

平成 29 年 6 月 26 日現在

機関番号：12612

研究種目：基盤研究(C) (一般)

研究期間：2014～2016

課題番号：26330144

研究課題名(和文) 整数演算における高精度および高性能なデータ並列処理を実現する計算手法の研究

研究課題名(英文) Research on data-parallel integer processing with high-precision and high-performance

研究代表者

村尾 裕一 (MURAO, Hirokazu)

電気通信大学・大学院情報理工学研究科・准教授

研究者番号：60174265

交付決定額(研究期間全体)：(直接経費) 2,400,000円

研究成果の概要(和文)：汎用プロセッサのSIMD命令やGPUのSIMTアーキテクチャ等の大量の同型の数値データを一括して並列に処理するための機構を、多倍長整数演算を主な対象として、様々な応用計算に至るまでの種々の整数演算に適用するために、適切な方法を検討し実証的な実験を行った。対象とした演算には、RNS表現による多倍長整数演算、ビット演算、浮動小数型で保持した整数データ、整数データ群で表現されたグラフ構造などを含み、これらでの高性能なデータ並列処理の有効性を示した。併せて、こうした実証実験を行うための適切なプラットフォームの構築法も習得することができた。

研究成果の概要(英文)：Most of the current processors equips with data-parallel processing facility, designed mainly for numeric processing, such as SIMD instruction sets and GPU's SIMT architecture. This research project investigates appropriate computing methods to apply their computing power to a wide variety of integer processing from arithmetics of integer data, usually with arbitrary precision, to some kinds of applications such as curve plotting of implicit functions and optimal graph-path search problem. Integer processing actually investigated includes integer arithmetics represented by RNS or represented in mantissa of floating point numbers, bit-operations for GF(2), and graph structures. Our demonstration experiments indicate the practical effectiveness. Also, we were able to acquire skills and methodology for constructing platforms appropriate for our parallel-processing experiments.

研究分野：数式処理

キーワード：高精度整数計算 中国剰余定理 GPGPU マルチコア並列処理 陰関数描画 WebGL 数式処理 高性能計算

1. 研究開始当初の背景

汎用プロセッサでは、複数のコアや演算器で構成して並列処理能力を高めて高性能化することが一般的になっている。汎用プロセッサの SIMD 命令や GPU の SIMT アーキテクチャは大量の同型の数値データを一括して並列に処理するための機能で、高性能計算に大きな役割を果たしている。その機能の主たる対象は浮動小数データだが、整数のデータや演算に対しても、一定の制限のもとでサポートされるようになってきている。特に Intel 系のプロセッサでは、AVX として大幅に拡張された SIMD 命令を整数型にまで拡張した AVX2 を実装した製品が登場してきた。また GPU に関しても、グラフィックス用と高性能計算 (HPC) 用へと分化し (その後更に AI 用も登場している)、各々で事情が異なる場合もあるが、浮動小数以外でも一定の精度に限定したり機械命令に即したプログラムの記述をしてやれば演算器の個数に見合った性能を引き出すことも可能になっているだけでなく、高い精度の整数計算の機能も導入される可能性も言及されている。つまり、整数計算を用いた処理でもデータ並列処理を活用することにより高性能計算を実現する可能性が高まってきてはいたが、現実にはあまり多くは研究されていない。一方、タブレット PC 等の小型機で用いられている ARM ベースのプロセッサでも、SIMD 命令や GPU が組み込まれ性能が大きく向上してきている。特に、既発表の 64bit ARM プロセッサにはタグ付きポイントが導入されているが、精度や表現の異なるデータなど複数のデータ型を実行時に型チェックをする場合に大きな性能向上をもたらす可能性があるのは既知のとおりである。(2016 年には国産スーパーコンは ARM ベースで開発されることが発表されたことも指摘しておく。)

