Javaによる数値計算のすすめ

九州工業大学　大学院情報工学研究院

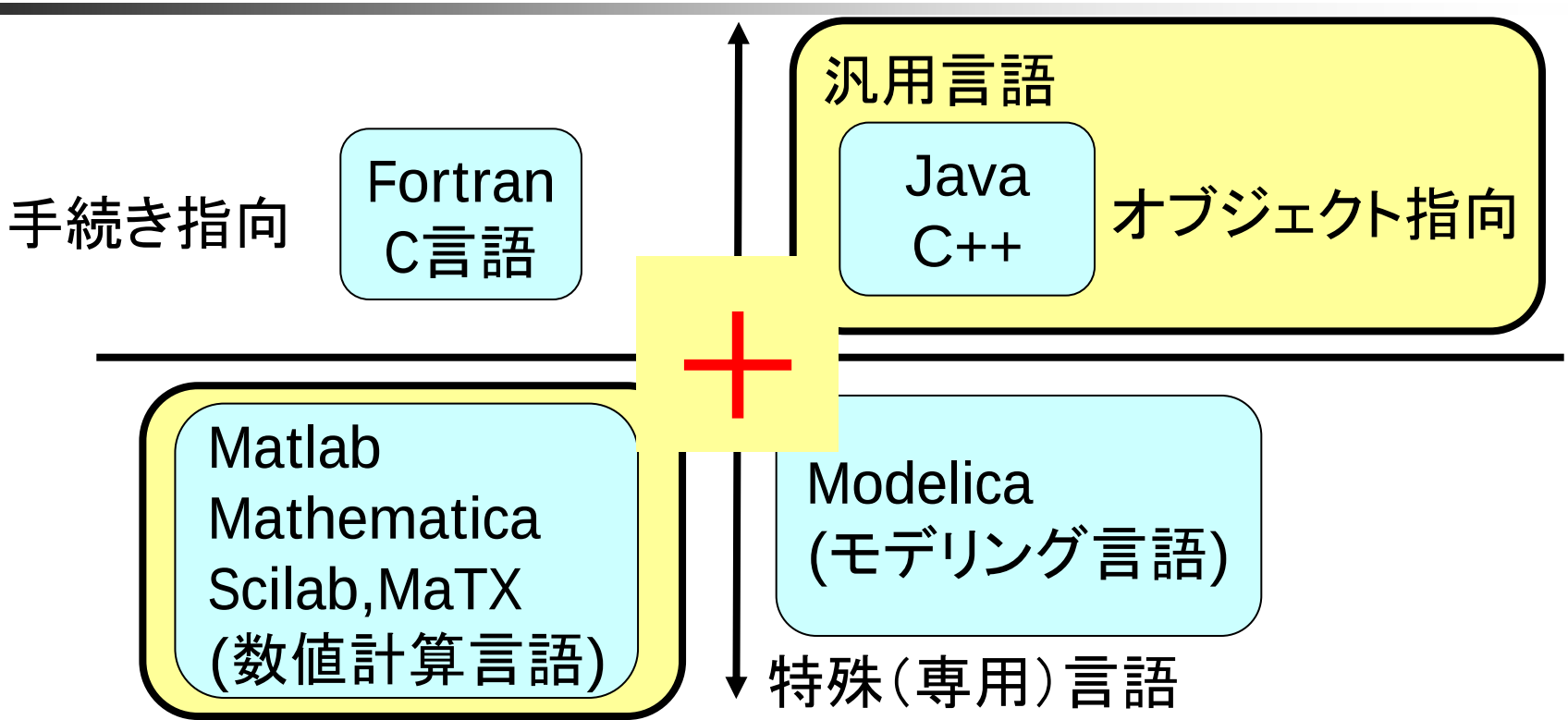
古賀 雅伸

目　次

- はじめに
- Javaは遅い？速い？
  - JavaとCのベンチマーク
  - JITとAOTとLLVM
- Javaによる数値計算に関する情報
  - 書籍（日本語、英語）
  - ライブラリ
- Javaによる数値計算の応用
  - 数値シミュレーション
  - 高品質数値計算
- まとめ

目　次

数値計算に用いるプログラミング言語



- 要求
  - 広く普及した標準技術（開発者の確保） ⇒ 汎用言語
  - 大規模・複雑化へ対応 ⇒ オブジェクト指向
  - 可読性・保守性に優れる ⇒ 数値計算言語

目　次

- はじめに
- Javaは遅い？速い？
  - JavaとCのベンチマーク
  - JITとAOTとLLVM
- Javaによる数値計算に関する情報
  - 書籍（日本語、英語）
  - ライブラリ
- Javaによる数値計算の応用
  - 数値シミュレーション
  - 高品質数値計算
- まとめ

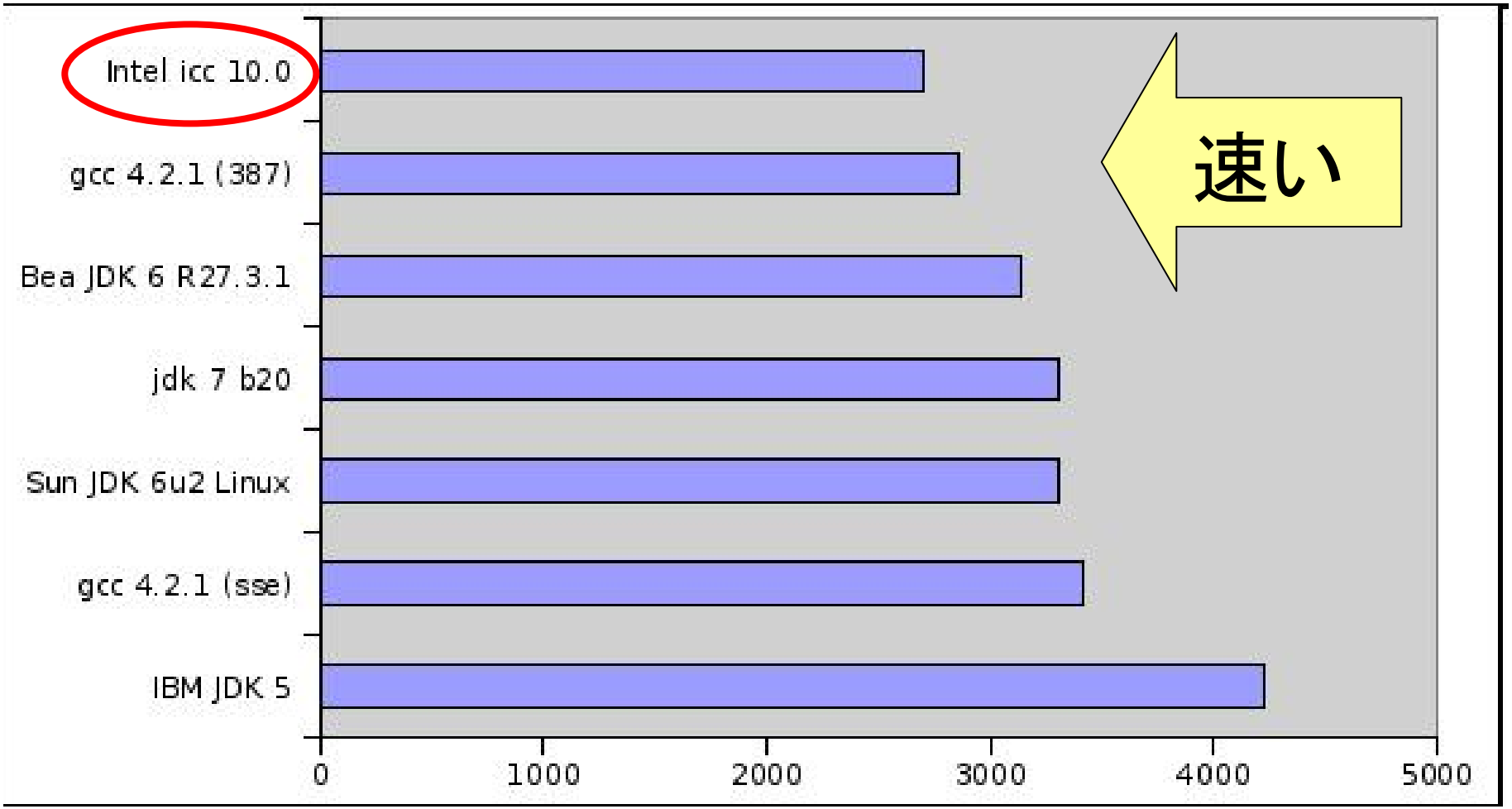
JavaとCのベンチマーク

- Stefan Krause (2009.1.29)
- http://www.stefankrause.net/wp/?p=4
- 問題(入出力無し、文字列処理無し)
  - Mandelbrot, NBody, Spectralnorm (実数)
  - Frannkuch (整数)
- 10回測定した平均
- Javaでは2回目以降使用(JITの起動コスト無し)
- 実行環境
  - CPU: Intel Core 2 Duo 2GHz, RAM: 2GB
  - OS: Ubunto 7.04 32Bit

Java環境とC環境

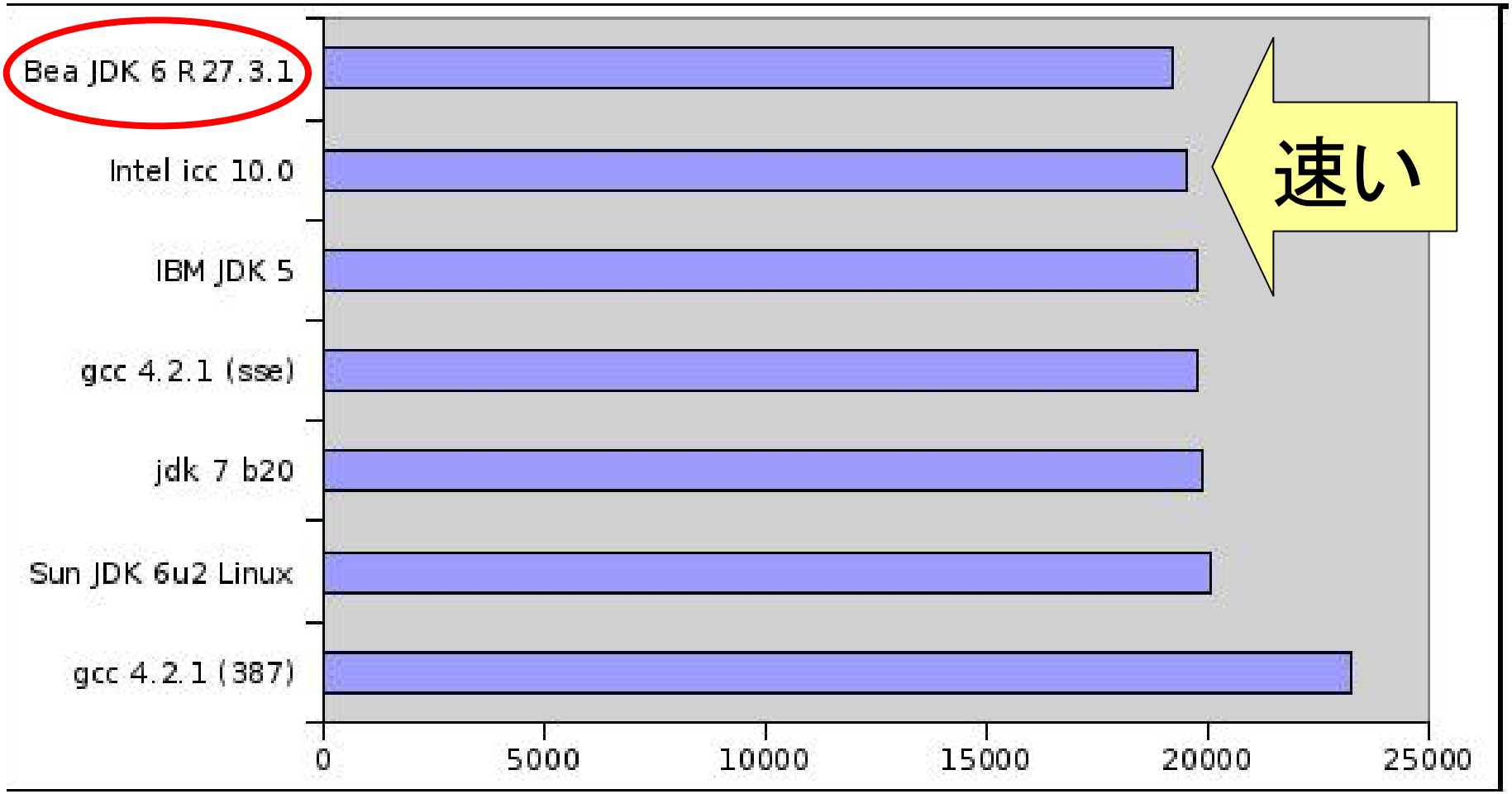
- Java
  - Sun JDK 6u2, Sun JDK 7b20
  - IBM JDK 5, Oracle JRockit JDK 6 R 27.3.1
  - -serverオプション
- C
  - gcc 4.2.1
    - -mfpmath=387 -march=native -O3 ...
    - -mfpmath=sse -march=native -O3 ...
  - Intels ICC 10.0
    - -xT -fast

