

# Oakforest-PACS を用いた格子 QCD の master-field simulation

藏 増 嘉 伸

筑波大学計算科学研究センター

石 川 健 一

広島大学大学院理学研究科

## 1. はじめに

われわれが住むこの自然界は、「強い力」、「電磁気力」、「弱い力」、「重力」の 4 つの力によって支配されている(「強い力」、「弱い力」は固有名詞)。日常生活において、重力と電磁気力は馴染み深い力であるが、弱い力を認識する機会は稀である。強い力はこの世の中に存在するありとあらゆる分子の中の原子核を構成している力である。強い力は物質の最小構成粒子(素粒子)の一つであるクォークにグルーオンを媒介粒子として作用する。これまでのところ、質量の異なる 6 種類のクォーク(質量の小さい順にアップ、ダウン、ストレンジ、チャーム、ボトム、トップという名前が付与されており、各々  $u$ ,  $d$ ,  $s$ ,  $c$ ,  $b$ ,  $t$  と表記される)が存在することが実験的に確認されており、このクォークの種類をフレーバーと呼ぶ。各フレーバーのクォークはカラーチャージ(赤、青、緑)と呼ばれる 3 つの内部自由度を持っており、このカラーチャージをグルーオンという質量ゼロのゲージ粒子を媒介として交換することによって相互作用している。

強い力は、文字通り 4 つの力の中で最も強く、2 番目に強い電磁気力の約 100 倍の強度を持っている。ただし、違いは強度だけではなく、相互作用の性質も全く異なる。強い相互作用を特徴づける最も奇妙な性質はクォークの閉じ込めである。実験で観測されるものはハドロンと呼ばれる複数のクォークの束縛状態(クォーク 3 つの束縛状態をバリオン、クォークと反クォークの束縛状態をメソンと呼ぶ)だけであり、クォーク単体が観測されることはない。つまり、クォークはハドロン内部に閉じ込められている。したがって、実験によって得られる観測量を理論的に計算しようとする場合、閉じ込めの効果を定量的に評価することが必要である。しかしながら、閉じ込めは相互作用の強さから生み出される非摂動的な現象であるため、非摂動的な解析手法が必要となってくる。そこで格子 QCD を用いた数値計算が求められる。格子 QCD は、離散化した 4 次元時空上で QCD を定義することによって自由度を有限化し、マルコフ過程モンテカルロ法に基づく数値計算を用いて、強い相互作用の第一原理である QCD に基づいた非摂動計算を可能とする。

格子 QCD は、1974 年 Wilson によって提唱され[参考文献 1]、1980 年頃から高性能計算機の登場とともに本格的な数値計算が始まった。それ以来、格子 QCD は常に世界最高レベルのスーパーコンピュータを用いた大規模計算によって計算素粒子物理学を牽引してきた。格子 QCD の数値計算が始まった当初は、クォークの真空偏極効果を見捨てることによって計算コストを抑えた、いわゆるクエンチ近似を用いた小中規模計算に限定されていた。そうした状況の中、筑

波大学計算物理学研究センター(1992 年度～2003 年度)は、計算機開発とそれを用いた素粒子物理学研究の両面において世界を牽引した。特に、1996 年には格子 QCD 計算を主目的の一つとする超並列計算機 CP-PACS(ピーク演算性能 0.61TFlops, 1996 年 11 月 Top500 で第 1 位)を完成させ[参考文献 2], その集中利用によりクエンチ近似でのハドロン質量の精密計算に成功した[参考文献 3]。そこで明らかになったことは、クエンチ近似計算でのハドロン質量は 10% 程度の精度で実験値を再現できるが、数%の精度で実験値と一致させるにはクエンチ近似では不十分であり、クォークの真空偏極の寄与が必要であるということであった。これにより、世界における研究の焦点はクォークの真空偏極効果を取り入れた計算(full QCD 計算)へと移行していくこととなった。

Full QCD 計算のアルゴリズムは 1980 年代から精力的に研究され、1990 年代初頭にはハイブリッドモンテカルロ法[参考文献 4]がベースアルゴリズムとして確立していた。しかしながら、クォーク質量が軽くなるにつれてシミュレーションに要する計算量が飛躍的に増大するため、物理的な u, d, s クォーク質量上でシミュレーションを行うことは、格子 QCD の創始以来の大目標であった。2000 年代前半までの典型的な計算においては、まず人為的に u, d クォーク質量を縮退させ(以下 ud クォーク質量と呼ぶ)、さらにその物理的な平均質量(物理点)の約 10 倍程度の質量で計算を行い、結果を縮退した ud クォーク質量に関して物理点まで外挿するという方法が取られてきた。この問題を解決すべく、われわれは筑波大学計算科学研究センター(2004 年度に計算物理学研究センターから改組)において開発された超並列クラスタ計算機 PACS-CS(ピーク性能: 14.3TFlops, 2006 年度～2011 年度)を用いて、物理的な ud, s クォーク質量上でのシミュレーションを実現し、更にそれが物理的に本質的な重要性を持つことを示した[参考文献 5]。

