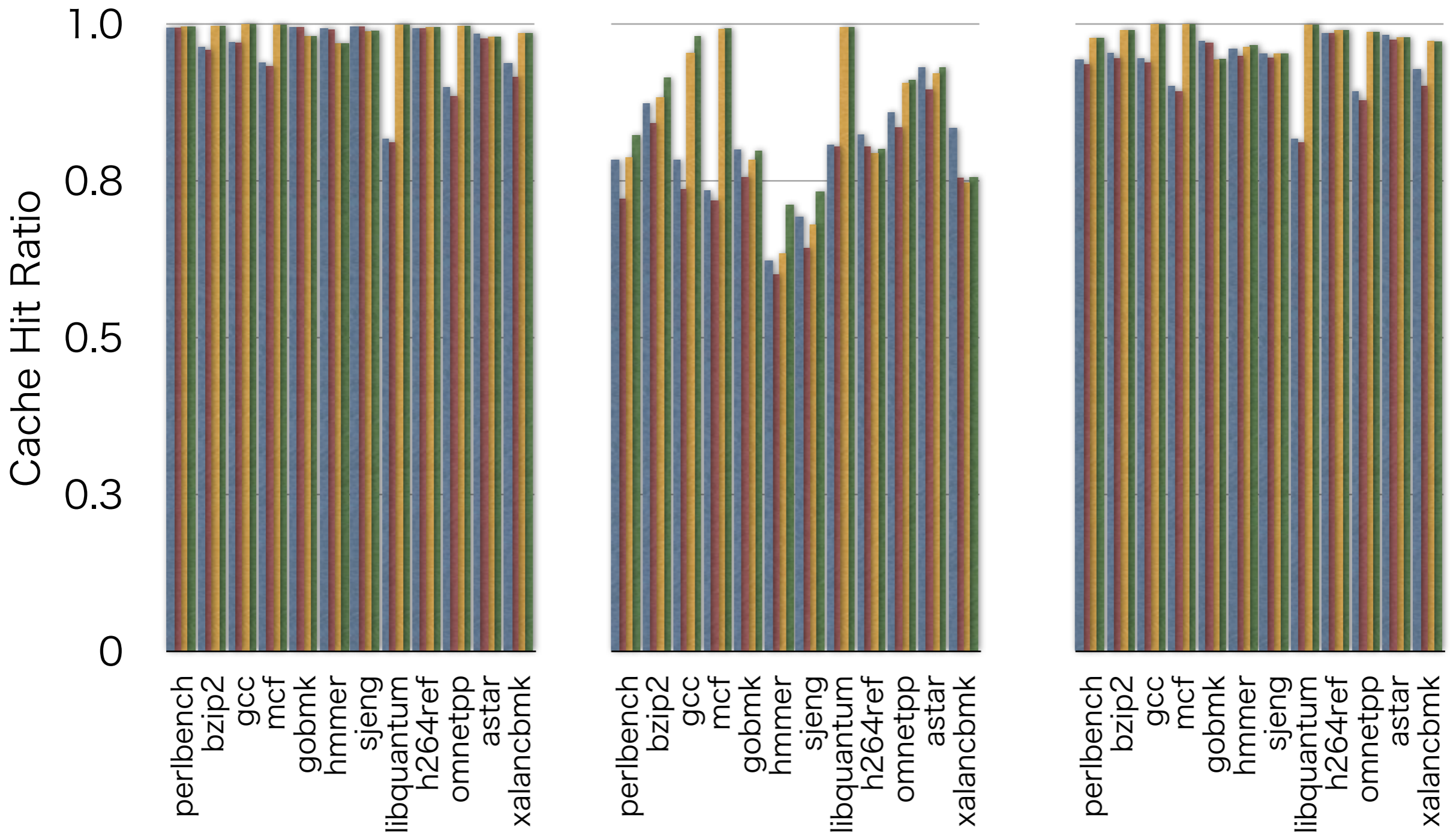
ケーススタディ2



ケーススタディ3

キャッシュヒット率(L1D)/INT2006

■ out-of-order read ■ in-order read
■ out-of-order write ■ in-order write

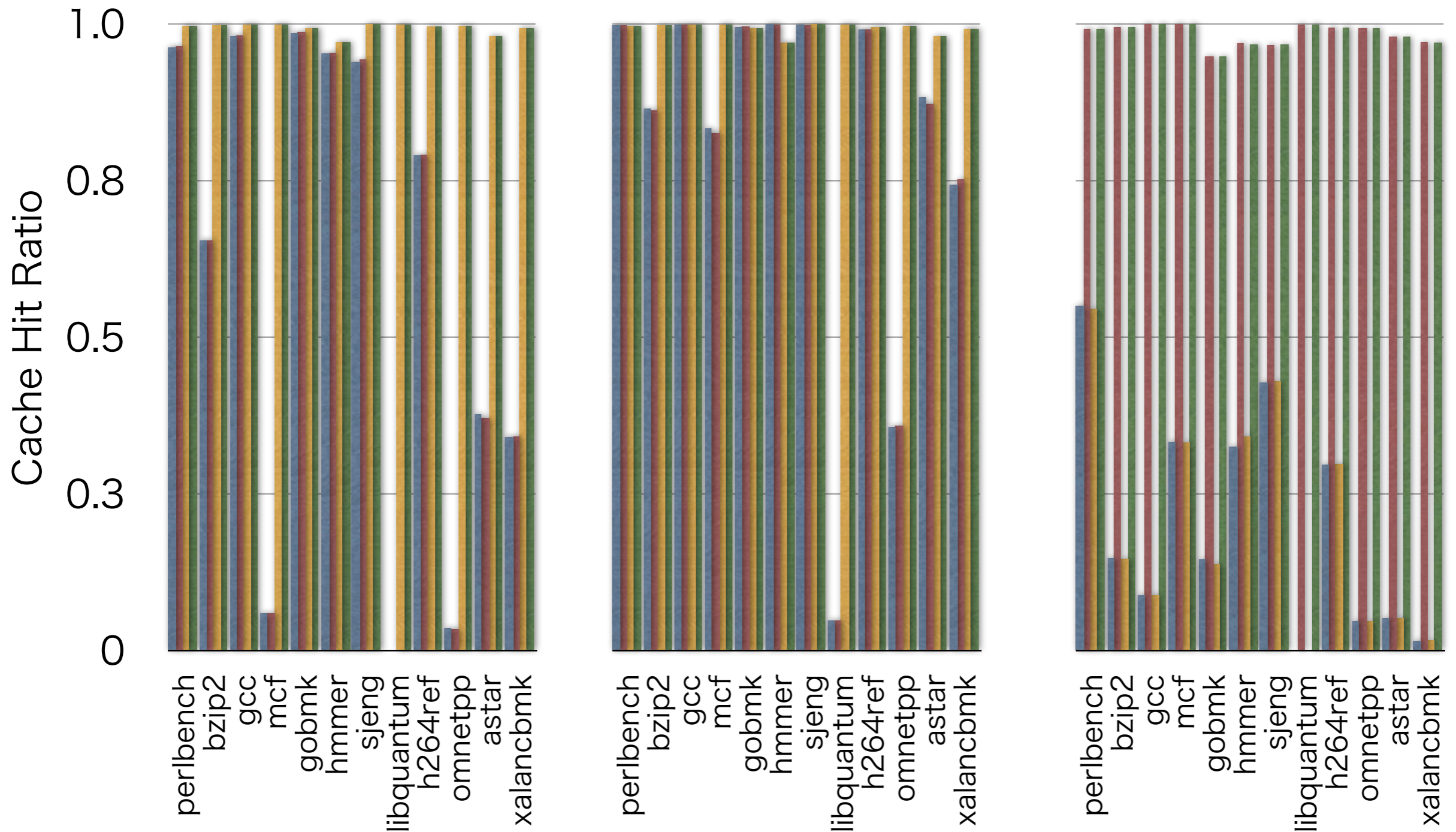