Mandelbrot



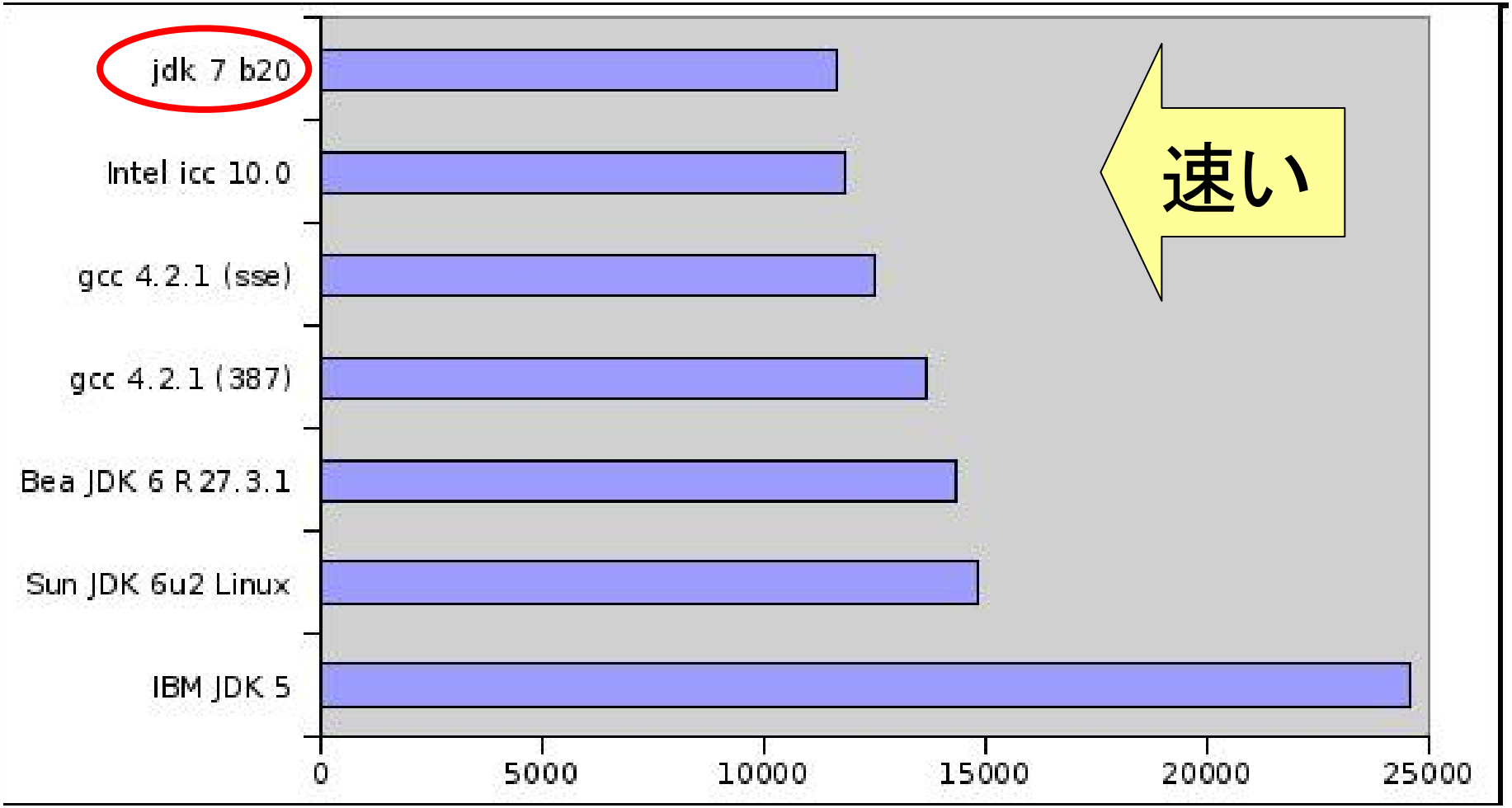
icc > gcc(387) > JRockit > jdk7 > jdk6 > gcc(sse)

Spectalnorm



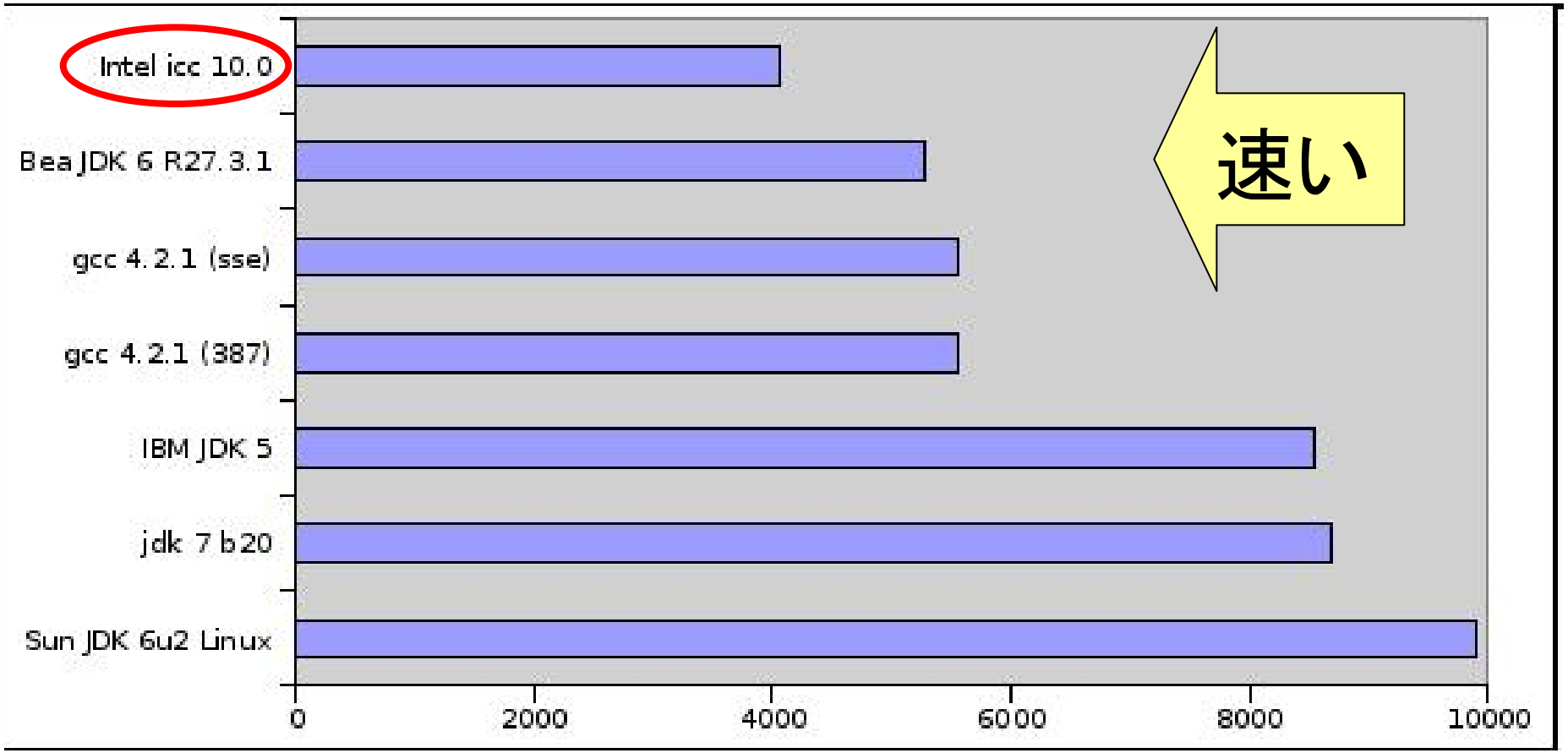
JRockit > icc > IBM jdk5 > gcc(sse) > jdk7 > jdk6

NBody



jdk7 > icc > gcc(sse) > gcc(387) > JRockit > jdk6

Frannkuch



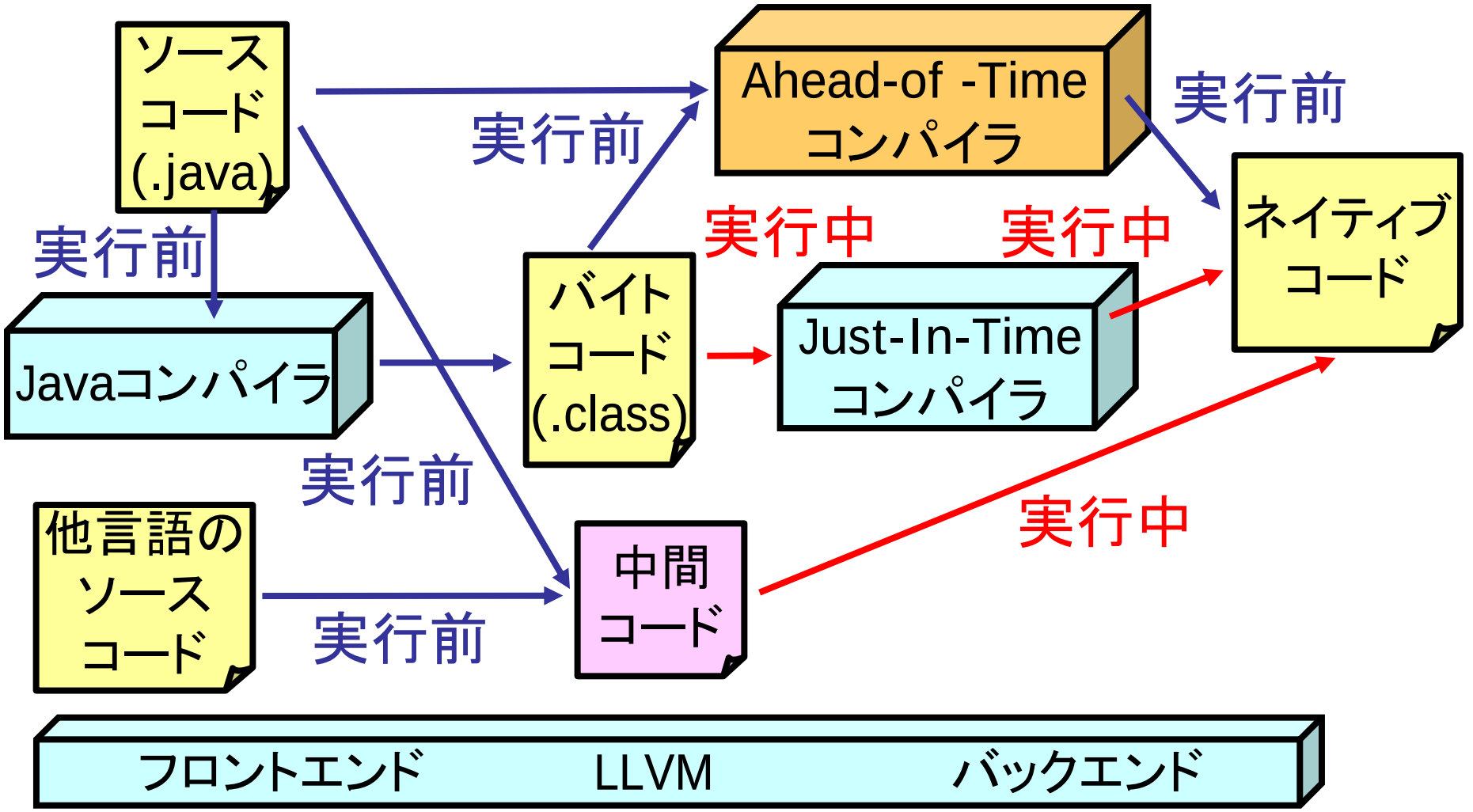
icc > JRockit > gcc(sse) > gcc(387) > IBM jdk > jdk7

ベンチマーク結論

- ICCはGCCより速い
- ICCは無難な選択
- Oracle JRockit JDKは最も早いJDK
- JDK7はJDK6より速い

CとJavaの速さは同程度

JITとAOTとLLVM



参考：http://www.shudo.net/article/Fedora-Core-Expert-200507-GCJ/

JITとAOTとLLVMの比較

- JIT(Just In Time)(実行)コンパイラ
  - Sun JDK
  - Oracle JRockit
  - IBM JDK
  - Apache Harmony
- AOT(Ahead Of Time)（事前）コンパイラ
  - Excelsior JET
  - GNU Compiler for Java
- LLVM(Lower Level Virtual Machine)コンパイラ
  - フロントエンド（中間コード生成）最適化
  - バックエンド（機械コード生成）最適化

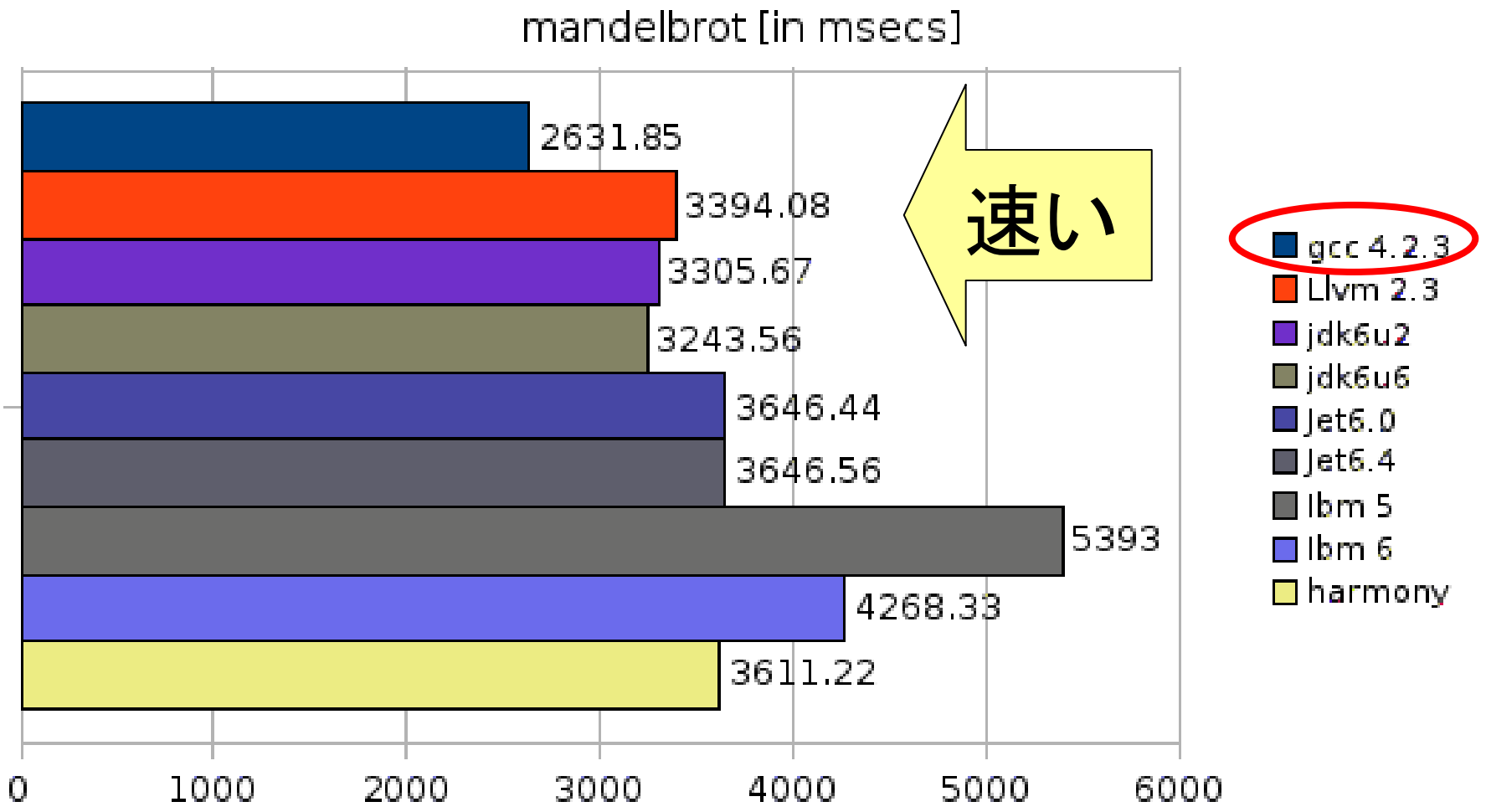
ベンチマーク実行方法

- Stefan Krause (2008.7.7)
- http://www.stefankrause.net/wp/?p=9
- 問題(入出力無し、文字列処理無し)
  - Mandelbrot, NBody,Spectralnorm (実数)
  - Frannkuch (整数)、Himeno
- 10回測定した平均
- Javaでは2回目以降使用(JITの起動コスト無し)
- 実行環境
  - CPU: Intel Core 2 Duo 2GHz, RAM: 2GB
  - OS: Ubunto 7.04 32Bit