代表者は、上述のプロセッサの発展の段階の早期の頃から、複数のコアや演算器による並列処理を準数値処理に適用することの可能性にいち早く着目し、科研費の補助も受けて研究開発を進めてきている。その成果にはパターン照合に基づくファイル圧縮とウィルス探索や暗号計算等、タブレット・スマホ等の小型システムでの有用性が見込まれるものも含まれる。SIMD/SIMT やマルチコアによる並列処理の機能を準数値処理において活用するための研究は、世界的に見ても例は多くはなかったが、その後徐々に論文発表が見られるようになってきている。実際、本計画の開始時点においても、この準数値的処理への並列処理の研究は多岐にわたり発展する可能性があることは明らかであった。

本計画で構想した研究対象において用いるアルゴリズムの理論的背景は、大部分が既成のものだが、多くの場合、プロセッサのアーキテクチャや構成に適合させて記述されてはいない。一方、数式処理の分野において整数を任意多倍長で扱い並列処理のために

多用される中国剰余定理が、回路設計の分野において RNS (Residue Number System) という名称で以前より研究されてきていることに代表者が気づいたのは割りと最近のことであった。その過去の成果は本計画でも有用なものが多いが、詳細で具体的な論文が複数の論文誌上で多数発表されており、活用するには幅広く精査する必要がある。

実証実験を行うための計算機環境についても、デスクトップやノート PC での汎用プロセッサ以外については検討が必要である。GPU は NVIDIA 社の CUDA が代表的・一般的だが、AMD 社の GPU と APU、あるいは、タブレット PC やスマホにも搭載されている小型の GPU を対象外とする理由は見当たらないばかりか、大きく発展する可能性が見込まれる。多様な並列アーキテクチャの機能を抽象化し、プログラムの記述を一元化するための言語として OpenCL が設計・開発されているが、その処理系が本計画の開始時点で一般的となってきた。OpenCL の利用は当然として、それまで固有の開発環境と処理系しか提供されていなかったプラットフォームについても、可搬性がより高まると予想される環境を、幅広い調査の上、利用可能な環境として整備していくことが必須となった。

2. 研究の目的

高性能プロセッサは、従来は主に浮動小数の数値計算で活用されてきたが、昨今のプロセッサ技術の発展により、多様な計算処理に有効に利用できるようになってきている。本研究では、その整数演算の機能を数式処理の基本演算やパターンの探索に適用し、処理の高速化を図る。この方面では並列アルゴリズムの理論的な研究が多く、実践的な知見は十分ではない。本研究では、多倍長整数に対する RNS 表現等により、処理対象のデータの数学的な階層構造やそれに基づく既存の計算法にとらわれずに、上位のデータ構造や処理全体の見方を変えることで多様な並列処理の方法を検討する。実証実験と各種アルゴリズムの理論的精査と計算量解析を繰り返すことにより、SIMD 命令や並列処理の効果的な利用法を探ることが、本計画全般にわたる主要な目的である。

前節の 1. の背景では、本研究の動機付けと位置付けに加え、本研究を実施する上で前提条件となる、理論的背景とその精査の必要性、および、計算機実験に用いるプラットフォームと開発環境の整備が必要であることを述べた。以降では、具体的な研究題材となる計算対象について説明する。いずれについても、目的は SIMD/SIMT/マルチコアによる並列処理への適合性を探ることである。また、以下に挙げる題材は検討の対象となるものを列挙するものであり、すべてに対し並列処理手法を確立しようとするものではない。(1) 整数演算を用いる並列処理の可能性のある題材には様々なものがあるが、準数値処

理・数式処理に関連した数種の基本的な演算を主要な題材として、SIMD/SIMT の利用と関連した並列処理手法や整数計算の精度及びこれらの記述言語による依存性等、題材毎に掘り下げて検討していく。整数の数値データを扱う計算としては、法による剰余数 (mod m) の組として表現 (RNS 表現) された任意精度の多倍長整数に対する四則演算と符号判定/大小比較、代数方程式の根の正確な分離、これらの応用として陰関数の正確描画、および、ビット消去による多倍長整数の GCD 計算を題材として用いる。他に、過去に一定の成果を挙げてきたパターン照合や最短経路探索などの準数値的処理についても従来成果を補う形で扱う。