ケーススタディ1

ケーススタディ2

ケーススタディ3

キャッシュヒット率(L2)/INT2006

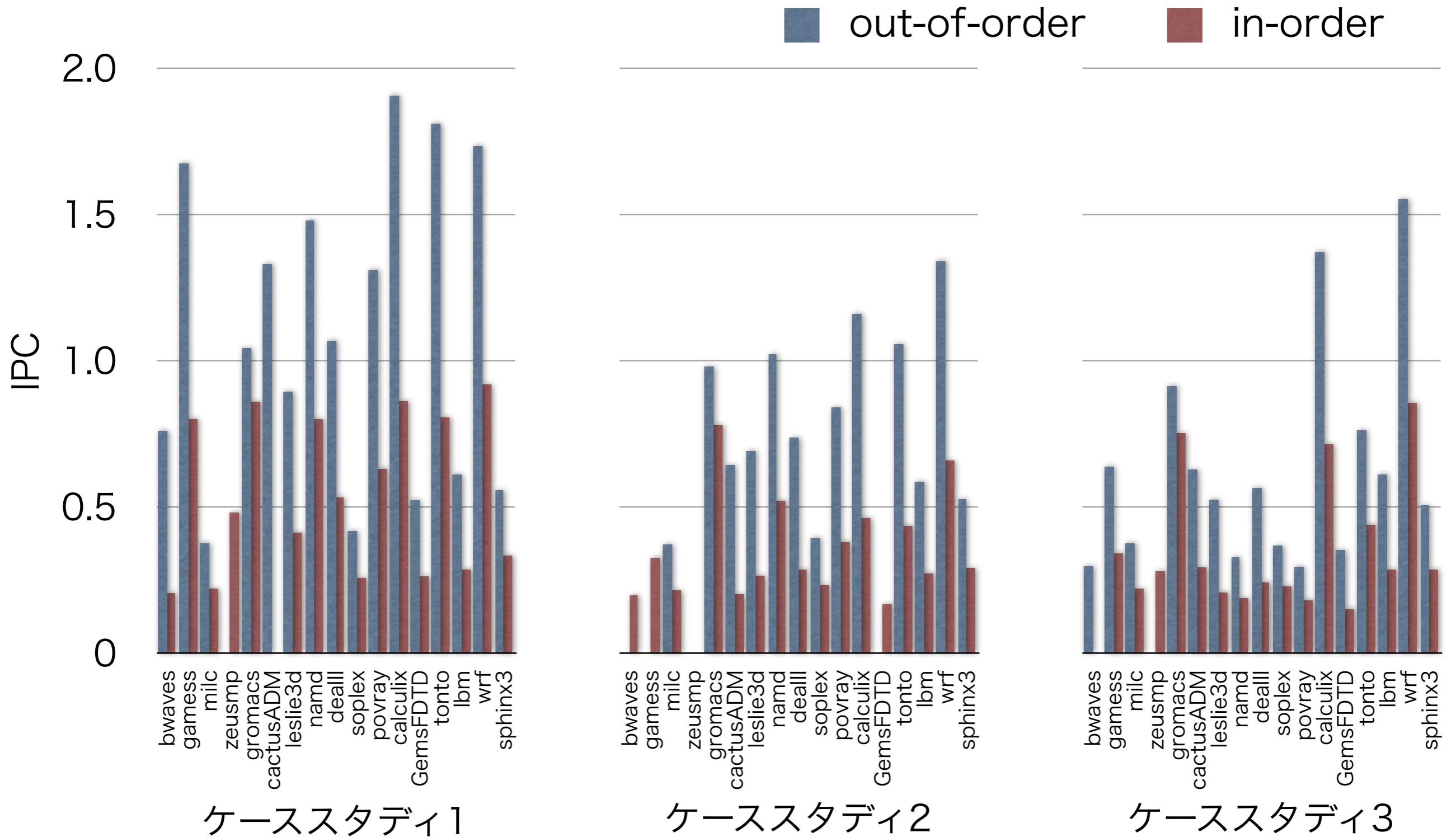


ケーススタディ1

ケーススタディ2

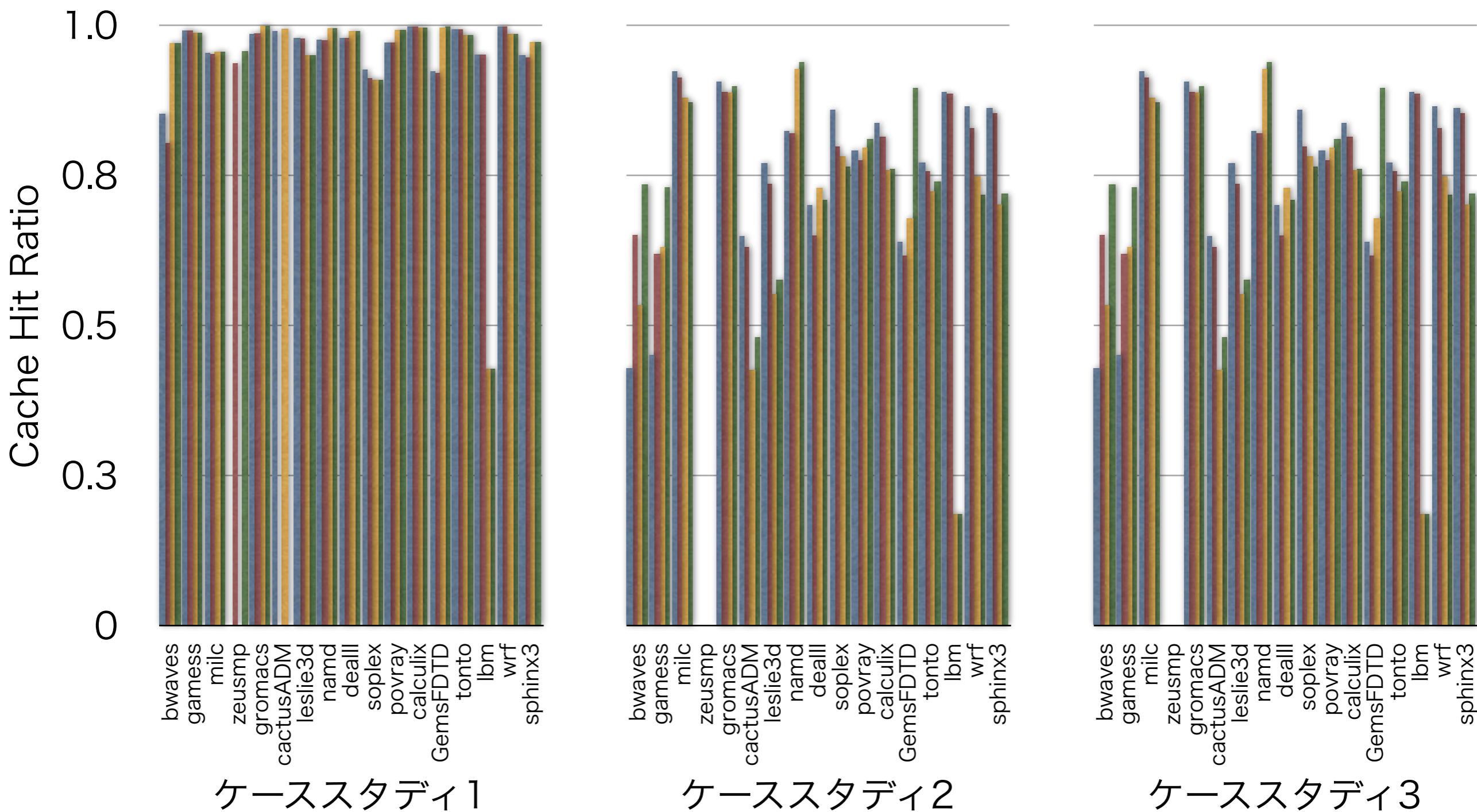
ケーススタディ3