実行環境

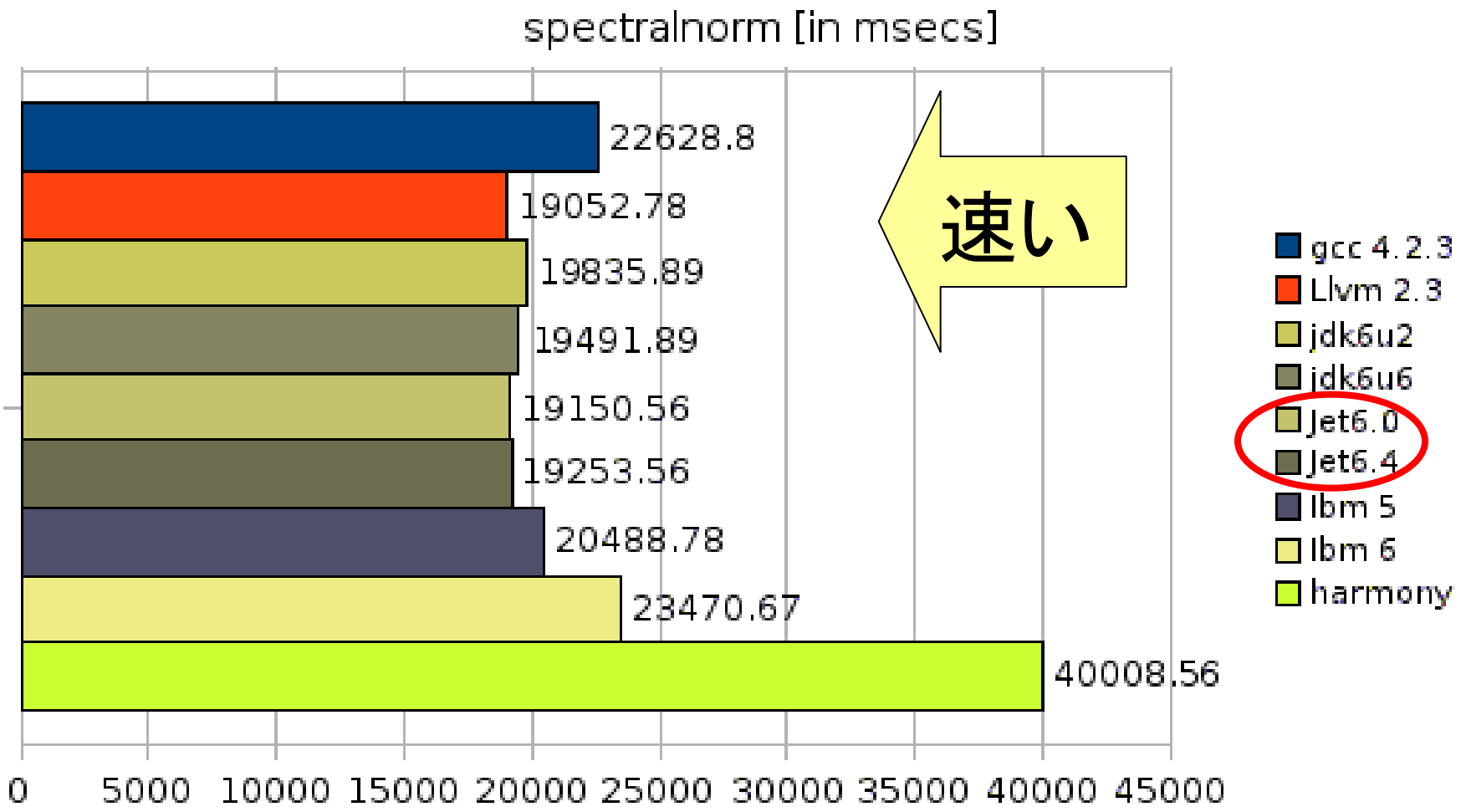
- JIT
  - Sun JDK 6u2, Sun JDK 6u6
  - Apache Harmony m6
  - IBM JDK5, IBM JDK6
  - -serverオプション
- AOT
  - gcc 4.2.3(-O3 -msse2 -march=native =mfpmath=387)
  - Excelsior JET 6.0, JET 6.4
- LLVM
  - LLVM 2.3 (-mcpu=core2 -mattr=sse42)

Mandelbrot



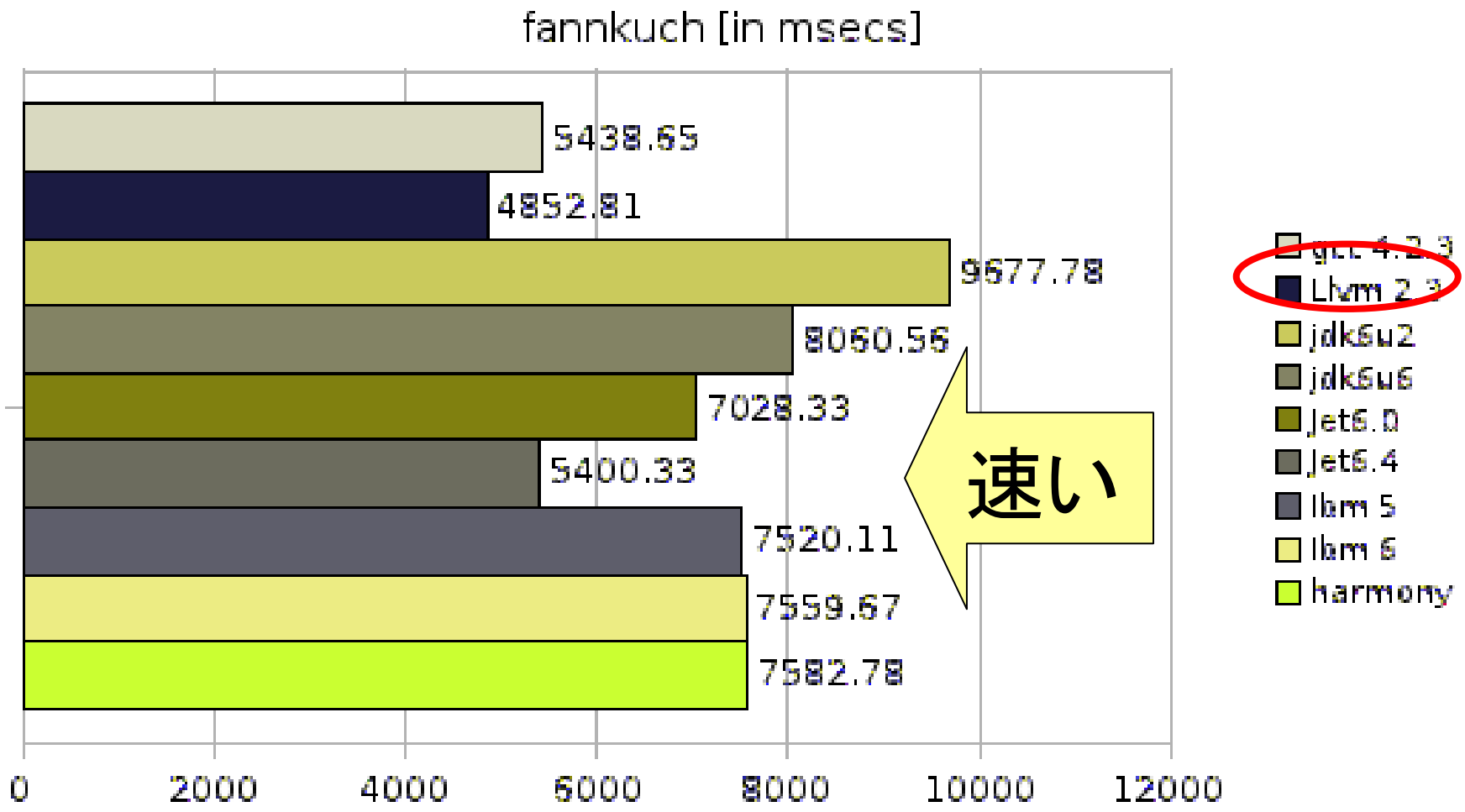
gcj > JDK 6 > LLVM > Harmony > JET > IBM JDK

Spectralnorm



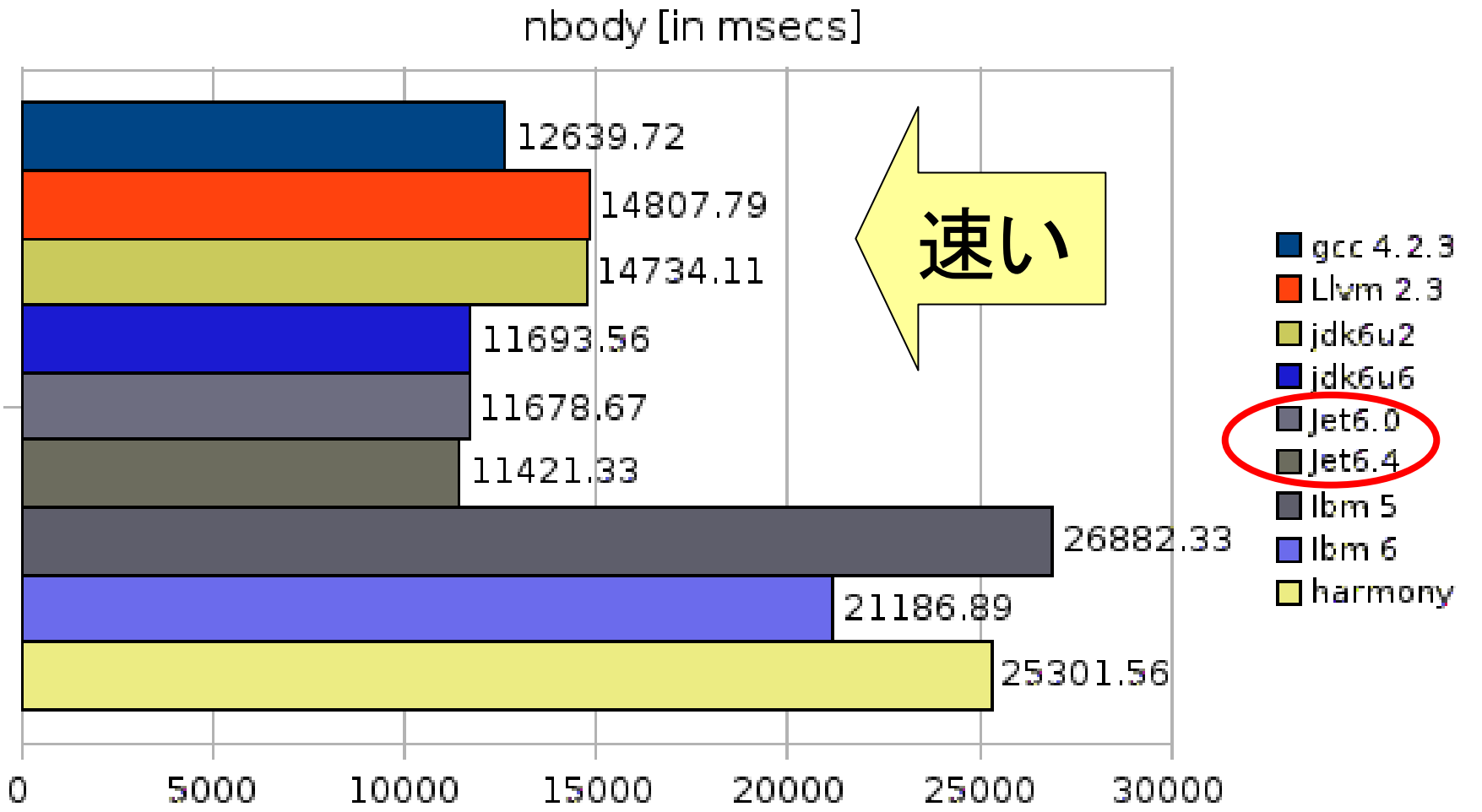
JET > LLVM > JDK 6 > IBM JDK5 > gcj > Harmony

Funnkuch



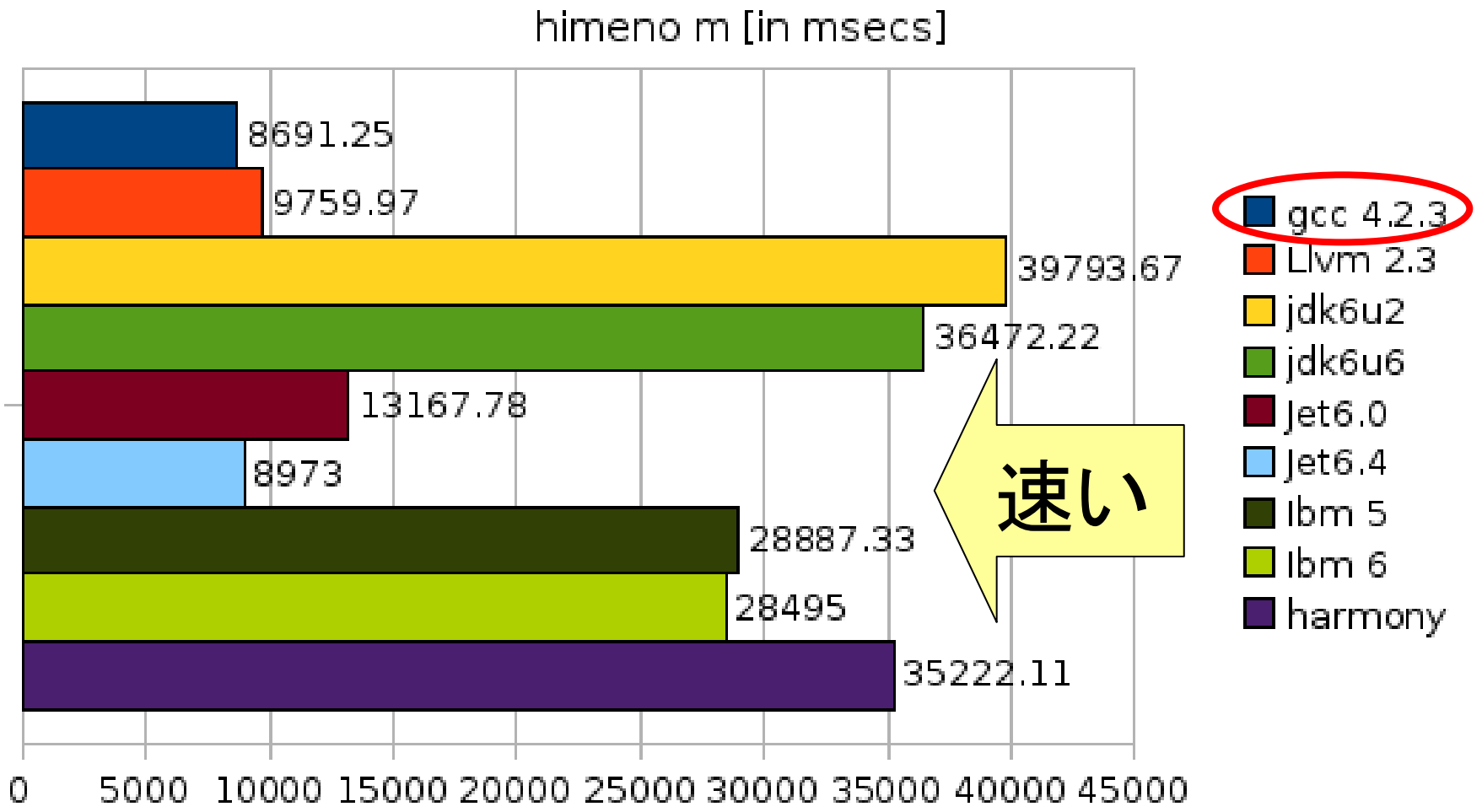
LLVM > JET > gcj > IBM JDK > Harmony > JDK6