図 1 (左)は、 $\pi$  メソン質量の 2 乗と ud クォーク質量の比を ud クォーク質量の関数としてプロットしたものである。PACS-CS 機を使って得た結果(黒丸)は明らかな曲率を示しているのに対して、CP-PACS 機を使って得た重い ud クォーク質量での結果(白抜き丸)はほぼ直線に見える。この図から、重いクォーク質量領域からの外挿(直線)は定量的に間違った答えを与えるが、その問題は物理的クォーク質量上での直接計算によって解決されることがわかる。図 1 (右)はハドロン質量スペクトルを実験値と比較したものである。大半の粒子について統計誤差の範囲で一致しているが、そうでない場合でもズレは最大で 2-3%程度である。

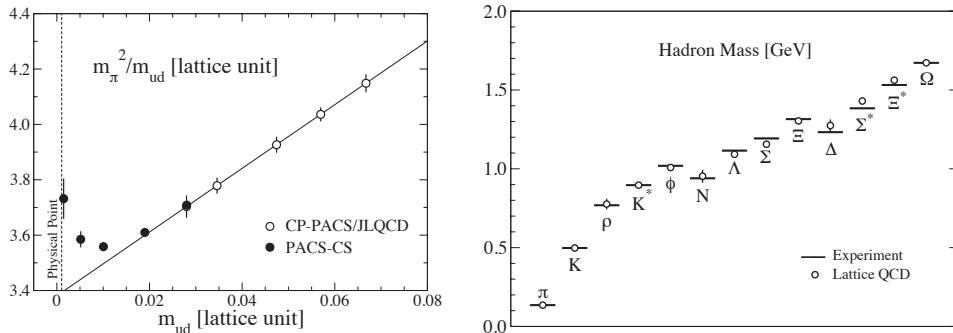


図 1 : PACS-CS を用いた 2+1 フレーバー格子 QCD 計算。

左図は  $\pi$  メソン質量の 2 乗をクォーク質量で割ったもののクォーク質量依存性。PACS-CS 機での結果(黒丸)と CP-PACS 機での重い ud クォーク質量での結果(白抜き丸)の比較。右図は各種のハドロン質量の格子 QCD

の計算結果と実験値の比較。

格子 QCD シミュレーションの発展は、計算機性能の向上とともに計算アルゴリズムの開発・改良が鍵であった。格子 QCD 計算でベースとなるアルゴリズム要素として、モンテカルロ法、分子動力学法、乱数生成、大規模連立線形方程式の反復解法、誤差解析、などが挙げられる。これらは、計算機アーキテクチャの変遷とともに、時代ごとのアーキテクチャの特性に即したアルゴリズム開発・改良が施されてきた。PACS-CS 機はコモディティを用いた大規模並列クラスタ計算機であったため、演算性能に対する通信性能の相対的な弱さを回避するために、領域分割型ハイブリッドモンテカルロ法[参考文献 6]が採用され、分子動力学法部分には Sexton-Weingarten の多重時間刻み積分スキーム[参考文献 7]や Omelyan-Mryglod-Folk 積分スキーム[参考文献 8]が組み込まれた。また、格子 QCD シミュレーションにおいて 8 割以上の計算コストを占める大規模連立線形方程式の反復解法に対して精度混合型二重 BiCGStab 法(2 節参照)が用いられ、格段の計算速度向上を実現した。

PACS-CS 機を用いた研究によって物理点直上での格子 QCD シミュレーションが可能となったが、2010 年当時、世界各国の複数のグループの核子形状因子(電荷半径などを与える)の計算結果から、強い相互作用の精密計算のためには、物理点直上という条件だけではなく、これまでの典型的な計算における  $(5 \text{ fm})^3$  程度の空間体積よりもはるかに大きな空間体積が必要なのではないかと、という可能性が指摘されていた。われわれは、HPCI 戦略プログラム分野 5(2011 年度～2015 年度)の課題として、「京」コンピュータを用いて物理点近傍( $m_\pi=145 \text{ MeV}$ , 物理点は  $m_\pi=135 \text{ MeV}$ )での  $(8 \text{ fm})^3$  超の空間体積を持つ大体積シミュレーション(PACS-CS 機を用いたシミュレーションの空間体積は  $(3 \text{ fm})^3$ , 格子間隔はともに  $0.09 \text{ fm}$  程度)を遂行し、世界で初めて核子の電荷半径を再現することに成功した[参考文献 9]。