IPC/FP2006



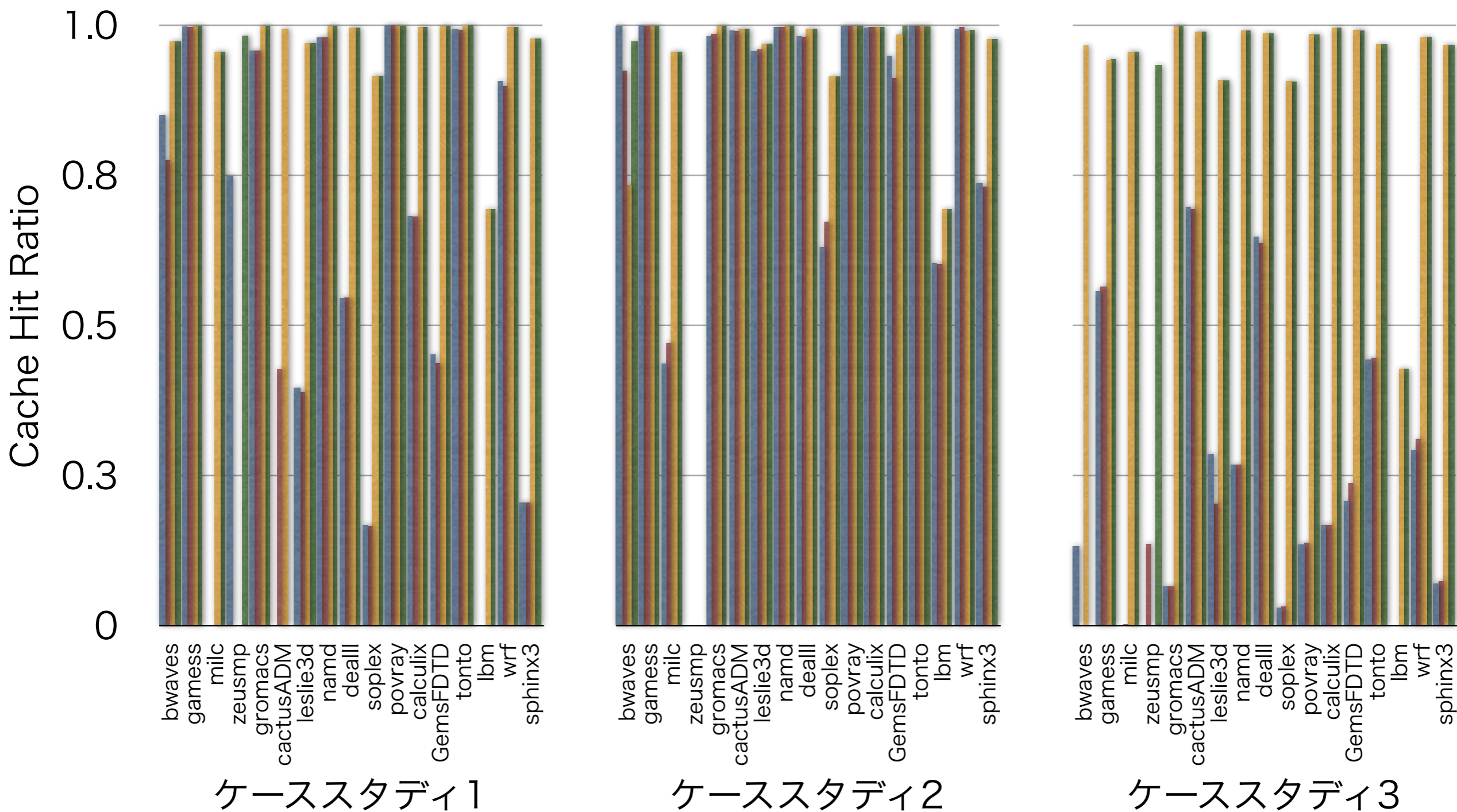
キャッシュヒット率(L1D)/FP2006

■ out-of-order read ■ in-order read
■ out-of-order write ■ in-order write



キャッシュヒット率(L2)/FP2006