NBody



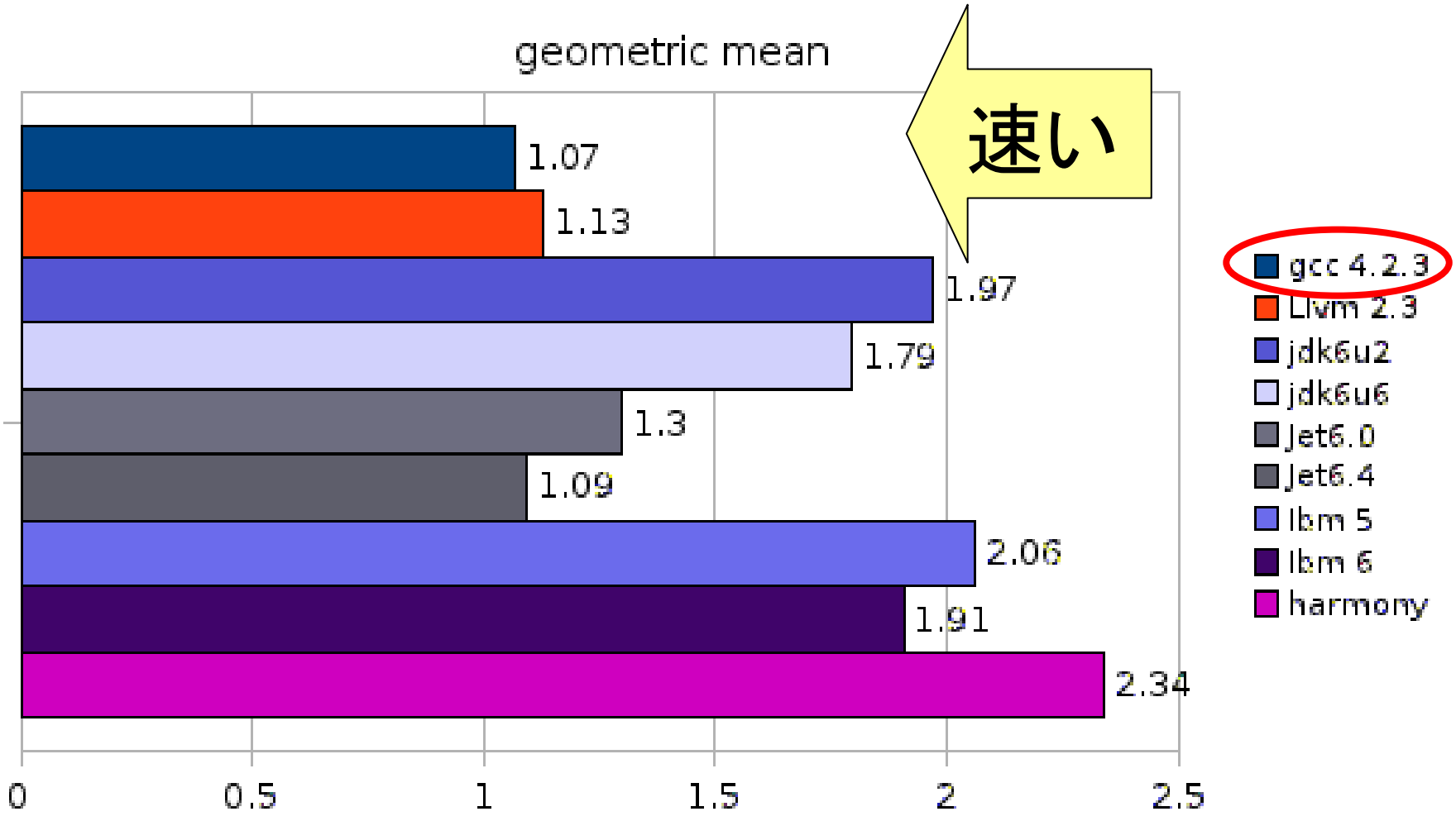
JET > JDK 6 > gcj > LLVM > IBM JDK6 > Harmony

Himeno



gcj > JET > LLVM > IBM JDK6 > Harmony > JDK6

5個のベンチマーク結果の幾何平均



gcj > JET > LLVM > JDK6 > IBM JDK6 > Harmony

ベンチマークの結論

- GCJが一番速い
- JET6.4が二番目に速い
- LLVMは、かなり速い
- Sun JDKには不得意な処理がある
- Harmonyは改善の余地がある

AOTやLLVMは有望である

目　次

- はじめに
- Javaは遅い？速い？
  - JavaとCのベンチマーク
  - JITとAOTとLLVM
- Javaによる数値計算に関する情報
  - 書籍（日本語、英語）
  - ライブラリ
- Javaによる数値計算の応用
  - 数値シミュレーション
  - 高品質数値計算
- まとめ

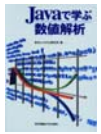
Javaによる数値計算に関する情報（2003年以前）

- Java Numerics
- http://math.nist.gov/javanumerics/
- 1998～2003
- JGF(Java Grande Forum)のNumerics WG
- 情報
  - ベンチマーク
  - ライブラリ
  - ツール、ユーティリティー
  - 参考文献
  - 関連リンク

Javaによる数値計算に関する書籍（日本語）(1/2)

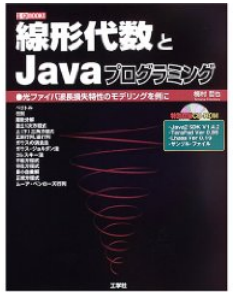
1999年		Java2による数値計算
		CとJavaで学ぶ数値シミュレーション入門
2000年		Javaで学ぶ数値計算
2001年		Javaによる連続体力学の有限要素法
		Javaによる流体・熱流動の数値シミュレーション
2003年		Javaによるオブジェクト指向数値計算法
2004年		理工系のJava
		Javaによる応用数値計算

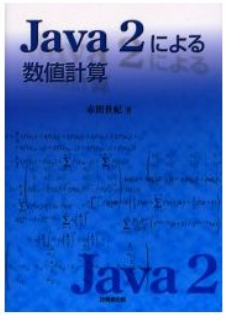
Javaによる数値計算に関する書籍（日本語）(2/2)

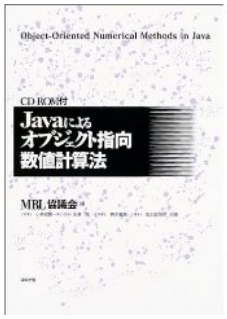
2005年		線形代数とJavaプログラミン
		Javaで初等数学のグラフを描く本
		Javaで学ぶ数値解析
		JavaとExcelで学ぶシミュレーションの基礎
2006年		Javaで学ぶシミュレーションの基礎
2007年		Javaで学ぶ遺伝的アルゴリズ
		CIP法とJavaによるCGシミュレーション

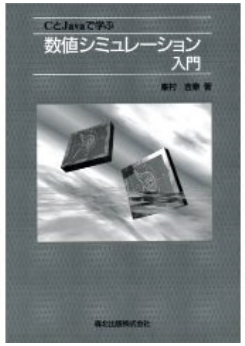
Javaによる数値計算に関する書籍（日本語）

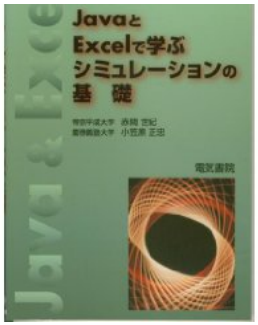
- staticメソッドによる実装
- ベクトルと行列を配列で表現
- 機能不足、拡張性が低い
- 計算と可視化のコードが混在

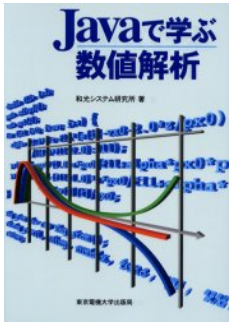




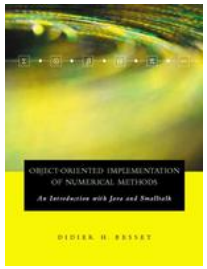
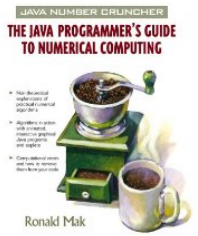
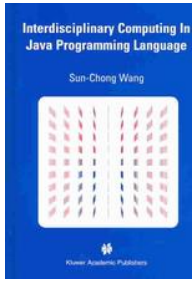




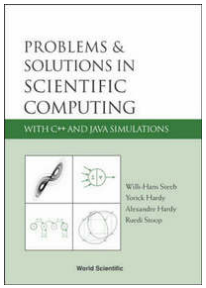
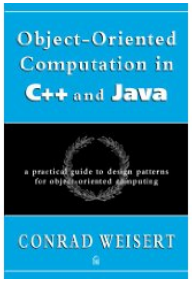
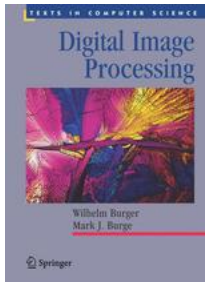




Javaによる数値計算に関する書籍(英語)(1/2)

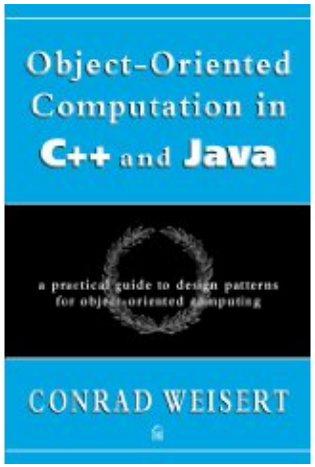
2000年		Object-Oriented Implementation of Numerical Methods An Introduction with Java & Smalltalk
2002年		The Java Programmer's Guide to Numerical Computing
2003年		Interdisciplinary Computing in Java Programming Language
		A Numerical Library in Java for Scientists and Engineers

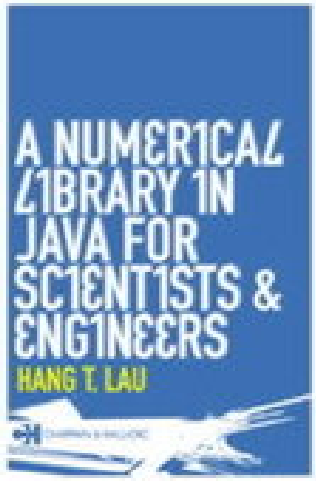
Javaによる数値計算に関する書籍(英語)(2/2)

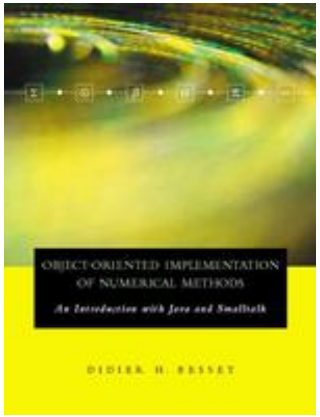
2004年		Problems & Solutions In Scientific Computing With C++ And Java Simulations
2007年		Object-Oriented Computation in C++ And Java
		Digital Image Processing (An Algorithmic Introduction Using Java)

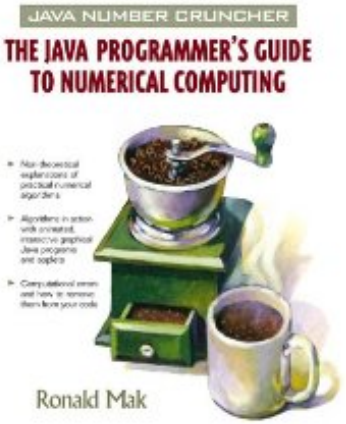
Javaによる数値計算に関する書籍（英語）

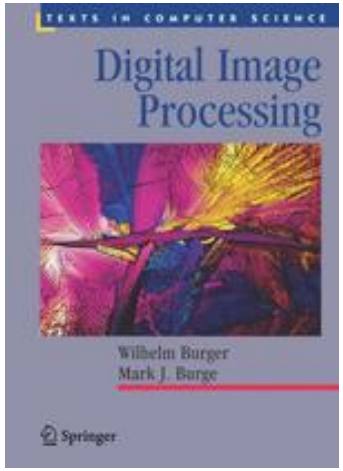
- staticメソッドによる実装
- ベクトルと行列を配列で表現
- 機能不足、拡張性が低い
- 単精度floatのみ

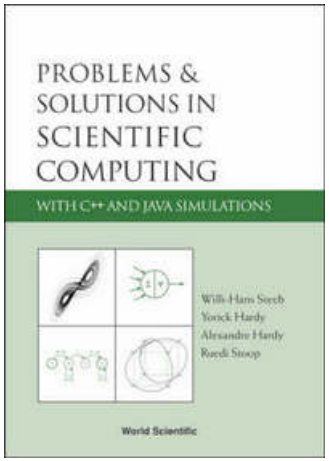












Javaによる数値計算ライブラリ

- 開発が終わった(停止？)ライブラリ
  - Java Numerical Toolkit (～1998.5.6)
  - JavaNumerics (1998～2003)
  - JAMA (A JAva MAtrix Package) (1998～2005.7.13)
  - JUMP (Java Ultimate Math Package) (～2002.6.7)
  - JAMPACK (A JAva Matrix PACKage) (～2005.7.13)
  - JScience (～2007.10.4)
- 開発継続中のライブラリ
  - 数学的構造: JSci、JAS
  - 他言語から変換: netlib-java
  - 商用: JSML
  - 基本数学: Apache Commons Math
  - オブジェクト指向: NFC