「京」コンピュータを用いた大体積シミュレーションは、われわれに新たな発見をもたらした。従来の格子 QCD 計算では、体積を大きくすることと統計精度を上げることは相克する課題であった。つまり、体積を大きくするにせよ、統計精度を上げるにせよ、各々独立に計算コストが増大するため、どちらかに注力した計算を行えば、他方は犠牲にせざるを得ないと考えられてきた。しかしながら、われわれは「京」コンピュータを用いて  $(8 \text{ fm})^4$  超の時空間体積を持つ大体積シミュレーションを行った結果、物理量の統計誤差が 4 次元時空間体積  $V$  に対して  $1/\sqrt{V}$  に比例して減少することを確認し、実は体積を大きくすることは同時に統計精度を向上させることを明らかにした。これにより、われわれは物理点直上における超大体積シミュレーションの優位性を提唱し、現在は約  $(11 \text{ fm})^4$  の時空間体積を持つシミュレーションを行っている(最近同様のアイデアが Lüscher によって提唱され、master-field(万能配位)シミュレーションと名付けられた)。これは、現在の世界の格子 QCD 計算が  $(5\sim 6 \text{ fm})^4$  の体積で行われていることを考えると、格段に大きな時空間体積でのシミュレーションである。4 次元体積を大きくすることは、統計誤差を縮減させるだけでなく、副次的な利点もある。まず物理的な観点から言えば、様々な物理量に対する有限体積効果を抑えることができるだけでなく、核子の電荷半径の決定において鍵となる空間運動量が小さい領域における核子形状因子を計算することが可能となる。また、計算機科学的観点から言えば、master-field シミュレーションは弱スケール性能が担保されている大規模並列計算機であれば実行可能である点で優れている。逆に、小体積で配位数を増やして統計誤差の縮減を目指す場合は強スケール性能を持つ計算

機が好ましいが、近年では技術的制限から強スケーリング性能を持つアーキテクチャの開発は難しい。

現在格子 QCD において master-field シミュレーションを実行・計画しているグループは、世界中でわれわれ筑波大学を中心とした PACS Collaboration のみである。われわれが世界の先陣を切っている理由は主に二つある。一つは、予備的計算として、「京」コンピュータを用いて  $(8 \text{ fm})^4$  程度の時空間体積を持つ大体積シミュレーション(ただし物理点からは少し外れている)を実行できたことである。これにより、体積を大きくする利点について知見を得るとともに、master-field シミュレーションの本格実施へと繋がる技術的ノウハウを得ることができた。二つ目は、平成 28 年末から稼働を開始した Oakforest-PACS (OFP) の存在である。このスーパーコンピュータは「京」コンピュータの 2 倍以上のピーク演算性能を持っており、master-field シミュレーションが要請する物理点直上での超大体積シミュレーションを実行できるポテンシャルを有している。

本稿では、2 節において格子 QCD アルゴリズムの概略と OFP 上でのコード最適化を説明し、その後、3 節において PACS Collaboration が推進中の OFP を用いた master-field シミュレーションプロジェクトの一例となる結果を紹介する。

## 2. 格子 QCD のアルゴリズムと OFP 上での最適化

格子 QCD の計算は大まかに二つのステージに別れている。ひとつめは、グルーオン場を量子論的な確率分布に従って生成するステージであり、ふたつめは、前者のステージで生成されたグルーオン場を背景にクォーク場の伝搬関数を計算し物理量に応じて組み合わせるステージである。前者のステージではグルーオン場が確率変数となるような格子 QCD 分配関数に基づくモンテカルロ法が用いられる。特に full QCD 計算では格子 QCD 分配関数にクォークの真空偏極効果を表す項が含まれており、マルコフ過程により変化していくグルーオン場と矛盾しないようにクォークの運動方程式を解き続ける必要がある。また、ふたつめのステージでもクォーク場の伝搬を表す伝搬関数をクォークの運動方程式の解から求める必要がある。実際 full QCD の計算では殆どの計算時間はクォークの運動方程式を解くことに費やされる。OFP を用い、物理的クォーク質量で大体積の full QCD シミュレーションを行うためにはクォークの運動方程式の解法部(クォークソルバーと呼ぶ)の OFP 向けの最適化が重要となる。我々は、以下で述べるような様々な OFP 向けの最適化を行い、その性能を測定した。