(2) RNS の計算法については、前述のとおり、回路設計の分野で蓄積された様々な技法が知られており、我々の過去の研究で定番として確立させた計算法と併せて、整理した上で適宜ソフトウェアでの実装や SIMD/SIMT に適した形にして利用していく。並列処理は、RNS の法毎に独立した計算の並列実行が基本だが、しばしばメモリ所要量と並列度がトレードオフとなるため単純ではない。符号判定・大小比較では浮動小数も用いる方法が提案されており、アルゴリズムの選択には使用するプロセッサへの適合性を検討する必要がある。これは多倍長整数の表現で符号をどのように扱うかにも影響してくる。

(3) CUDA をはじめとして、進展著しいプロセッサをターゲットとする言語処理系では、本来あって然るべき機能が制限されていたり、ハードウェアの進化に伴って実用化・提供されるようになる有用で主要な機能もある。整数 GCD のアルゴリズムでも用いられる (多倍長データを分割) 分割統治法や経路探索では再帰呼出しを用いるのが普通だが、漸くサポートされるようになってきている再帰呼出しやスレッドが子供のスレッドを作成する dynamic parallelism の機能が、そうした実際的な応用の場面において、実用性がどの程度あるのかも興味深い。

(4) 他の題材については、従来研究計画でも検討し利用してきた既知のアルゴリズムを多用するが、その性能は剰余数の表現と計算法に大きく依存する。これは、倍精度の整数型が必要となる乗算では、C 言語の類では整数の精度拡張が記述できないため、機械語を用いたり元々の数の表現を予め半分の精度にしておく等の工夫が必要となったり、SIMD 命令を用いたとしても演算器の性能を生かしきれないといった問題が存在するためである。これに対処するには、昨今高性能数値計算の分野で研究の盛んな 4 倍精度や高精度での計算法のように数理的な解決法を見出すことと、複数の法を用いて RNS 表現された多倍長整数を係数とする多項式を複数の法に対する剰余数係数の多項式群として扱うという具合に、計算対象の数学的構造を一般的な捉え方と異なる見方で計算す

ることの両面から検討する必要がある。これらの要因の実行性能への影響はプロセッサのアーキテクチャに依存すると予想されるが、どのような条件の元で実効性能にどのような傾向が現れるかに関して一定の知見を得ることは難しいが重要な課題である。

(5) これらの応用としての代数方程式の根の正確な分離は、近年研究を進めてきた陰関数の正確描画を主たる目的としており、高次式を扱う場合には高精度な計算は必須である。効率のよい浮動小数計算と正確な RNS による計算を混合させるのが実践的だが、他にも零点探索を 2 分法に基づく木構造の計算と並列探索をどのように融合させるのか等を検討する必要がある。また、近年研究を続けている独自の描画法では、孤立した特異点も検出するためにグレブナ基底を計算するため数式処理ソフトウェアと並列計算ソフトウェアとの連携も必要となる。そのためのソフトウェア技法を検討する必要もある。

3. 研究の方法

研究目的に記した題材 (r) 多倍長整数を表現する RNS の四則演算・符号判定と (a) 代数方程式の根の正確な分離、(p) 陰関数の正確描画、(g) ビット消去による多倍長整数の GCD 計算) の各々に対して、アルゴリズムの理論的な究明と開拓、基本となる実装、SIMD/SIMT 処理を利用した並列処理と高速化のための実装の改良、OpenCL による記述、これらの実装の実機上での性能評価と性能の理論的解析を題材間で適宜オーバーラップさせながら進めていく。以降に、より具体的な研究の方法と内容を列挙する。但し、本計画でターゲットとする並列処理のハードウェアやソフトウェア環境は発展・変革のスピードが速いため、当初の計画は小規模でかつ机上での検討を主体とし、研究対象となりそうな題材を以下に列挙するが、計算機環境の状況に応じて具体的に扱う題材は取捨選択し内容も適切に変化させていく。最初の 2 つの項目は直接研究の対象や内容となるものではないが、無視できない量の実作業の時間と労力を要するものであり、研究を遂行するために必須の事項として挙げておく。