Java Numerical Toolkit

- http://math.nist.gov/jnt/
- NIST(米国標準技術局)
- ～1998.5.6
- 特殊関数、非線形方程式
- 線形代数
  - 疎行列
  - 簡単な演算
  - LU分解、QR分解

JAMA (A JAva MAtrix Package)

- http://math.nist.gov/javanumerics/jama/
- 1998.8.5～2005.7.13
- MathWorks社とNIST(米国標準技術局)
- 線形代数用ライブラリ
  - Matrixクラス(実行列のみ、倍精度、四則演算)
  - Cholesky分解(対称正定)
  - LU分解(長方形)、QR分解(長方形)
  - 固有値(対称、非対称)
  - 特異値(長方形行列)

JUMP (Java Ultimate Math Package)

- http://jump-math.sourceforge.net/
- Ernst de Haan
- ～2002.6.7
- 任意精度整数、任意精度実数、有理数

JAMPACK (A JAva Matrix PACKage)

- ftp://math.nist.gov/pub/Jampack/Jampack/AboutJampack.html
- ～2005.7.13
- NIST(米国標準技術局) とMaryland大学
- 線形代数用ライブラリ
  - 複素数行列のみ
  - ピボット付きLU分解、Cholesky分解、QR分解
  - 固有値(対称、一般)、特異値
  - Hessenberg形式、Schur分解

JScience

- http://jscience.org/
- ～2007.10.4
- JSR-275(javax.measure)のリファレンス実装
- 数学的構造(群、環、体、ベクトル空間)
- 複素数、有理数、多項式、有理多項式
- 任意精度整数、任意精度実数
- 複素行列、疎密行列、LU分解

JSci

- http://jsci.sourceforge.net/
- JSci e-group
- 数学的構造(群、環、体、ベクトル空間)
- 非線形方程式、統計、ウェーブレット、作図
- 行列
  - 密疎行列、実行列のみ
  - 四則演算、LU分解、Cholesky分解、QR分解
  - 固有値、特異値

JAS (Java Algebra System)

- http://krum.rz.uni-mannheim.de/jas/
- 代数計算用ライブラリ
- タイプセーフ(ジェネリクス)、マルチスレッド対応
- 可換で可解な多項式クラス
- 四則演算、最大公約数、乱数

netlib-java

- http://code.google.com/p/netlib-java/
- netlib(blas, eispack, lapackなど)をF2Jで変換
- 行列
  - 実数行列のみ、密疎行列、圧縮行列
  - LU分解、QR分解、Cholesky分解
  - 固有値、特異値

Apache Commons Math

- http://commons.apache.org/math/
- 数学の基本的ライブラリ
- データ生成、複素数、統計、最適化、GA
- 非線形方程式、常微分方程式
- 線形代数
  - 実行列のみ、ブロック行列、疎行列
  - LU分解、QR分解、Cholesky分解(対称のみ)
  - 固有値(対称のみ)、特異値

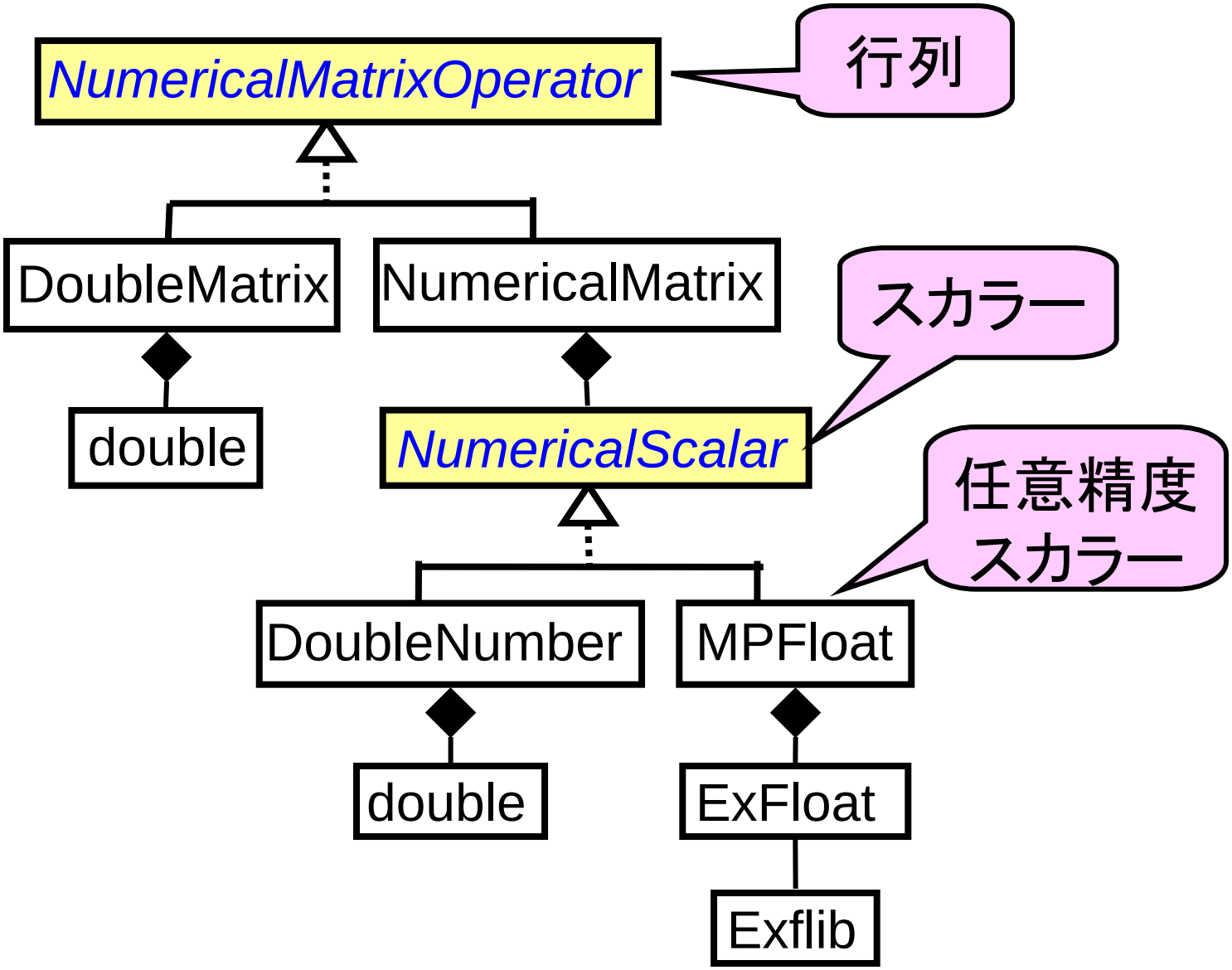
JMSL

- http://www.vnij.com/products/imsl/jmsl/jmsl.html
- Visual Numerics社の製品
- 数値計算ライブラリIMSLのJava版
- 多くの機能はstaticメソッド実装
- 常微分方程式、FFT、補間、非線形方程式
- 線形計画法、回帰分析、フィルタ
- 行列
  - ベクトルと行列データを配列で表現
  - 疎行列、複素行列、倍精度
  - 四則演算、LU分解、QR分解、Cholesky分解
  - 固有値、特異値

NFC(Numerical Foundation Classes)

- http://jamox.mklab.org/
- 九州工業大学　古賀研究室
- オブジェクト指向設計
- 日本語Javadoc(http://jamox.mklab.org/doc/javadoc/)
- 線形方程式、非線形方程式
- 常微分方程式、FFT、信号処理、制御系設計
- 汎用数値型、複素数、多項式、有理多項式
- 行列
  - 汎用数値型を成分とする行列
  - 四則演算、LU分解、QR分解、Chelesky分解
  - 固有値、一般化固有値、特異値

NFCの汎用数値クラス



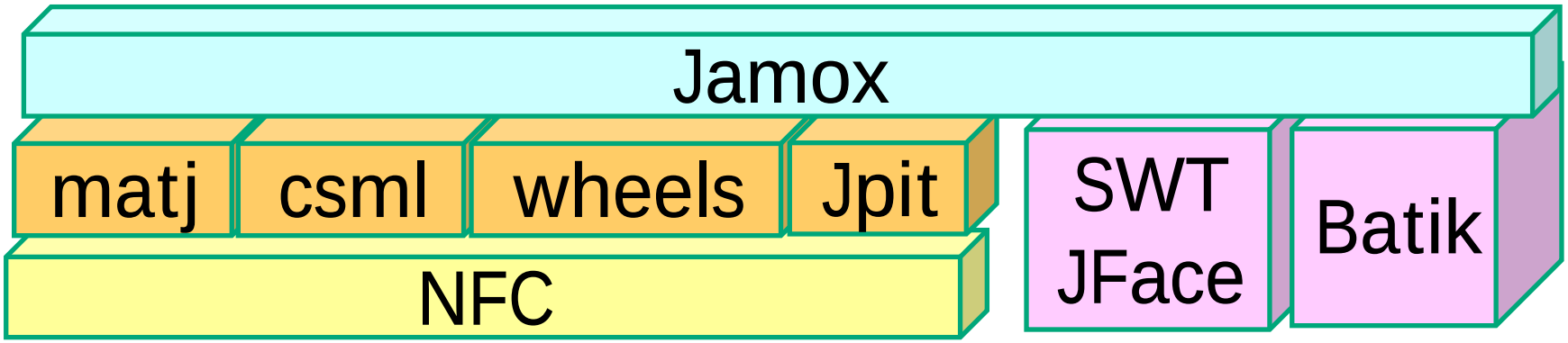
目　次

- はじめに
- Javaは遅い？速い？
  - JavaとCのベンチマーク
  - JITとAOTとLLVM
- Javaによる数値計算に関する情報
  - 書籍（日本語、英語）
  - ライブラリ
- Javaによる数値計算の応用
  - 数値シミュレーション
  - 高品質数値計算
- まとめ

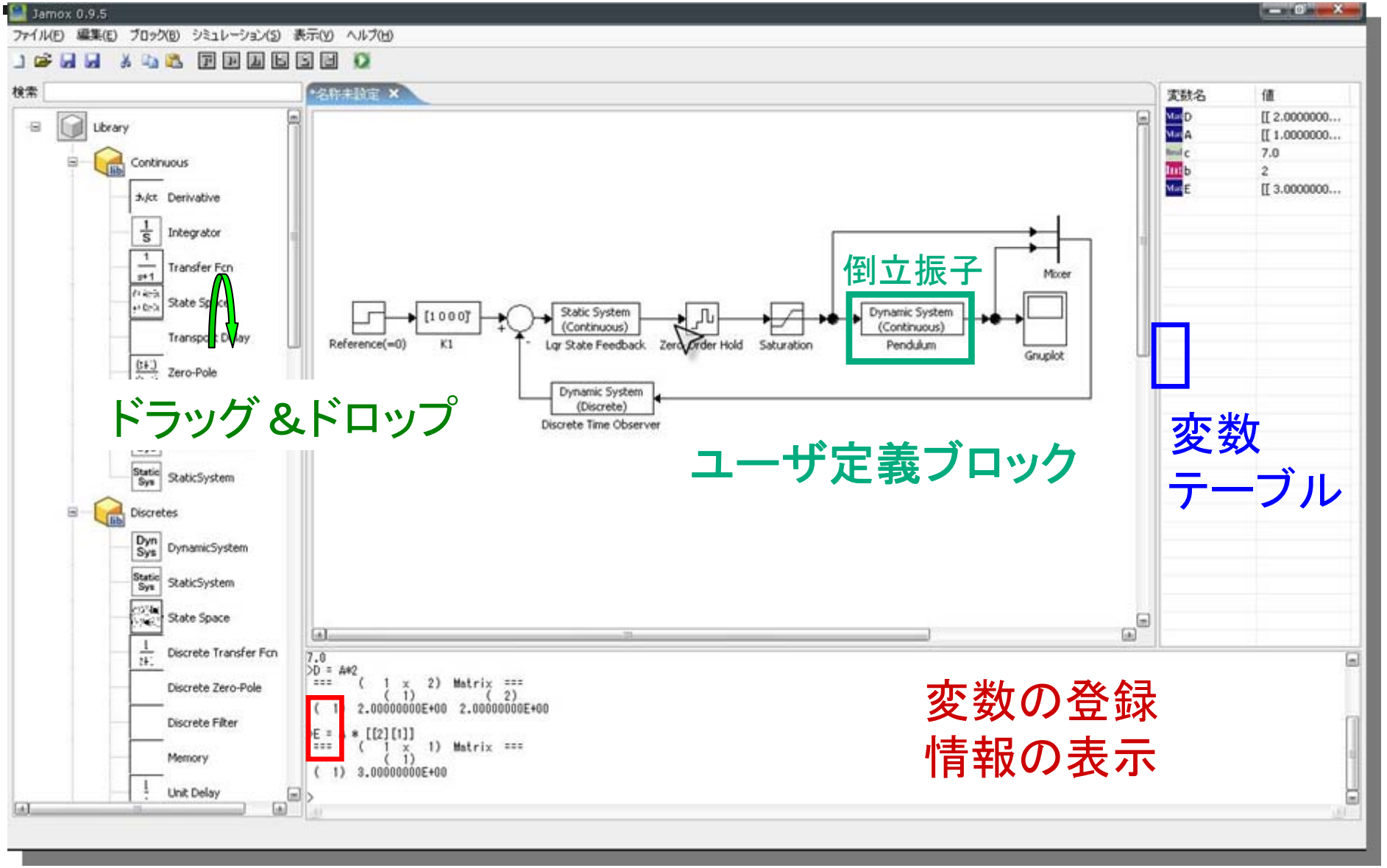
Jamox (http://jamox.mklab.org/)