### 2-1. クォークの運動方程式

グルーオンとクォークの運動方程式には様々な離散化の方法がある。我々は、物理的クォーク質量で大体積の計算を行うのに有利な  $O(a)$  の格子間隔誤差を取り除いたウィルソンクォーク ( $O(a)$  改良ウィルソンクォーク) を用いている。離散化された時空は四次元の規則格子となり、その上でのクォーク運動方程式は四次元の 9 点ステンシルの差分方程式となる。クォークは四つのスピン自由度  $\alpha$  と三つのカラーチャージの自由度  $i$  を組み合わせた 12 個の複素数の自由度を持ち、格子点「サイト」 $n$  上に存在する  $(x_\alpha^i(n))$ 。グルーオン場はクォーク間のカラーチャージによる相互作用を表現するように格子点をつなぐ辺「リンク」上に存在し、 $3 \times 3$  のユニタリー行列のベクトル  $U_\mu^{ij}(n)$  として表される(図 2)。解くべき運動方程式は以下のような疎行列  $D$  を係数行列とする連立方程式  $Dx = b$  となる。この連立方程式を解く部分がクォー

クソルバーに該当する。

$$\sum_{j=1}^3 \sum_{\beta=1}^4 \left[ F_{\alpha,\beta}^{i,j}(n) x_{\beta}^j(n) - \kappa \sum_{\mu=1}^4 \left[ (1 - \gamma_{\mu})_{\alpha,\beta} U_{\mu}^{i,j}(n) x_{\beta}^j(n + \hat{\mu}) + (1 + \gamma_{\mu})_{\alpha,\beta} (U_{\mu}^{j,i}(n - \hat{\mu}))^* x_{\beta}^j(n - \hat{\mu}) \right] \right] = b_{\alpha}^i(n),$$

$$\Leftrightarrow Dx = b, D = \left( D_{\alpha,\beta}^{i,j}(n, m) \right), x = \left( x_{\alpha}^i(n) \right), b = \left( b_{\alpha}^i(n) \right).$$

ここで、 $n, m$ は四次元格子点上のサイト座標、 $\hat{\mu}$ は四次元の $\mu$ -方向の単位ベクトル、 $i, j$ はカラーチャージ番号、 $\alpha, \beta$ はスピン番号である。係数 $(1 \pm \gamma_{\mu})_{\alpha,\beta}$ は $4 \times 4$ の定数行列、 $F_{\alpha,\beta}^{i,j}(n)$ は $O(a)$ 改良のための係数、 $\kappa$ はクォークの質量に関するパラメータである。この係数行列 $D$ の条件数はクォークの質量がゼロとなるときに発散する。実数の連立方程式として考えたときのベクトル次元は、四次元の一辺のサイト数が $N$ のとき $2 \times 3 \times 4 \times N^4 = 24N^4$ 次元となる。例えば格子サイズを[参考文献 9]にて用いた $N^4 = 96^4$ とすれば、その次元は約20億次元となる。この連立方程式は並列化された反復法を用いて数値的に解く必要がある。

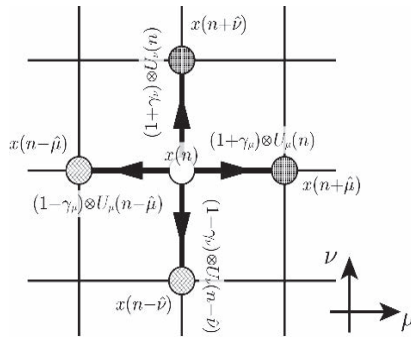


図2：ステンシルの構造。

反復法で最も計算時間を要する部分が、行列ベクトル積の部分である。クォークソルバーではクォークを表すベクトル $w$ に係数行列 $D$ をかける演算 $v = Dw$ がこれに該当する。以下ではこれを MULT と呼ぶ。格子 QCD での並列化は一般に四次元格子点を均等に領域分割して行う。したがって MULT の部分も四次元格子点分割されて実装される。並列計算を担当する計算ノードはそれぞれ四次元格子点の分割された一部の計算を担当する。MULT は四次元格子点に対してのステンシル計算であるので、並列化によりノード間でのステンシルデータの通信が必要となる。MULT への最適化は大まかにノード内でのステンシル計算の最適化と並列化による通信を最適化する部分に分類される。

## 2-2. 最適化手法

$O(a)$ 改良ウィルソンクォークの係数行列は単純なステンシル計算であるため、コード上の局所的な最適化のみで達成できる計算速度の改良は、1 サイトのステンシル計算に必要なデータ量と計算量の比であるいわゆる Byte/Flop から強い制限を受ける。このため、コードの局所的な最適化に加え、全体の並列化を見越した最適化や、連立方程式を解くためのアルゴ



リズム上の改良も非常に重要となる。我々は筑波大学計算科学研究センターの Intel Parallel Computing Center の協力のもと、クォークソルバーの OFP 向けへの最適化として以下の最適化と改良を行った。