■ out-of-order read ■ in-order read
■ out-of-order write ■ in-order write

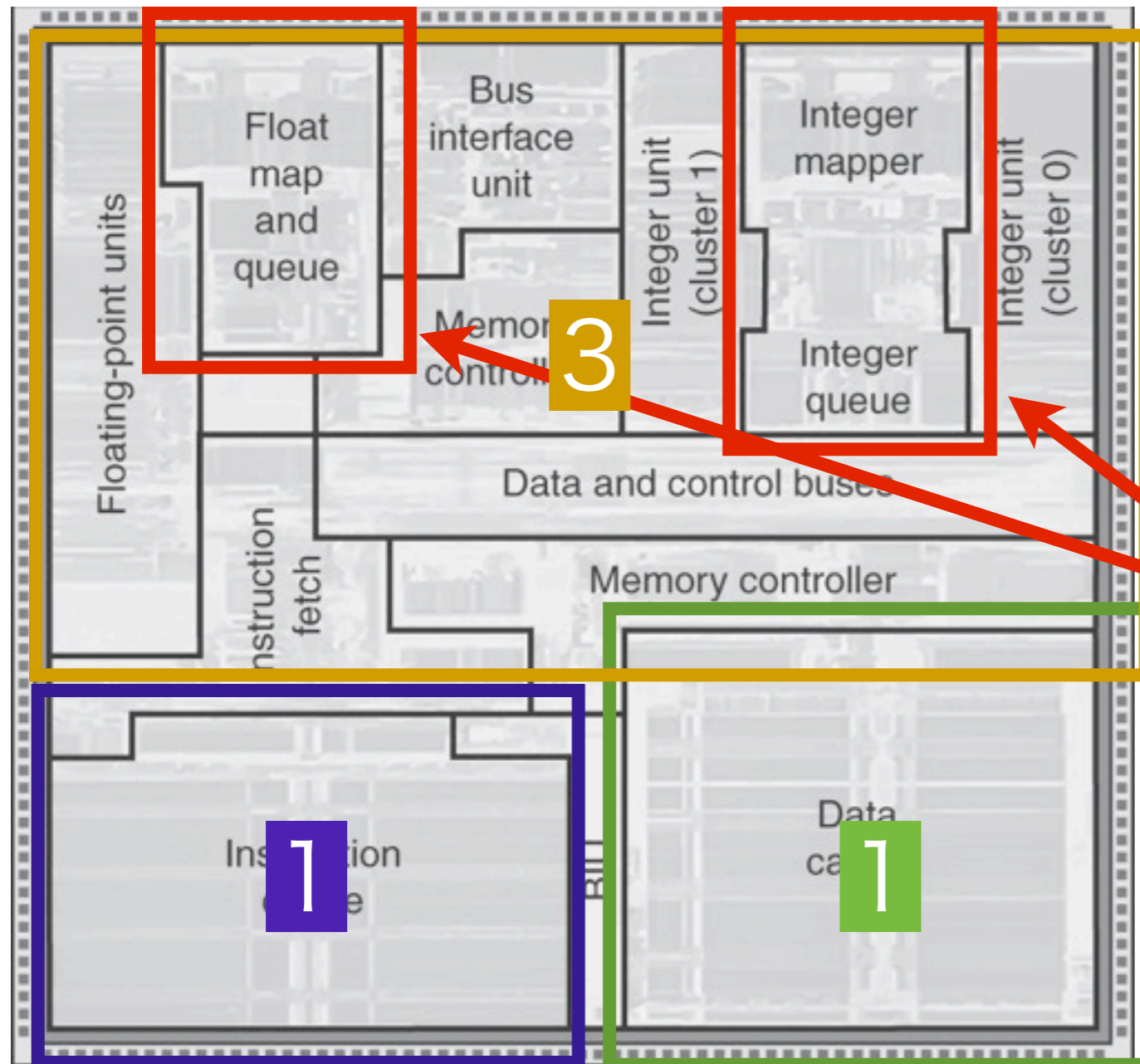


アウトオブオーダー v.s. インオーダー

キャッシュ:演算器 = 1:3

インオーダー:アウトオブオーダー = 2:3

} と仮定してみる