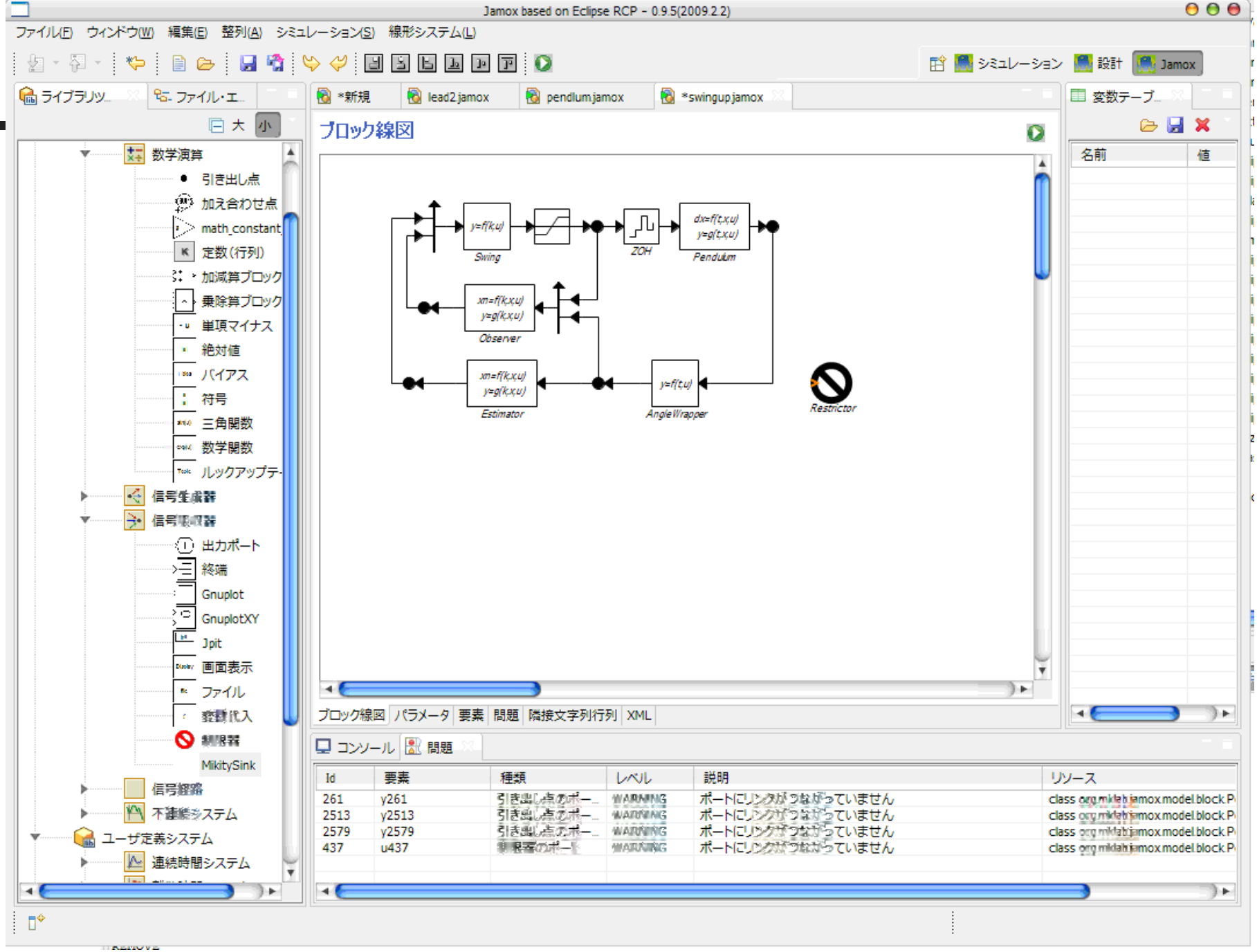
『制御系モデリング・シミュレーションツール』

- モデリング
  - モデル作成GUI（SWT、JFace、Batik）
  - データのXML保存、読込（JAXB）
  - システム解析・設計（Scripting　API）
- シミュレーション
  - 時間応答(ステップ応答等)
  - 周波数応答(ボード線図等）



Jamoxの利用画面


Jamox 0.9.5
ファイル(F) 編集(E) ブロック(B) シミュレーション(S) 表示(V) ヘルプ(H)
検索
Library
Continuous
Derivative
Integrator
Transfer Fcn
State Space
Transport Delay
Zero-Pole
StaticSystem
Discretes
DynamicSystem
StaticSystem
State Space
Discrete Transfer Fcn
Discrete Zero-Pole
Discrete Filter
Memory
Unit Delay
*名称未設定
Reference(=0)
[1 0 0 0]
K1
Static System (Continuous)
Lqr State Feedback
Zero-Order Hold
Saturation
Dynamic System (Continuous)
Pendulum
Mixer
Gnuplot
Dynamic System (Discrete)
Discrete Time Observer
倒立振子
変数名
値
D
A
c
b
E
7.0
2
ドラッグ &ドロップ
ユーザ定義ブロック
変数
テーブル
変数の登録
情報の表示


Jamox based on Eclipse RCP - 0.9.5(2009.2.2)
ファイル(F) ウィンドウ(W) 編集(E) 整列(A) シミュレーション(S) 線形システム(L)
シミュレーション
設計
Jamox
ライブラリツ...
ファイル・エ...
大
小
数学演算
引き出し点
加え合わせ点
math_constant
定数(行列)
加減算ブロック
乗除算ブロック
単項マイナス
絶対値
バイアス
符号
三角関数
数学関数
ルックアップテ
信号生成器
信号[illegible]器
出力ポート
終端
Gnuplot
GnuplotXY
Jpit
画面表示
ファイル
変数代入
[illegible]器
MikitySink
信号経路
不連続システム
ユーザ定義システム
連続時間システム
*新規
lead2.jamox
pendlum.jamox
*swingup.jamox
ブロック線図
y=f(k,u)
Swing
ZOH
dx=f(t,x,u)
y=g(t,x,u)
Pendulum
xn=f(k,x,u)
y=g(k,x,u)
Observer
xn=f(k,x,u)
y=g(k,x,u)
Estimator
y=f(t,u)
AngleWrapper
Restrictor
ブロック線図
パラメータ
要素
問題
隣接文字列行列
XML
変数テーブ...
名前
値
コンソール
問題
Id
要素
種類
レベル
説明
リソース
261
y261
引き出し点のポー...
WARNING
ポートにリンクがつながっていません
2513
y2513
引き出し点のポー...
WARNING
ポートにリンクがつながっていません
2579
y2579
引き出し点のポー...
WARNING
ポートにリンクがつながっていません
437
u437
[illegible]器のポー...
WARNING
ポートにリンクがつながっていません

matj (http://matj.mklab.org/)

『数値計算言語MaTXの処理系』
(http://www.matx.org/)

- JavaCCによる構文定義
- MaTXコードからJavaコード生成
- MaTXコードからCコード生成
- 数値計算エンジン(インタプリタ)
  Scripting API(JSR223)に対応

数値計算言語

- Matlab, Mathematica, Scilab, MaTX
- 行列を言語レベルで扱う事ができる

E = AB + CD　　（数学的表現）

E = A * B + C * D;　　（数値計算言語）

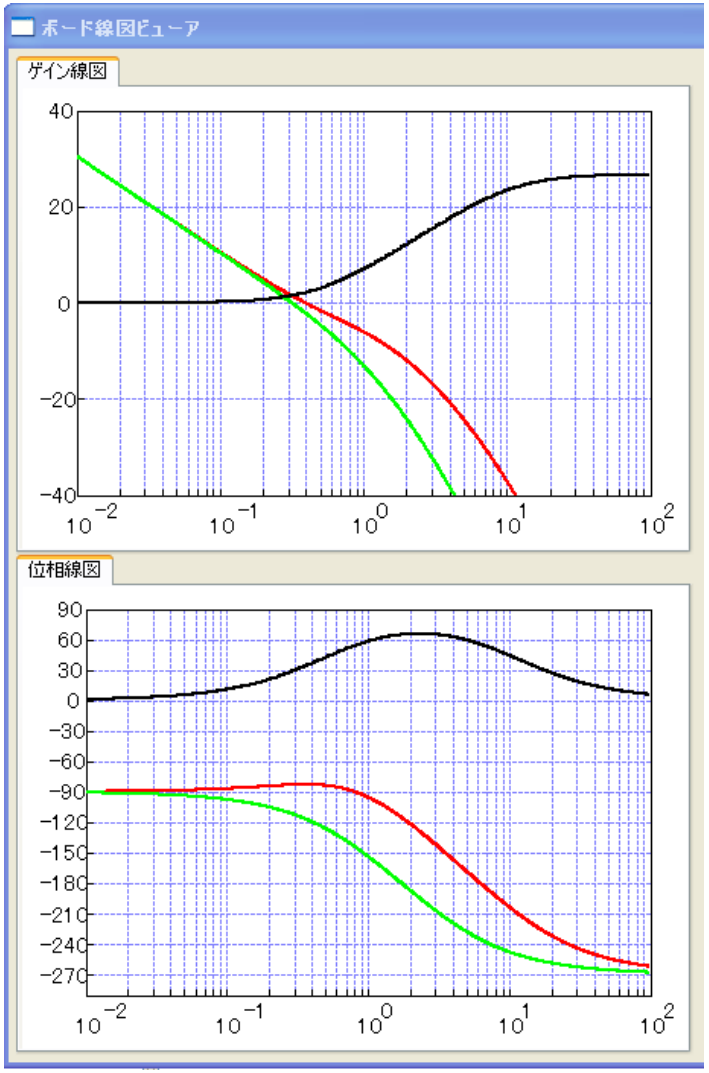
E = A.multiply(B).add(C.multiply(D));（Java言語）

可読性、保守性が高い

Jpit (http://jpit.mklab.org/)

『対話型グラフ描画パッケージ』

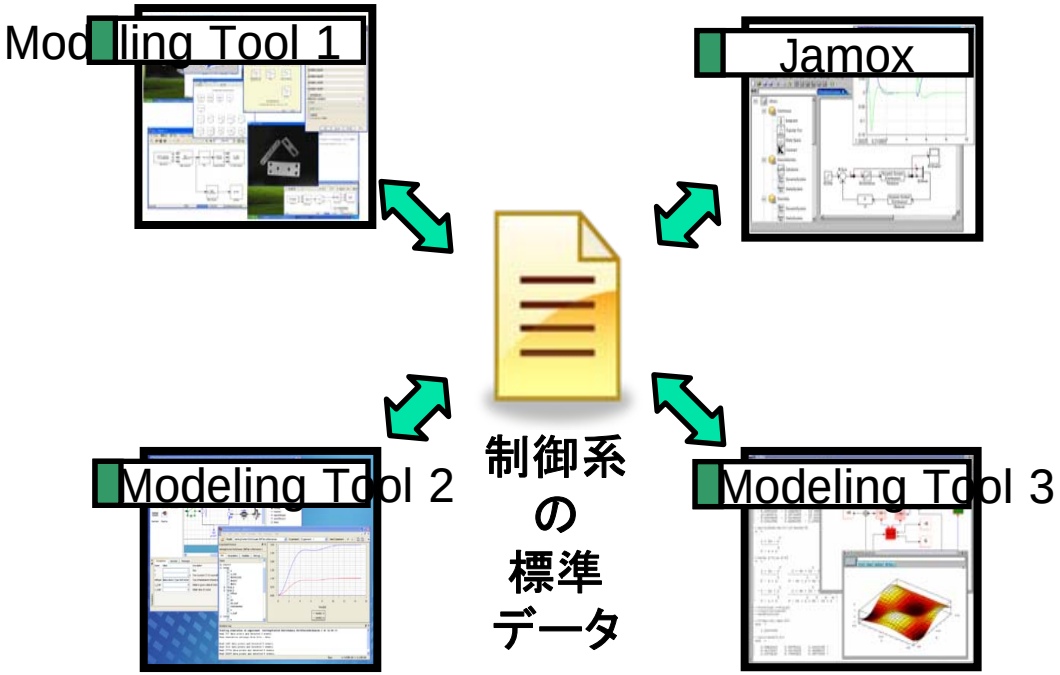
- SWT，JFaceを利用
- 時間応答グラフ
- 周波数応答グラフ
- グラフとパラメータが連動
- Jamoxと連携



CSML (http://csml.mklab.org/)

『制御系モデリング言語パッケージ』

- XML Schemaでデータ構造定義（妥当性検証、拡張性）
- JAXBを用いてアクセスコード生成
- データの相互変換



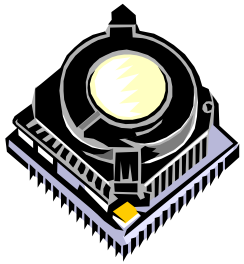
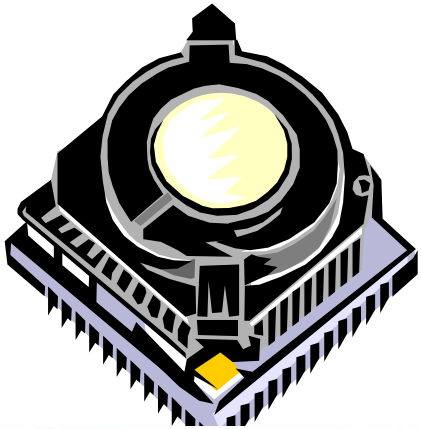
目　次

- はじめに
- Javaは遅い？速い？
  - JavaとCのベンチマーク
  - JITとAOTとLLVM
- Javaによる数値計算に関する情報
  - 書籍（日本語、英語）
  - ライブラリ
- Javaによる数値計算の応用
  - 数値シミュレーション
  - 高品質数値計算
- まとめ