- ① 精度混合型二重 BiCGStab 法を用いた。これは、倍精度の BiCGStab 法に、連立方程式の前処理として単精度の BiCGStab 法を組み込んでいる。ほとんどの計算を前処理である単精度計算に費やすことでキャッシュの有効利用, Byte/Flop やメモリバンド幅, 通信バンド幅の削減により性能向上できる。
- ② すべての単精度計算には、インテル C/C++コンパイラの AVX-512 のイントリンシック関数を用いた SIMD ベクトル化を施した。単精度 16 成分の SIMD ベクトル化による格子サイズへの制限を緩和するために、時間方向にひとつとびに連続する 4 格子点分のスピン 2 自由度に対応する 8 つの複素数がメモリ上で連続になるようにデータを並べた (図 3)。

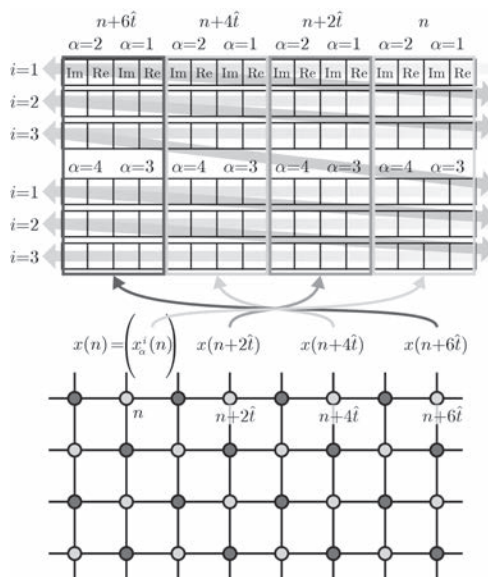


図 3 : 2 色塗り分け格子点(下段)と SIMD ベクトルに最適なデータレイアウト (上段)の関係。

- ③ 四次元ループを比較的小きなループ ( $2 \times 2 \times 4 \times 8 = x \times y \times z \times t$ ) にブロック分割し, 1 ブロックに 1 OpenMP スレッドを割り当てた。 $t$ 方向のブロック番号については SMT 並列になり $x, y, z$ 方向のブロック番号については物理コア並列になるようにスレッドを割り当てた。更にブロック内の四次元ループ内にはソフトウェアプリフェッチを挿入した。この小さなループ ( $2 \times 2 \times 4 \times 8$ ) 内でのステンシル計算に必要なデータ量はおよそ 336KiB となり, Xeon Phi の物理コア 1 つのもつ L2 キャッシュサイズ 512KiB に収まる (Xeon Phi では物理コア二つが 1 タイルを構成し 1MiB の L2 キャッシュを共有している)。OpenMP スレッドには Workstealing 法[参考文献 10]を用いて動的にスレ

ッド間のロードバランスを平滑化した。これらの最適化により Xeon Phi の有する多くのコアとキャッシュを有効活用できる。

- ④ 並列化は、格子を均等に領域分割し MPI により並列化する。MULT の計算を表面領域のサイトデータを送信バッファに詰める計算 (MULT\_PRE), 内側領域のステンシル計算 (MULT\_IN), 表面領域のステンシル計算 (MULT\_PST) に三分割し, 内側領域の計算と表面領域の通信を並行に行い通信時間の隠蔽を行った (図4)。通信と計算の並行性を高めるため MPI 通信用の特別なスレッドを Xeon Phi の 0 番目のタイルの 2 コアに pthread 機能により割り当てた。MPI 通信用スレッドは OpenMP の制御外になるようにインテル MPI の実行時環境変数 (I\_MPI\_PIN\_PROCESSOR\_EXCLUDE\_LIST) により指定し並行性を高めた。また, OFP の通信ハードウェア Intel Omni-Path ネットワークの持つネットワークバンド幅を最大限活用するため 1 ノードに 2 MPI プロセスを実行するようにした。