(1) SSE・AVX などの SIMD 命令、プロセッサ・コア群とキャッシュやローカルメモリの構成等、プロセッサ・アーキテクチャの現状と将来の動向を調査・学習する。ターゲットとする主なプロセッサは、一般的な Intel 系以外にも、32/64 ビットの ARM プロセッサ、AMD 社の APU、ATI 社と NVIDIA 社の GPU と、広く普及・流通しているものとする。

SIMD 命令については、整数データ向けの拡張が導入された AVX2 命令体系に注目する。また CUDA に関しては、本計画開始時において、新たな機能であった再帰呼出しと dynamic parallelism とその安定性、及び、将来導入が予定されていた GPU への仮想メモリ空間の導入や CPU メモリと GPU メモリの統

合といった新機能に注目する。64ビットARMプロセッサのタグ付きポインタについては、利用可能なOSと、どんな条件で利用可能なかを調べる必要がある。

(2) 計算機実験の環境の整備・構築・研究期間の初期の頃はPC用のOSが32ビットから64ビットに移行していく時期に重なり、CUDA等の並列処理実行環境も32/64ビットの両方の版が提供されるばかりか、LinuxではOS/カーネルの版に依存する状況にあった。またCUDAについては、強化された機能の提供がリリースに依存している。そのため、複数のリリースが共存可能かを調べることや、計算機資源を有効活用するためには、マルチブート等複雑な利用環境を構築することが必要であった。更に、複数種のハードウェアターゲット(数種のCPU/APU/GPU)への対応や、そのためのOpenCLの利用可能性にも検討が必要であった。ARMプロセッサについては、スマートフォンやタブレットPCでは一般的ではあるものの、アドレス空間の32/64ビットについてはPCと同様の状況があると同時に、一般的な開発環境が未知であった。

(3) 題材(r)のRNS表現された多倍長整数の演算については、既存アルゴリズムを理論的に精査し実装法を検討する。RNS表現の整数では、符号判定に浮動小数点数を用いる方法では必要精度と実行性能を調整する必要があり、また、乗算では精度拡張のための基数変換とリスト処理が必要となる。試験的なプログラムを作成し、GPUやSIMD命令の利用方針を実証実験を通して定めていく。

(4) 題材(g)の整数GCDについては、主にGPUでの利用を目指して、一般的な多倍長の表現を用いずにバイトデータの列に対するアルゴリズムへと分解することと、再帰呼出しを除去する方法まで検討する。

(5) 題材(a)の代数方程式の根の分離は(p)の陰関数の描画でも必要となるが、題材(r)中の符号判定を用いるだけでなく、2分法やそれを一般化したn分法の再帰呼出しやdynamic parallelismを考慮した高レベルでの並列処理の手法を検討する必要がある。

(6) 題材(p)の陰関数の描画は、上記の項目の内のいくつかの最終的な目標だが、グレブナ基底の計算のような代数的な手法の計算の軽減と3次元の場合への拡張という理論とアルゴリズム面での研究が必要である。

以上の研究は、従来から共同で研究を進めてきた研究者らと連携しながら進めていく。

4. 研究成果

整数計算を任意精度で正確かつ高速に行い、数式処理関連の計算や非数値的な処理において効率的・効果的なデータ並列処理を実践するために、研究対象とした題材のアルゴリズムとその骨格を精査し、それに基づく適切なデータ表現の方法の検討を進め、適切な計算手法の開拓を行った。具体的には、陰関数

の正確描画をはじめとした典型的な問題例において、SIMD命令等の今日のCPUに共通するデータ並列処理の手法を活用するための適切な方法やアルゴリズムを選択し、更に大規模なソフトウェアライブラリとして一元的に記述する方法を検討することを当初の計画とした。しかしながら、検討を進めるに従いGPU、SoCはおろか汎用CPUでさえアーキテクチャが予想外に発展し多様化していることが明らかとなり、その準数値処理分野での特定の問題に対する活用可能性と有効性の検証に重点を置くように若干の方針の修正をすることとした。併せて、OpenCLと(当初の計画にはなかった)WebGL等のアーキテクチャに特化していない言語の記述性と効率の検証も進めた。