次世代の高品質数値計算

- CPUの高速化



- 利用可能メモリの増大




次世代の数値計算

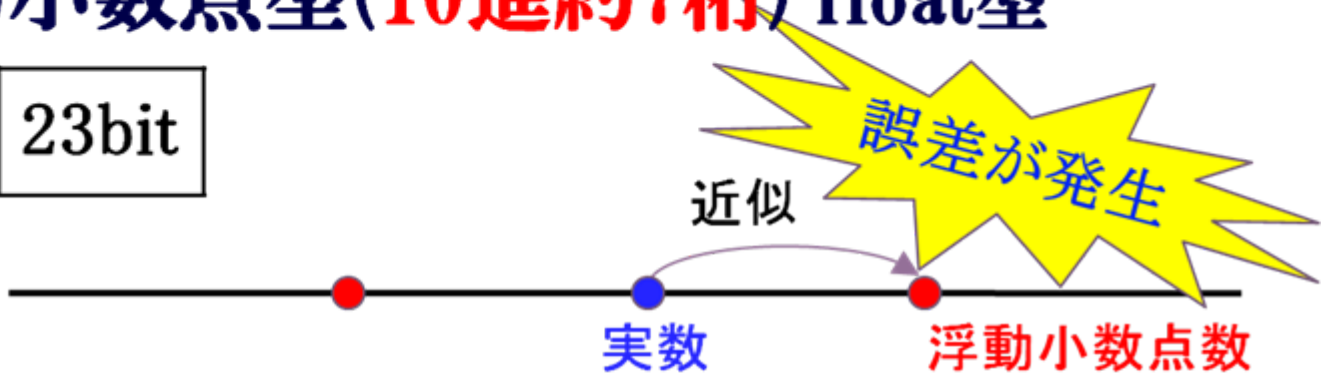
高精度 ＋ 品質保証 ⇒ 高品質数値計算

浮動小数点数の丸め誤差

符号	指数部	仮数部

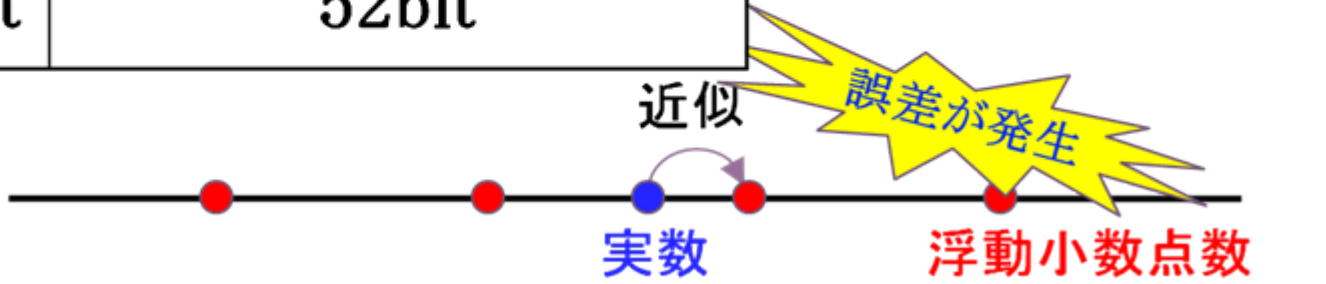
- **単精度浮動小数点型(10進約7桁) float型**

1bit	8bit	23bit



- **倍精度浮動小数点型(10進約16桁) double型**

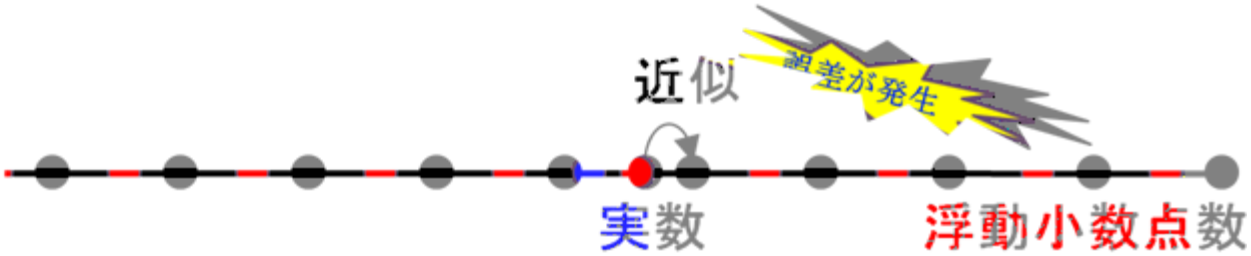
1bit	11bit	52bit



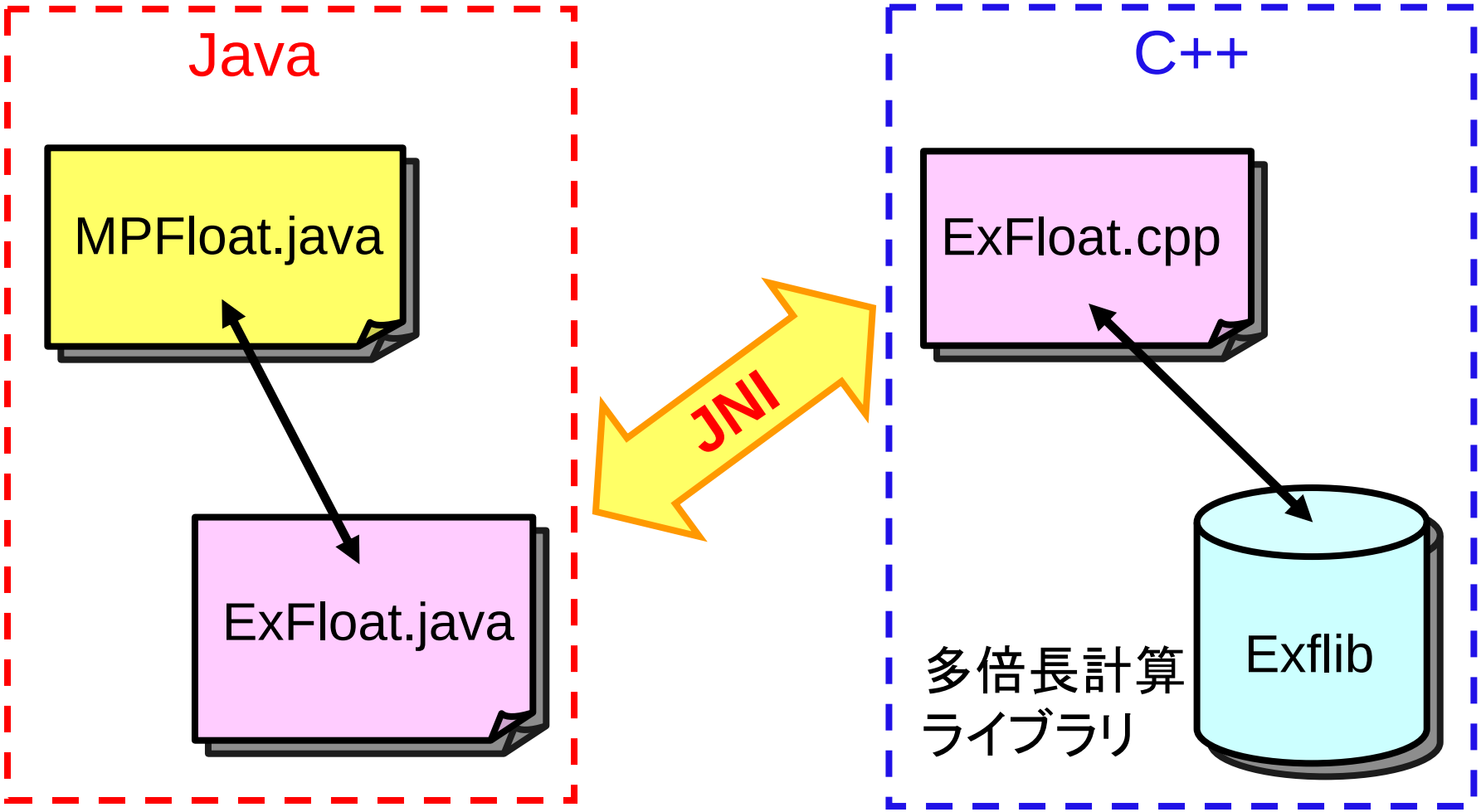
多倍長精度浮動小数点数（任意桁）

1bit	α bit	β bit

- 指数部と仮数部が任意ビット長
  - 仮数部を大きくすることで、誤差の影響を低減

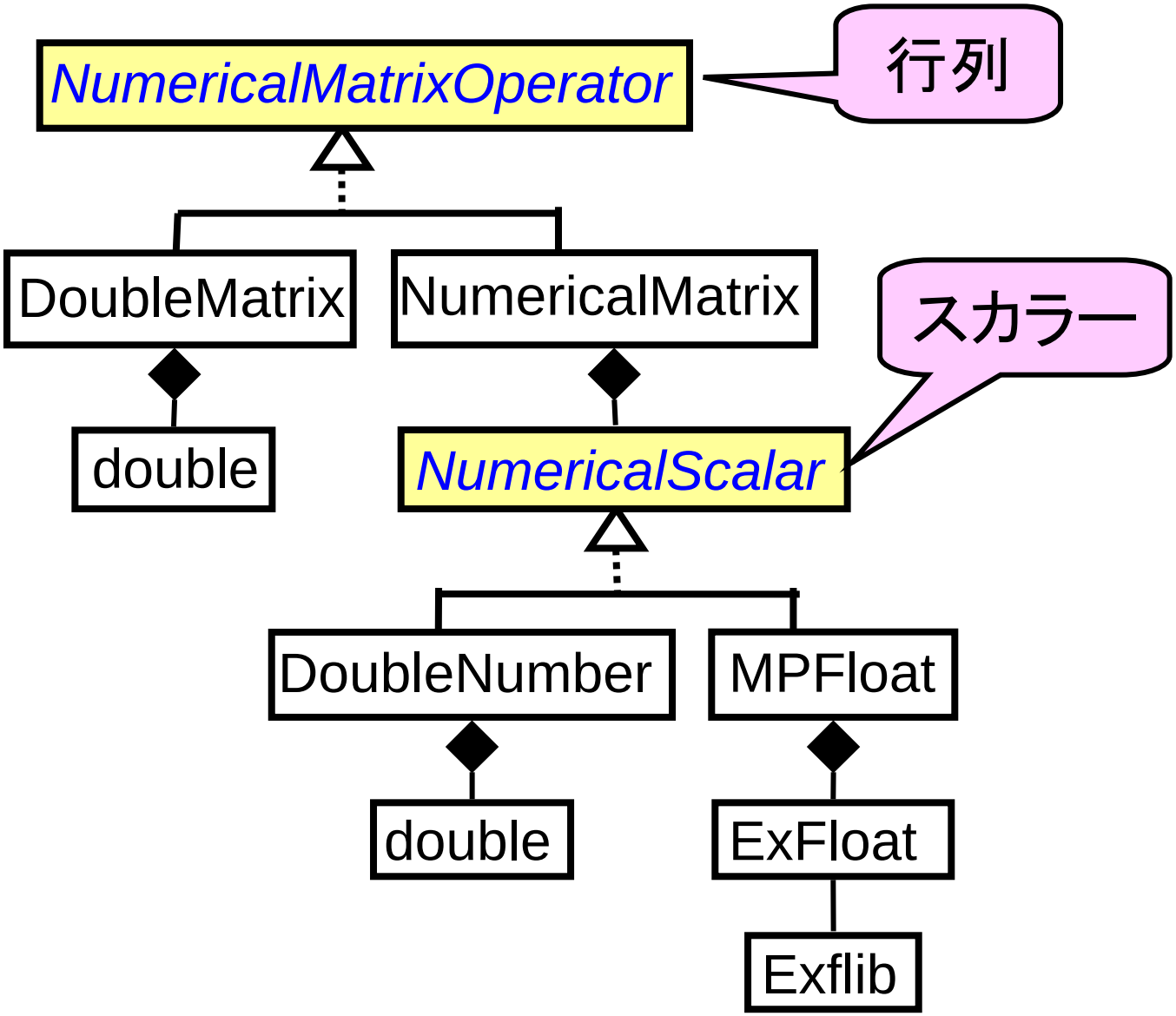


MPFloatパッケージ



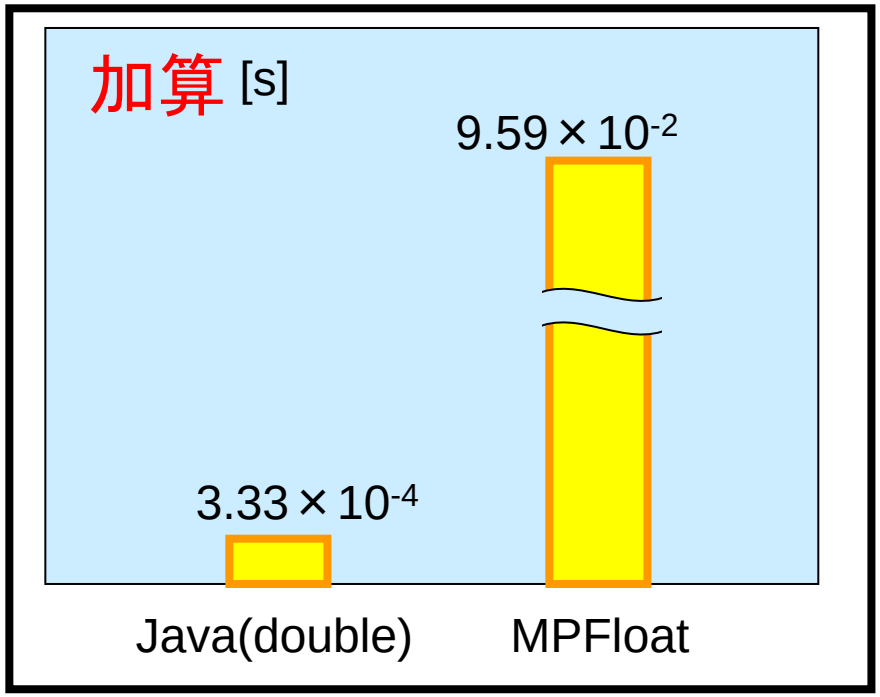
Exflib：藤原宏志(京都大学)

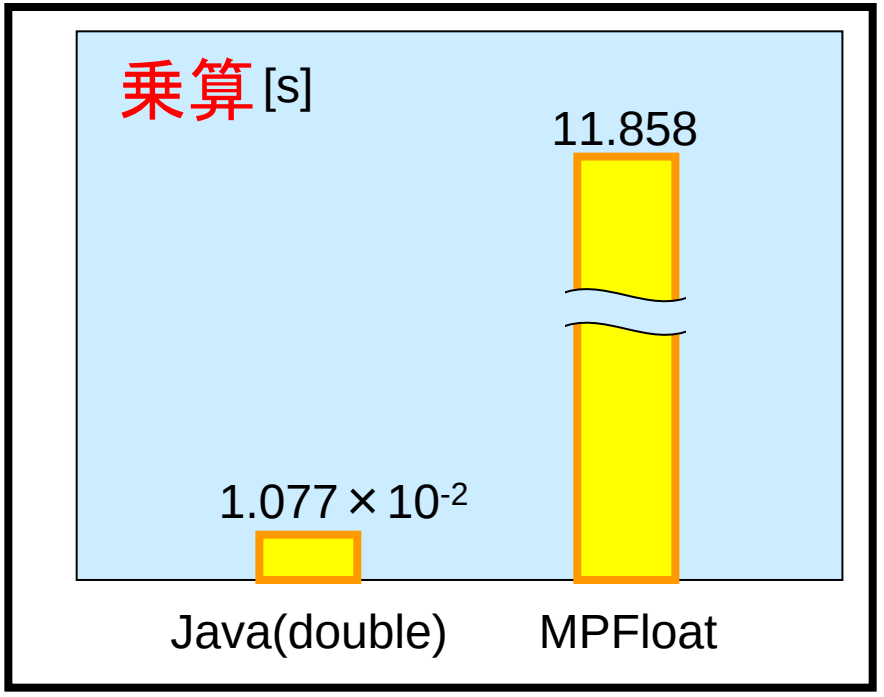
NFCの汎用数値クラス



速度性能評価

- 100×100の行列の加算と乗算
  倍精度（10進16桁）と多倍長（10進115桁）
- Window XP, Athlon 64 (1.8GHz)
- 数百倍の速度差(JNIのオーバーヘッド)