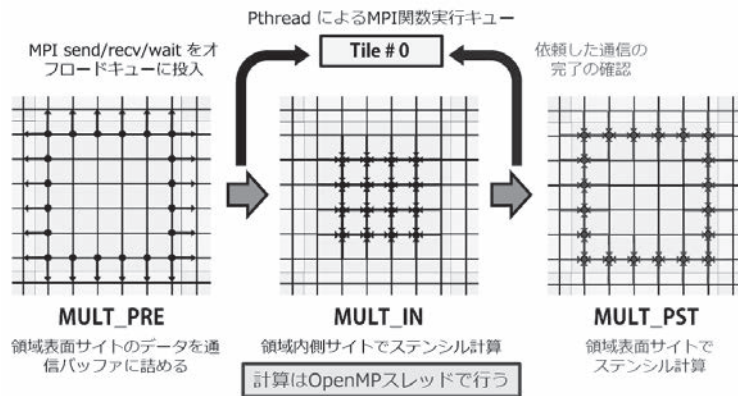


図4：内側領域の計算と表面領域の通信のオーバーラップによる通信時間の隠蔽。

### 2-3. ベンチマークの結果

以上の最適化を施したクォークソルバーのベンチマークを OFP 試用期間の終わりの 2017 年 3 月に行った。ベンチマークは (1) 128MPI ランクまでの低並列度での強・弱の両スケーリングの測定と (2) 1 MPI ランクの格子サイズを  $20^3 \times 400$  に固定した 2000MPI ランクから 16000MPI ランクまでの弱スケーリングの測定を行った。スケーリングのベンチマークを行うため、クォークソルバーの行列ベクトル積の回数を 2000 回に固定した。以下に (1) (2) それぞれのベンチマークの結果を紹介する。

#### (1) 128MPI ランクまでの強・弱スケーリング

図5に MPI ランクあたりの空間格子サイズを  $12^3$  (または  $24^3$ ) に固定したまま時間格子サイズとランク数を変化させた様子を示す。性能は MULT の計算量を MULT\_PRE 開始から MULT\_PST 終了までの時間で除した性能であり、通信の時間も含んでいる。MPI ランクの四次元並列分割 ( $x \times y \times z \times t$  方向) への割り当ては以下の 9 通りである： $1 = 1 \times 1 \times 1 \times 1, 2 = 2 \times 1 \times 1 \times 1, 4 = 2 \times 2 \times 1 \times 1, 8 = 2 \times 2 \times 2 \times 1, 16 = 2 \times 2 \times 2 \times 2, 16 = 4 \times 2 \times 2 \times 1, 32 = 4 \times 4 \times 2 \times 1$

1, 64 = 4 × 4 × 4 × 1, 128 = 4 × 4 × 4 × 4。

MPI ランクあたりの格子サイズを固定したまま MPI ランク数を変化させたときのスケーリング性能（弱スケーリング）は、空間格子サイズが小さい場合（図5左）は 8MPI プロセス以上で性能が一定となり、大きい場合（図5右）は 2MPI プロセス以上で性能が一定となっている。ランク数の変化による性能の変化は、最適化④で行った通信と計算を並行に行う最適化の効き具合により様子が異なる。このベンチマークの結果は空間格子サイズが小さい場合（図5左）は通信時間を計算時間で隠蔽できないため待ち時間が発生しているが、大きい場合（図5右）はほぼ隠蔽できていることを示している。通信時間を隠蔽するためにはおおよそ 1 MPI ランクあたり  $24^3 \times 24$  以上の格子サイズを割り当てれば良いことがわかる。

MPI ランク数を固定したままランクあたりの時間格子サイズを 24 から 128 まで増加させたスケーリング性能（強スケーリング）は、空間格子サイズが小さい場合（図5左）は、時間格子サイズの増加とともに増加しており、大きい場合（図5右）は時間格子サイズが 48 から 64 以上で性能が変化しなくなっている。良い強スケーリング性能を引き出すためには 1 MPI ランクあたり  $24^3 \times 48$  から  $24^3 \times 64$  以上の格子サイズを割り当てれば良いことがわかる。

MPI プロセス数が 16 の場合には時間方向の並列度が 1 の場合と 2 の場合がある。性能の低いほうが時間方向に並列度 2 の場合である。最適化②で述べたように時間方向の格子座標は SIMD ベクトルのベクトル化に使われており、時間方向の並列化は SIMD ベクトルへの特別な処理が必要になるため性能が悪化する。

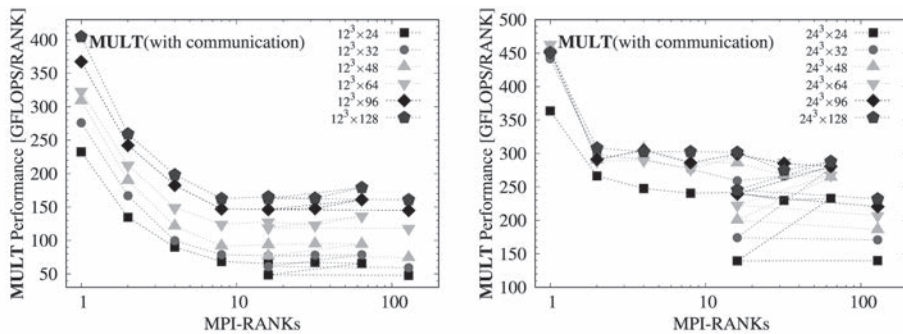


図5：MULT(通信時間を含む)の性能。MPI ランクあたりの空間格子サイズが  
（左） $12^3$ の場合と（右） $24^3$ の場合

## (2) 16000MPI ランクまでの弱スケーリング

MPI ランクあたりの四次元格子サイズを  $20^3 \times 400$  に固定し、MPI ランク数を 2000 から 16000 まで変化させたときの実行時間を図6に示した。