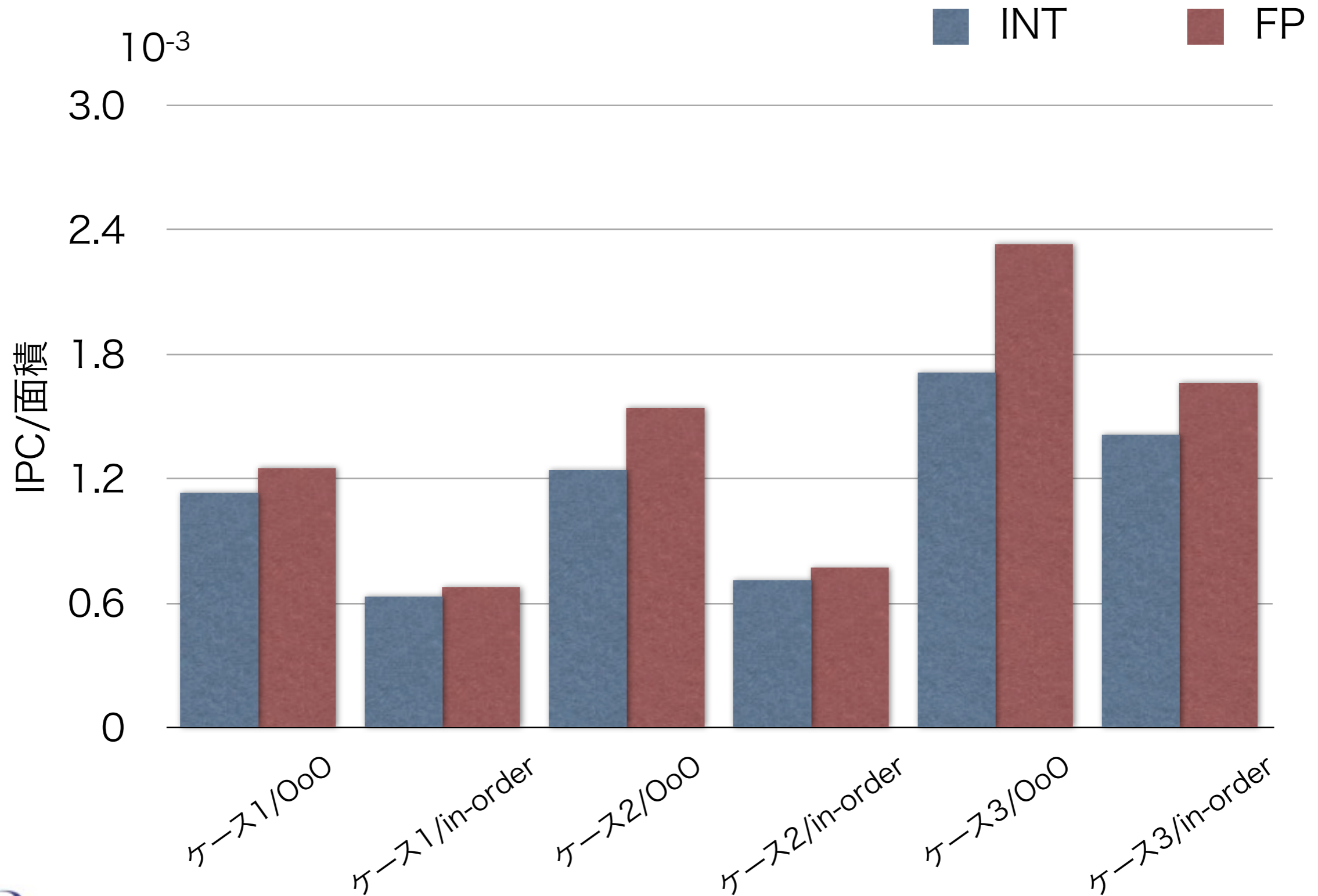


[12] J.L. Hennessy and D.A. Patterson, Computer Architecture, Fourth Edition: A Quantitative Approach, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.