(1) 多倍長整数関連の演算の並列処理については、研究計画の検討や実験に加え、連携研究者による大規模な行列関連の計算の計算機実験とアルゴリズムと並列処理との適合性の検証(論文、発表)と、学生によるGF(2)上の線形方程式の解法にGPUを適用する実証実験(発表)が将来性のある成果として挙げられる。

再帰呼出しを行う分割統治法に基づくGCD計算については、GPU上での再帰呼出しやdynamic parallelismの機構は一般的な手法として制限も強く安定性にも疑問があるといった情報を熟達者から得たため、アルゴリズムの分析と簡単な実験で留めている。一方、従来より再帰呼出しを除去してA*アルゴリズムを実装していたGPUを用いた最短経路探索は、より多くの問題に対して実証実験を行い、CPUのマルチコアとGPUの比較を進めた。(論文)

(2) ARMプロセッサの開発環境については、先ず32bit版のAndroidに対する標準的な方法を、研究者仲間からの情報に基づき構築することができた。次に、機(関連ソフトの整備状況)を見て、64bit版の実験環境の整備を行った。複数のデータ型(複数のデータ表現法を扱う数値データおよびその行列)を扱う場合に、扱う値の範囲とタグの使い方を上手く設定し、即値データと混在させることにより高速化を図ることが可能と思われる。構築した計算環境上において、その性能評価を行うための計算手法と実験法の検討を進めたが、実測結果を得るまでにはいたらなかった。継続して研究を進めていくべき今後の研究課題である。ARMプロセッサの利用法の調査のために、いち早くコンパイラを開発し、過去にタグを活用した処理系を開発した英国の知人を訪問することを計画していたが、主に健康上の理由から取りやめた。

(3) 陰関数/代数関数の正確な描画については、二変数多項式で定義される平面上の曲線の場合は、微小な閉曲線や特異点が描画領域にある場合でも、0次元となるグレブナ基底を求め、基底として得られる1変数多項式の根を分離すれば、そうした点や曲線を見落

とさずに正確に描画することが可能である。その3次元化へ向けて、2次元の場合の原理と拡張法の検討を数理的に続けてきている。3次元の場合の完璧な方法は見つからず簡易な方法の提案にとどまっているが、2次元の場合の簡易な方法の改良法を数種類提案し、実証実験に基づいて実用性を示した。(論文 , 発表)

また2次元の場合について、数値処理のみを行う2種類の単純な方法を WebGLによるブラウザ上での GPGPU 処理の実験題材として用い、その実証実験を通して高い汎用性と実用性を有するという新たな知見を得るに至った。(論文 , 発表)

(4) 前項で記した JavaScript と WebGL 及び WebCL(ブラウザ向けの OpenGL. 開発は初期段階で停止)による GPGPU は、ブラウザというごく一般的な環境で利用可能なだけでなく、タブレット PC やスマホに搭載される GPU まで利用可能とする(可能性がある)。高性能処理を手軽にかつ多様なプラットフォーム上で可能とすることから、有望と考え、主要な実験環境として扱うこととした。その後の実験により、様々な整数演算でも有用であること、浮動小数演算は WebGL の最新の版では倍精度がサポートされる一方、現状の多くのプラットフォームでは半精度であることが判明している。当初は単精度から精度拡張の方法を検討したが、半精度から我々の応用で必要となる数値の値域まで拡張することは難しいため、今後他の方法を検討することとした。(論文 , 発表)

以上、当初計画した項目の部分部分を計算機環境や調査・検討の進捗状況に合わせて研究を実施した。また、当初計画になかった事柄でも、関連研究として有望と思われることに関しては優先させて研究を進めた。そのため様々にやり残している部分があり、それらについては今後も進展させていく予定である。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文](計14件)

北村竜之介, 大墨礼子, 近藤祐史, 村尾裕一, 齋藤友克. 行列式演算における演算速度と速度要因に関する考察. 数式処理. 査読有. 第23巻第1号. 2017. 10-19.