MULT は行列ベクトル積に費やした時間、COMM はステンシル計算のための隣接通信完了待ち時間、ALLREDUCE は反復法のベクトルの内積計算に必要な MPI\_Allreduce の時間、OTHRES はその他のベクトル計算である。④の最適化により COMM の時間は、ほぼ MULT の計算時間に隠れてしまい、この図では見えていない。全体の計算時間はランク数を増やしていくと徐々に増えているが、その原因は主に MPI\_Allreduce によるリダクション通信時間の増加で説明できる。しかしながら図のように 16000MPI ランク（8000 ノード）まで概ね良く弱スケーリングするよ



うな最適化ができていることがわかる。

MULT のノードあたりの計算性能はおよそ単精度で 560GFLOPS (MPI ランクあたりは 280GFLOPS) となった。16000 MPI ランク実行時のクォークソルバー全体での単精度計算性能は約 2.6PFLOPS を記録した。

## 2-4. 最適化のまとめ

ベンチマーク計測した最大の四次元格子サイズは $400^3 \times 800$ であり、このベンチマークにより将来の大体積・微細格子での大規模計算に向けての計算科学的知見を得ることができた。この最適化されたクォークソルバープログラムはすでに OFP 上での大体積 master-field シミュレーションに使われている。本稿で紹介したクォークソルバーの最適化とベンチマークの結果の詳細については[参考文献 11]をご覧ください。

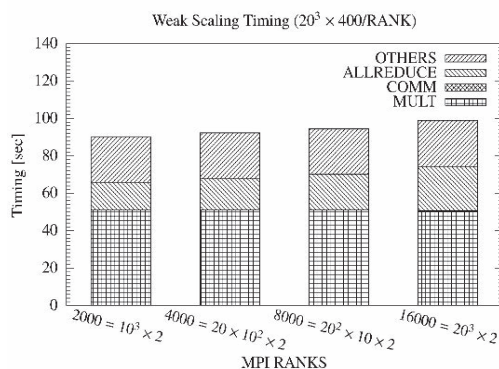


図6：弱スケーリングベンチマークの実行時間内訳。

## 3. 2+1 フレーバーQCD の master-field シミュレーション

OFP を用いた master-field シミュレーションプロジェクトでは、物理点において $(11 \text{ fm})^4$ 程度の超大体積を持つゲージ配位(モンテカルロサンプル)生成を行う。ゲージ配位は異なる格子間隔 3 点( $a=0.085 \text{ fm}$ ,  $0.065 \text{ fm}$ ,  $0.045 \text{ fm}$ , 対応する格子サイズは各々 $128^4$ ,  $160^4$ ,  $256^4$ )において生成し、系統誤差となる格子間隔依存性を取り除くために連続極限( $a \rightarrow 0$ )をとる。現在、最も細かい格子間隔  $a=0.045 \text{ fm}$ (格子サイズ  $256^4$ )においてゲージ配位を生成中であり、並行して粗い格子間隔 2 点( $a=0.085 \text{ fm}$ ,  $0.065 \text{ fm}$ )において幾つかの物理量計算を行っている。以下では、物理量計算の代表例として核子形状因子に基づく核子半径の計算について格子間隔  $a=0.085 \text{ fm}$  での結果[参考文献 12]を報告する。

陽子の電荷半径は極めて基本的な物理量でありながら、現在素粒子・原子核物理分野において多くの研究者の関心を引きつけている。陽子の電荷半径は、1950 年代から電子-陽子(ep)散乱を用いて陽子の電気形状因子を測定することにより、その  $q^2$ (e-p 間での移行運動量)依存性から電荷半径が評価されてきた(陽子半径は  $q^2 \rightarrow 0$  での核子形状因子の  $q^2$  依存性で定義される)。格子 QCD による電荷半径の計算も電気形状因子計算に基づいている。陽子の電荷半径が多くの研究者の耳目を集めたのは、2010 年の  $\mu$  水素原子分光(陽子の周りの  $\mu$  粒子軌道のエネルギーシフト、 $\mu$  水素原子は人工的に生成)の精密測定実験の結果[参考文献 13]に端を発し