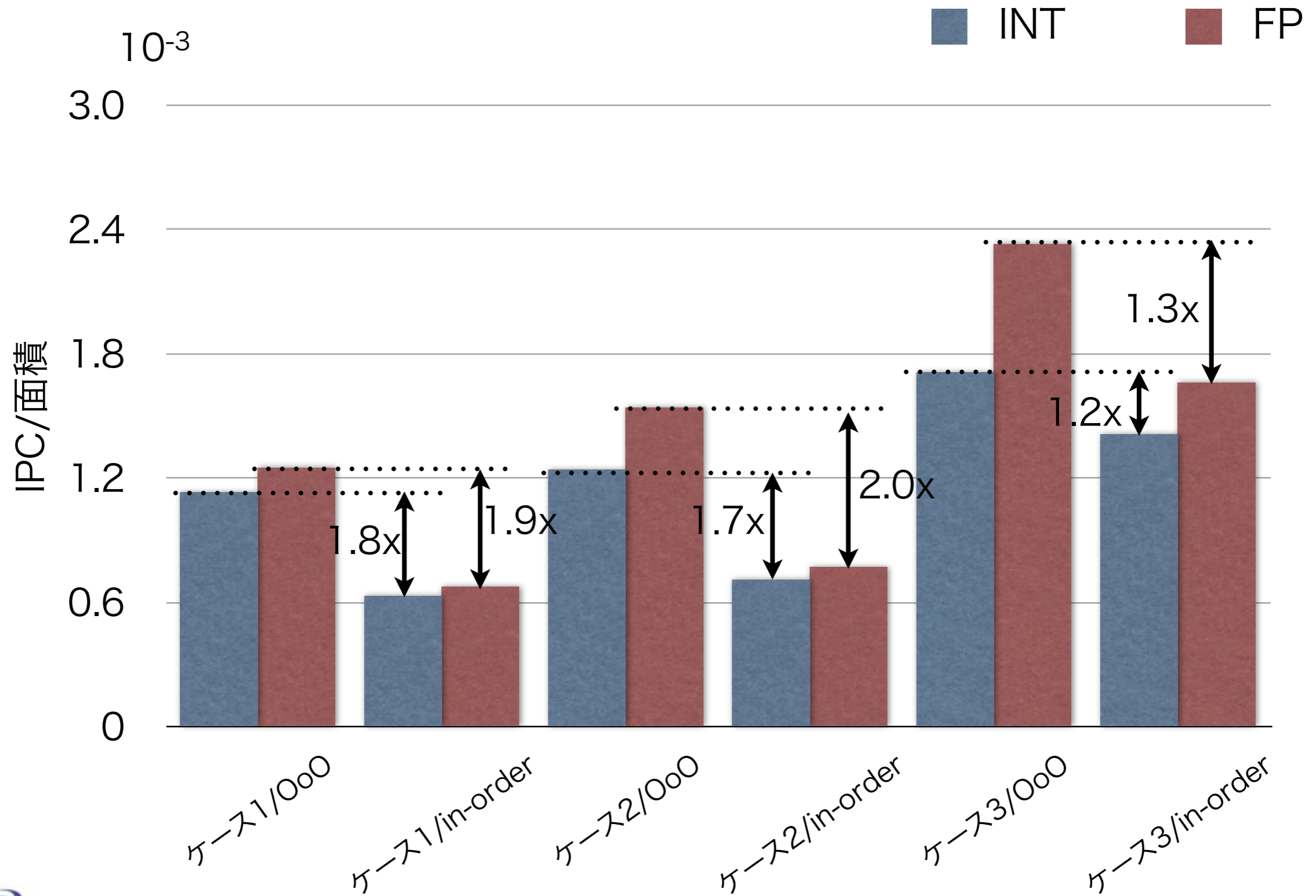
アウトオブオーダー v.s. インオーダー

	面積(\$換算[KB])	IPC	
		INT	FP
ケース1/OoO	768	0.870	0.962
ケース1 /in-order	704	0.443	0.474
ケース2/OoO	480.5	0.597	0.740
ケース2/in-order	416.5	0.295	0.320
ケース3/OoO	248	0.424	0.552
ケース3/in-order	184	0.259	0.304