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克. 3変数陰関数描画の実装について. 数式処理. 査読有. 第23巻第1号. 2017. 20-29.

北村竜之介, 大墨礼子, 近藤祐史, 村尾裕一, 齋藤友克. 行列演算の高速化要因に関する一考察. 数式処理. 大会報告, 査読無. 第22巻第2号. 2016. 75-79.

木村欣司, 村尾裕一. ハードウェアの特性を考慮した有限体の実装法について.

数式処理. 大会報告, 査読無. 第22巻第2号. 2016. 80-83.

N.Hyodo, Y.Kondoh, H.Murao, T.Saito and T.Takahashi. Practice of Drawing Graphs of Implicit Functions of Three Variables. Communications of JSSAC. 査読有. Vol.2. 2016, pp.33-42.

桑原国仁, 村尾裕一. Web ブラウザ上での陰関数グラフの描画法と並列処理による高速化. 信学技報. 査読無. 第114巻. 2016. 139-144.

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克. Asir での3変数描画について. 京都大学数理解析研究所講究録. 査読無. No.1976. 2015. 17-26.

近藤祐史, 兵頭礼子, 村尾裕一, 齋藤友克. Asir における陰関数描画 ifplot の改良. 数式処理. 大会報告, 査読無. 第21巻第2号. 2015. 76-79.

木村欣司, 村尾裕一. 中国剰余定理による高速化のための浮動小数点数の符号 bit の利用について. 数式処理. 大会報告, 査読無. 第21巻第2号. 2015. 68-71.

H.Hayakawa, N.Ishida and H.Murao. GPU-acceleration of optimal permutation-puzzle solving. Proceedings of the 2015 International Workshop on Parallel Symbolic Computation(PASCO'15). 査読有. 2015. 61-69. DOI:10.1145/2790282.2790289

北村竜之介, 兵頭礼子, 近藤祐史, 村尾裕一, 齋藤友克. Risa/Asir での行列演算高速化の試み. 数式処理. 査読無. 第21巻第1号. 2015. 17-23.

兵頭礼子, 近藤祐史, 村尾裕一, 齋藤友克. ifplot での3次元描画の拡張. 京都大学数理解析研究所講究録. 査読無. No.1907. 2014. 166-173.

兵頭礼子, 北村竜之介, 近藤祐史, 村尾裕一, 齋藤友克. Risa/Asir での行列演算高速化の試み. 京都大学数理解析研究所講究録. 査読無. No.1927. 2014. 122-127.

兵頭礼子, 近藤祐史, 村尾裕一, 齋藤友克. ifplot の改良. 数式処理. 大会報告, 査読無. 第20巻第2号. 2014.

兵頭礼子, 近藤祐史, 村尾裕一, 齋藤友克. ifplot の改良. 数式処理. 大会報告, 査読無. 第20巻第2号. 2014.

[学会発表](計15件)

菟雅文, 村尾裕一, 木村欣司. GF(2)上の疎な線形方程式に対する block Wiedemann アルゴリズムの GPU を用いた実装と評価. Risa/Asir Conference 2017. 2017年3月28~30日. 金沢大学(石川県金沢市).

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克. 陰関数描画における描画関数の改良. 日本数式処理学会合同分科会. 2017年1月21,22日. 京都大学産連本部東京日本橋サテライトオフィス(東京都中央区). 菟雅文, 村尾裕一. GF(2)上の疎な連立

線形方程式に対する Block-Wiedemann アルゴリズムの GPU 実装実験 . 日本数式処理学会合同分科会 . 2017 年 1 月 21, 22 日 . 京都大学産連本部東京日本橋サテライトオフィス (東京都中央区) .

木村欣司, 村尾裕一 . 高次多項式の判別式計算と固有多項式計算による Intel Xeon Phi KNL の性能評価 . 研究集会「数式処理とその周辺分野の研究」. 2016 年 12 月 7 ~ 9 日 . 京都大学数理解析研究所 (京都府京都市) .