精度性能評価(1)

- 状態フィードバックによる極配置問題

System:

$$A=\begin{bmatrix} 1\times10^{-4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\times10^{-4} & 0 & 0 & 0 & 0 \\ 0 & 0 & 3\times10^{-4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 4\times10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 5\times10^{-4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 5\times10^{5} \end{bmatrix}, \quad B=\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

指定極: $P=-1,-5,-10,-15,-20,-25$

指定極と(A-BF)固有値の差

- 倍精度 (商用ツール)

0.0038秒

```
===  Error of poles (   6  x  1)  ===
          (  1)
(  1) (-5.743007289519024e+2,   0.000000000000000e+0)
(  2) ( 2.861960362183150e+2,   4.957929225226331e+2)
(  3) ( 2.861960362183150e+2,  -4.957929225226331e+2)
(  4) ( 3.986909989200000e-3,   0.000000000000000e+0)
(  5) (-3.393522412700000e-3,   0.000000000000000e+0)
(  6) ( 2.083002602000000e-4,   0.000000000000000e+0)
```

- 多倍長（10進115桁）

0.4008秒

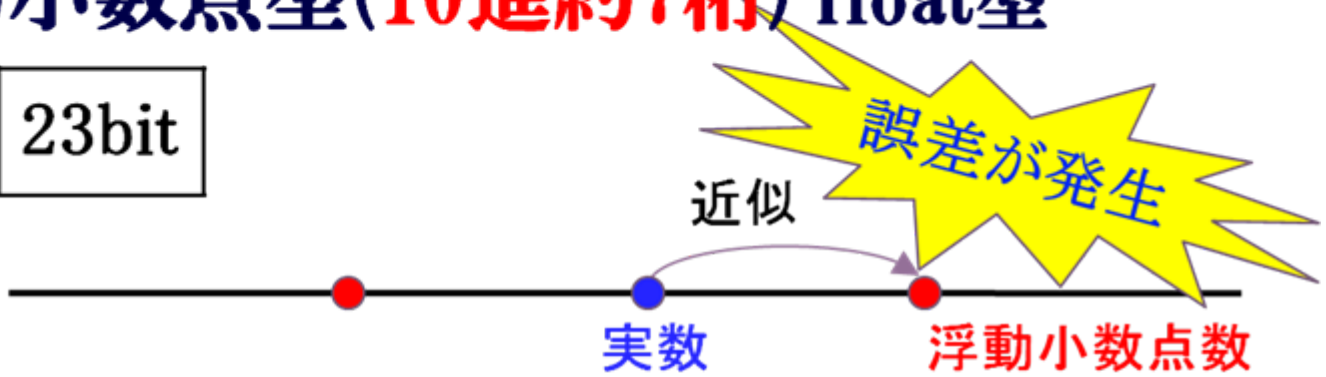
```
===  Error of poles (   6  x   1) ===
          (  1)
(  1) ( 1.048064351759036 2e-85,0.000000000000000e+0)
(  2) (-5.200415456411133 5e-83,0.000000000000000e+0)
(  3) ( 8.223035721123921 8e-82,0.000000000000000e+0)
(  4) (-3.125041499307045 3e-81,0.000000000000000e+0)
(  5) ( 4.370743734648816 2e-81,0.000000000000000e+0)
(  6) (-2.053413725447167 8e-81,0.000000000000000e+0)
```

浮動小数点数の丸め誤差

符号	指数部	仮数部

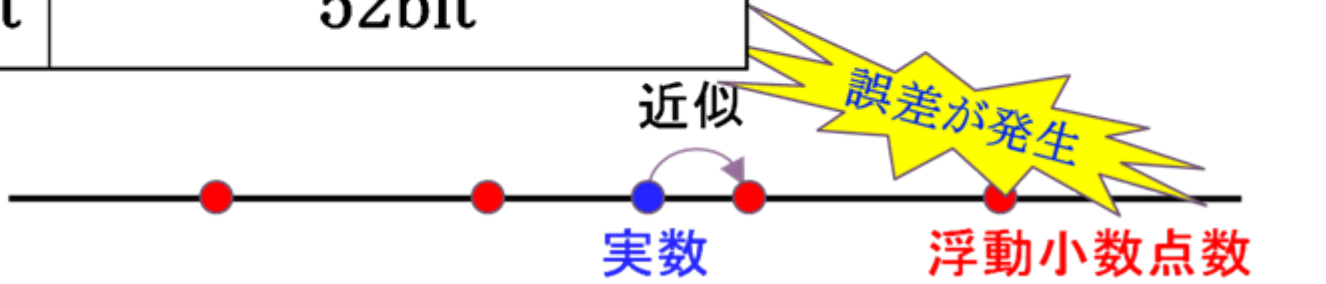
- **単精度浮動小数点型(10進約7桁) float型**

1bit	8bit	23bit



- **倍精度浮動小数点型(10進約16桁) double型**

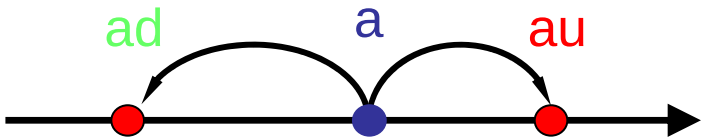
1bit	11bit	52bit



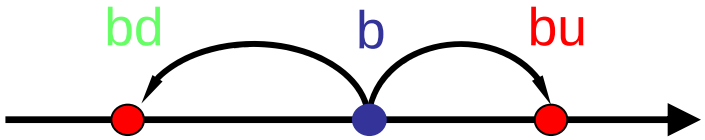
c = a + bの精度保証付き数値計算

区間演算

上向きの丸め: au = △a, bu = △b



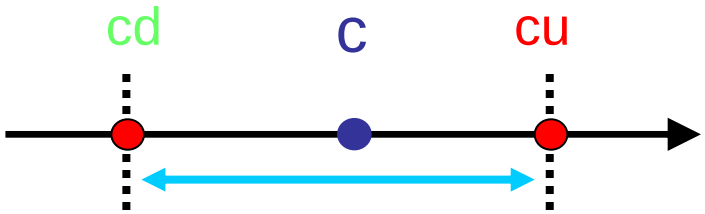
下向きの丸め: ad = ▽a, bd = ▽b



上限と下限 : cu = △(au + bu), cd = ▽(ad + bd)

⇩

$cd \leq c \leq cu$



真の解が含まれる(浮動小数点数を境界とする)集合

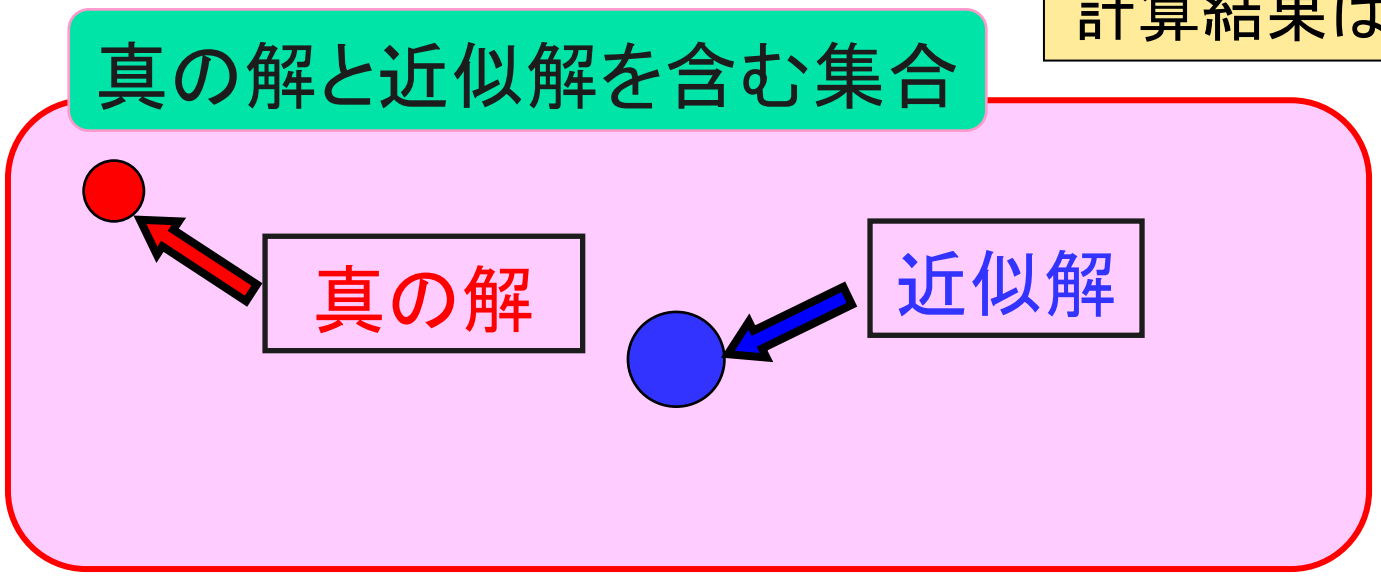
精度保証付き数値計算

計算結果の品質保証を行う

真の解は存在する？
近似解の精度は？

真の解と近似解を
含む集合を求める

計算結果は集合



JCGA (Java Computing Guaranteed Accuracy)

主要クラス

- Interval・IntervalMatrixクラス
  区間演算、区間行列演算
- LinearVerifierクラス
  線形方程式求解の精度保証
- EigenVerifierクラス
  固有値問題の精度保証
- HornerVerifierクラス
  多項式評価の精度保証
- IntervalDerivativeクラス
  微分値の精度保証
- NonLinearVerifierクラス
  非線形方程式求解の精度保証

参考文献

大石進一、2001、「精度保証付き数値計算」、コロナ社
G. I. Hargreaves、2002、Interval Analysis in MATLAB

G. I. Hargreaves、2002、Interval Analysis in MATLAB
山本の定理:
大石進一、2003、「応用解析セミナー　数値計算」、裳華房

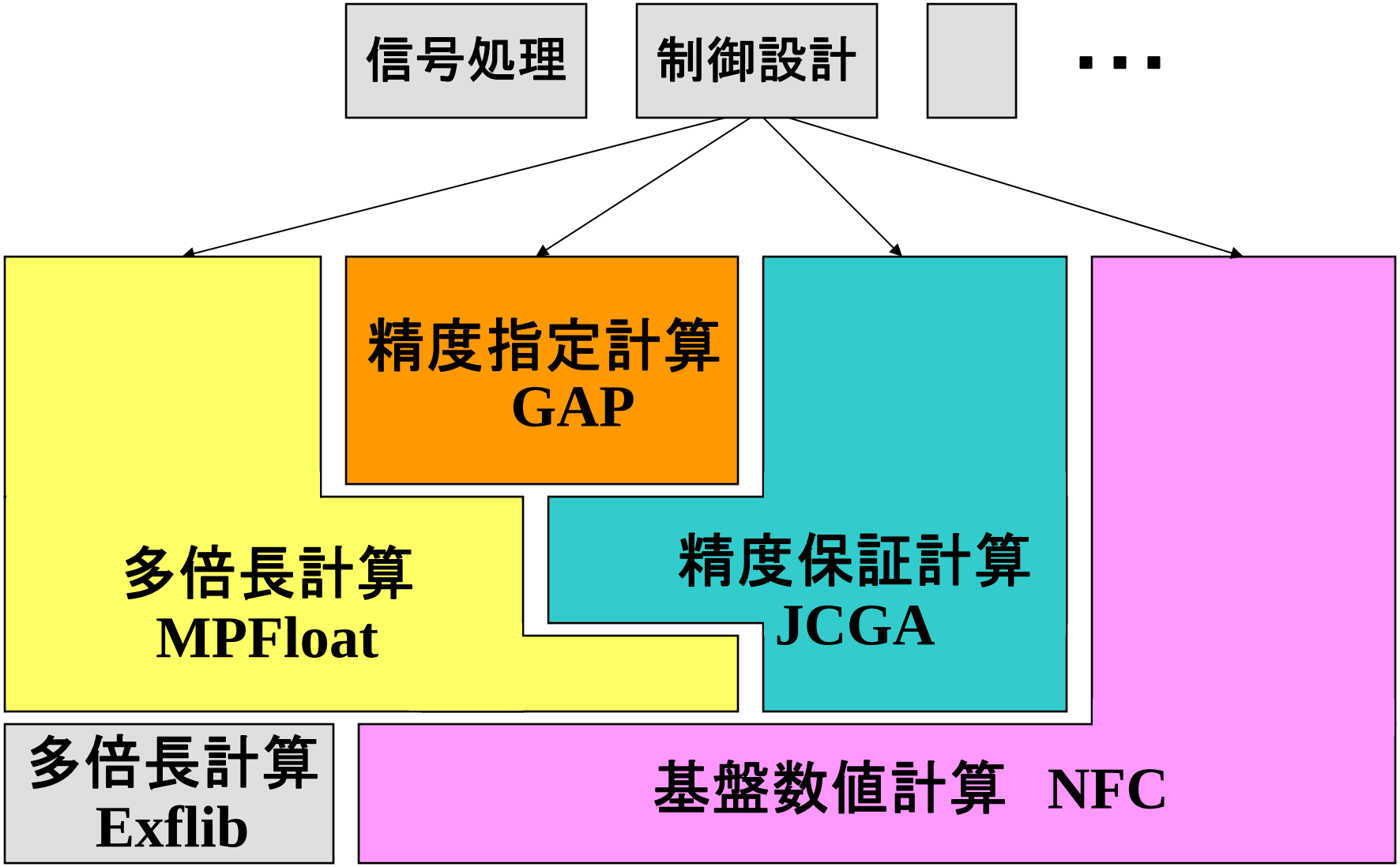
Rumpの方法:Siegfried M. Rump、2000
Computational error bounds for multiple eigenvalues

山本の定理の応用:
大石進一、2003、「応用解析セミナー　数値計算」、裳華房

自動微分法:
大石進一、2000、「Linux数値計算ツール」、コロナ社

クラフチック法:
大石進一、2001、「精度保証付き数値計算」、コロナ社

数値計算環境の構成


信号処理
制御設計
・・・
精度指定計算
GAP
多倍長計算
MPFloat
精度保証計算
JCGA
多倍長計算
Exflib
基盤数値計算　NFC

まとめ

- CとJavaの速さは同程度
- AOTやLLVMは有望である
- 書籍中のコードは機能不足、拡張性が低い
- 実用的なライブラリが利用可能
- 数値シミュレーションツールが利用可能
- 計算機資源を活用した高品質数値計算