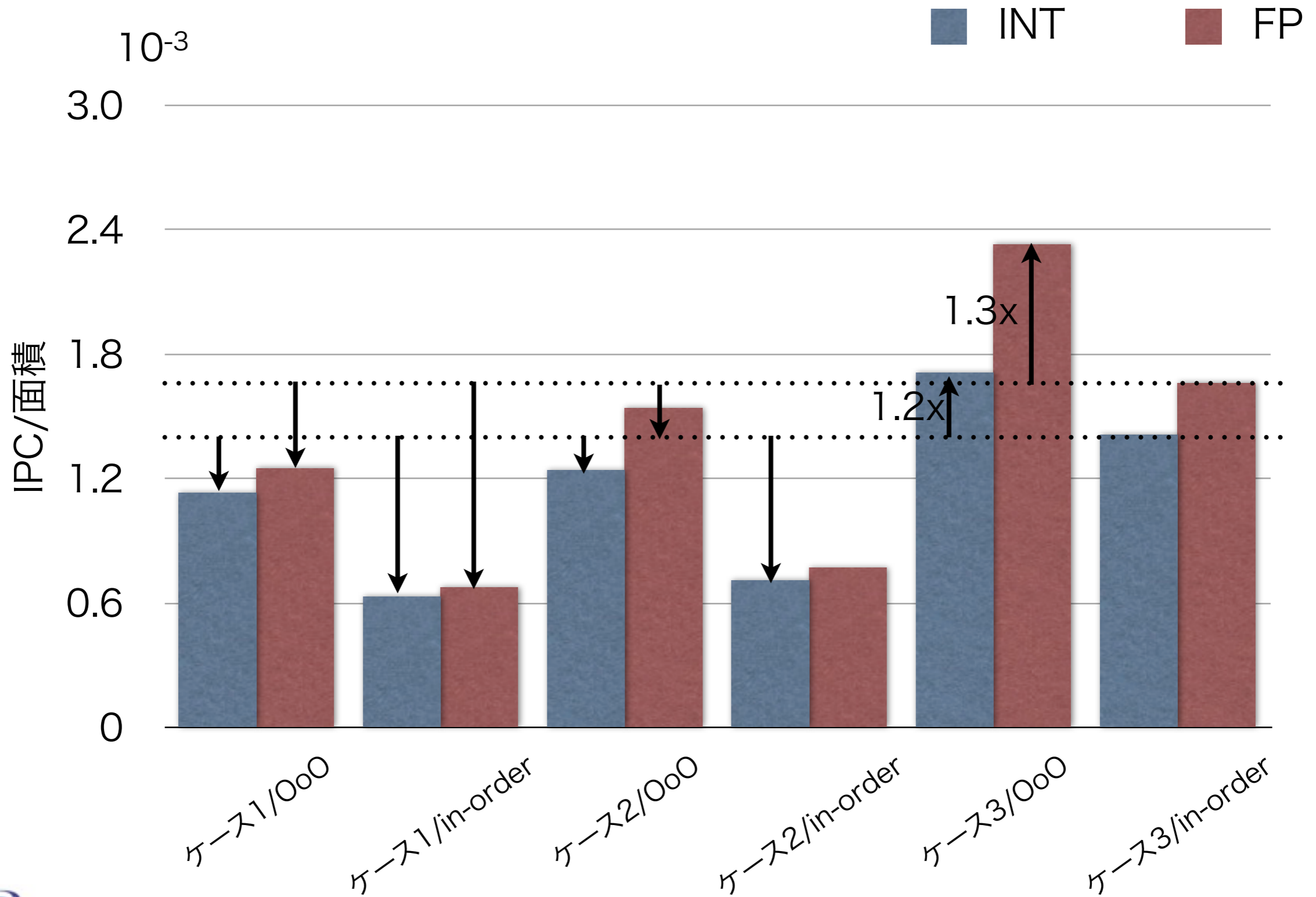
アウトオブオーダー v.s. インオーダー



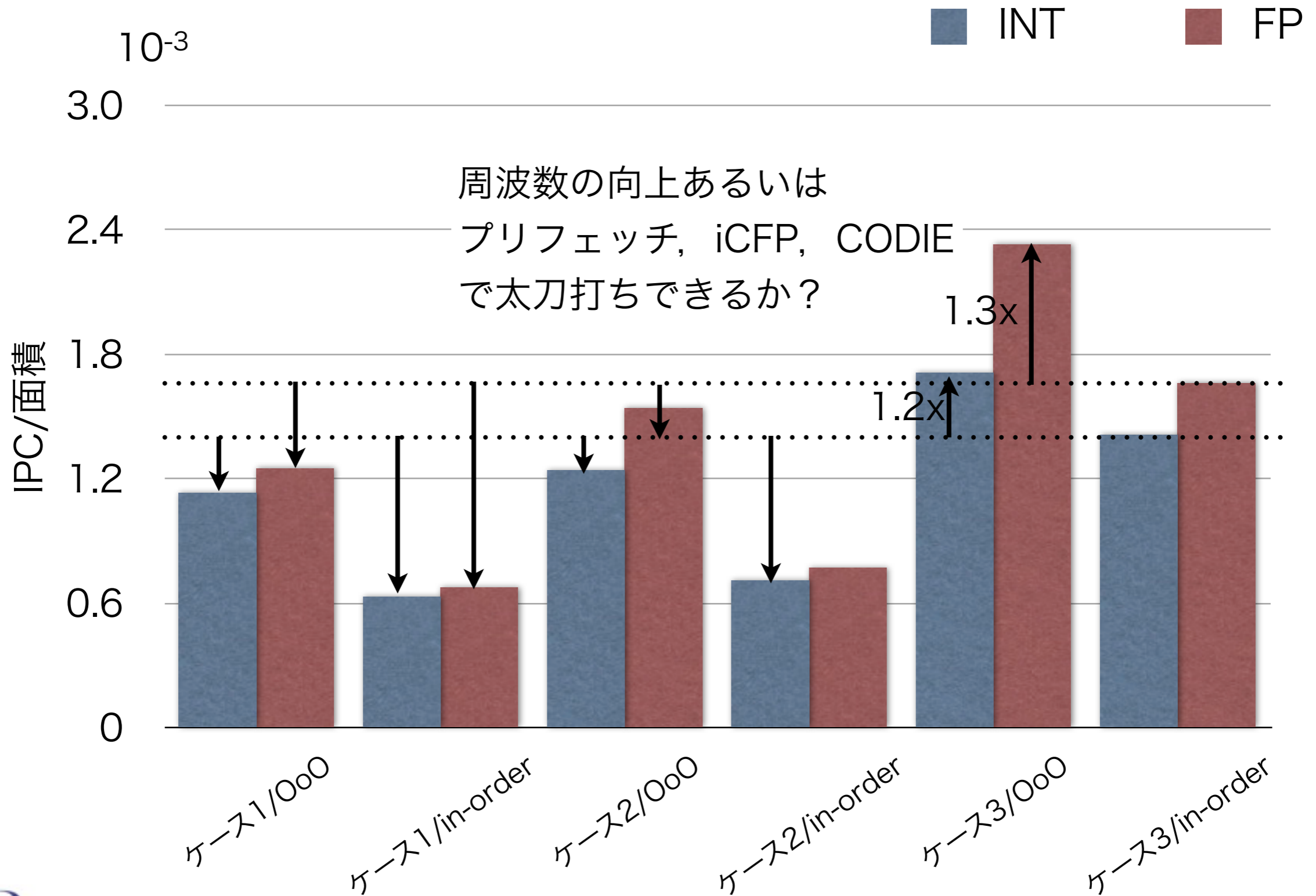
アウトオブオーダー v.s. インオーダー



アウトオブオーダー v.s. インオーダー



アウトオブオーダー v.s. インオーダー



まとめ

- ▶ マルチ/メニーコアプロセッサ時代のアウトオブオーダー v.s. インオーダーについて議論
- ▶ メモリ階層アーキテクチャの3パターンでIPCおよびキャッシュヒット率を評価
- ▶ ケーススタディ3の構成のインオーダーではIPC/面積で他のケースを凌駕

今後の課題

- ▶ 動作周波数などの考慮
- ▶ 種々の研究成果を考慮した評価
 - ▶ メモリレイテンシ隠蔽手法
 - ▶ プリフェッチ, iCFP, CODIE
 - ▶ アウトオブオーダーアーキテクチャの最適化
- ▶ マルチ/メニーコアプロセッサとしての評価
- ▶ 他のベンチマークでの評価