村尾裕一, 桑原国仁, 大墨礼子, 近藤祐史, 齋藤友克 . WebGL を用いたブラウザ上での陰関数描画 . Risa/Asir Conference 2016 . 2016 年 3 月 28 ~ 30 日 . 金沢大学 (石川県金沢市)

D. Mend-Amar, 村尾裕一 . Web ブラウザ上での符号化処理の高速化 . 電子情報通信学会 2016 年総合大会, 情報・システムソサイエティ, ISS-P-57 . 2016 年 3 月 15 ~ 16 日 . 九州大学 (福岡県福岡市) .

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克 . Asir における陰関数描画について . 日本数式処理学会合同分科会 . 2016 年 1 月 23 ~ 24 日 . 名古屋大学 (愛知県名古屋市)

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克 . 3 変数陰関数描画に関する考察 . 研究集会「数式処理とその周辺分野の研究」. 2015 年 12 月 2 ~ 4 日 . 京都大学数理解析研究所 (京都府京都市)

北村竜之介, 大墨礼子, 近藤祐史, 村尾裕一, 齋藤友克 . 行列式演算における演算速度と速度要因に関する考察 . 日本数式処理学会特別研究会 . 2015 年 10 月 17, 18 日 . 東北大学 (宮城県仙台市) .

近藤祐史, 大墨礼子, 村尾裕一, 齋藤友克 . 3 変数陰関数描画の実装について . 日本数式処理学会特別研究会 . 2015 年 10 月 17, 18 日 . 東北大学 (宮城県仙台市) .

北村竜之介, 大墨礼子, 近藤祐史, 村尾裕一, 齋藤友克 . 行列演算の高速化要因に関する一考察 . 日本数式処理学会第 24 回大会 . 2015 年 6 月 5 ~ 7 日 . 筑波大学 (茨城県つくば市) .

木村欣司, 村尾裕一 . ハードウェアの特性を考慮した有限体の実装法について . 日本数式処理学会第 24 回大会 . 2015 年 6 月 5 ~ 7 日 . 筑波大学 (茨城県つくば市) .

北村竜之介, 兵頭礼子, 近藤祐史, 村尾裕一, 齋藤友克 . Risa/Asir における行列の演算高速化 . 日本数式処理学会合同分科会 . 2015 年 1 月 23, 24 日 . 山形大学 (山形県山形市) .

近藤祐史, 兵頭礼子, 村尾裕一, 齋藤友克 . Asir における陰関数描画 ifplot の改良 . 第 23 回日本数式処理学会大会 . 2014 年 5 月 30 日 ~ 6 月 1 日 . 徳島大学 (徳島県徳島市) .

木村欣司, 村尾裕一 . 中国剰余定理によ

る高速化のための浮動小数点数の符号 bit の利用について . 第 23 回日本数式処理学会大会 . 2014 年 5 月 30 日 ~ 6 月 1 日 . 徳島大学 (徳島県徳島市) .

〔図書〕(計 0 件)

〔産業財産権〕

出願状況 (計 0 件)

名称 :
発明者 :
権利者 :
種類 :
番号 :
出願年月日 :
国内外の別 :

取得状況 (計 0 件)

名称 :
発明者 :
権利者 :
種類 :
番号 :
取得年月日 :
国内外の別 :

〔その他〕
ホームページ等

6. 研究組織

(1) 研究代表者

村尾 裕一 (MURAO, Hirokazu)
電気通信大学・大学院情報理工学研究所・
准教授
研究者番号 : 6 0 1 7 4 2 6 5

(2) 研究分担者 なし

(3) 連携研究者

木村 欣司 (KIMURA, Kinji)
京都大学・大学院情報学研究所・特定准教
授
研究者番号 : 1 0 4 4 7 8 9 9

大墨(兵頭) 礼子 (OSUMI (HYODO), Noriko)
サレジオ高等専門学校・情報工学科・准教
授
研究者番号 : 4 0 7 1 2 6 0 9

(4) 研究協力者 なし