ている。 $\mu$  水素原子分光の結果は ep 散乱の結果と 4%、誤差にして  $7\sigma$  も乖離しており ( $7\sigma$  は統計的に 99.999999997%の確率で両者が異なっていることを意味している)、この差は非常に大きい。現在その解決に向けて世界各国で多数の追試実験が実施・計画されている。もし、実験的にどうしてもその違いが説明ができない場合は、電子と  $\mu$  粒子の間に質量以外の相違があることを意味しており、それは標準理論を超える新たな物理の存在を意味する。格子 QCD は、これまでの計算を精密化することにより、標準理論として ep 散乱と  $\mu$  水素原子分光のどちらの実験結果が正しいか(標準理論の結果と整合しているか)を判定することができる。図 7 (左)は 2014 年当時の国際会議「Lattice 2014」(毎年世界各国持ち回りで開催され、格子 QCD 研究者が一堂に会する国際会議)における核子半径計算のレビュー講演において、世界中の各グループの計算結果を纏めたものである[参考文献 14]。実験値(\*, ×シンボル)に対して、格子 QCD 計算の結果は 40%~50%程度小さな値しか出せていない。図 7 (右)は、現在われわれが Oakforest-PACS を用いた master-field シミュレーションによって得た核子半径計算の結果(丸印)を実験結果と比較したものである[参考文献 12]。上側の横帯は ep 散乱実験の結果(帯幅は誤差の大きさ)を表しており、下側の横帯は  $\mu$  水素原子分光を用いた実験結果を示している。4 つのデータは、計算結果の解析方法依存性を表している。図 7 (左)と比較すれば、master-field シミュレーションによる格子 QCD 計算の劇的な改善が確認できる。われわれの結果は  $\mu$  水素原子分光の実験結果を支持しているようにも見えるが、科学的に決定的な結論を導くために 1%程度まで誤差を縮減することを目指している。

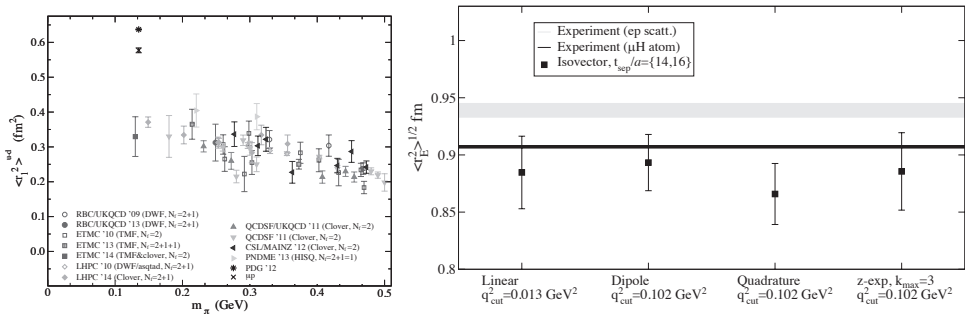


図 7：格子 QCD による核子半径計算。

左図は 2014 年時点での世界各国のグループの計算結果のまとめ[参考文献 14]。\*印：ep 散乱実験の結果，×印： $\mu$  水素原子分光実験の結果。右図は master-field シミュレーションによるわれわれの計算結果[参考文献 12]。上帯(グレー)：ep 散乱実験の結果，下帯(黒)： $\mu$  水素原子分光実験の結果。

2018 年 2 月末の大規模 HPC チャレンジでは、格子間隔  $a=0.045$  fm(格子サイズ  $256^4$ )のゲージ配位生成をシステム全系を用いて実行することによって、大規模計算における課題を把握することができた。現在は、実行効率やジョブスループットなどの観点から 2048 ノードジョブを用いて格子間隔  $a=0.045$  fm のゲージ配位を生成中である。格子間隔 3 点( $a=0.085$  fm,  $0.065$  fm,  $0.045$  fm)のそれぞれにおいて物理量計算を行った後、連続極限へ外挿( $a \rightarrow 0$ )することによって、系統誤差である格子間隔依存性を取り除くことを目指して。

## 参考文献

1. K.G.Wilson, “Confinement of quarks”, Physical Review D **10**, 2445 (1974).

2. CP-PACS プロジェクトの経緯は、以下の URL 下の資料に詳しい。  
[https://www.ccs.tsukuba.ac.jp/research\\_project/cp-pacs/](https://www.ccs.tsukuba.ac.jp/research_project/cp-pacs/)
3. CP-PACS Collaboration: S.Aoki, G.Boyd, R.Burkhalter, S.Ejiri, M.Fukugita, S.Hashimoto, Y.Iwasaki, K.Kanaya, T.Kaneko, Y.Kuramashi, K.Nagai, M.Okawa, H.P.Shanahan, A.Ukawa, and T.Yoshie, “Quenched light hadron spectrum”, *Physical Review Letters* **84**, 238 (2000); “Light hadron spectrum and quark masses from quenched lattice QCD”, *Physical Review D* **67**, 034503 (2003).
4. S.Duane, A.D.Kennedy, B.J.Pendleton, and D.Roweth, “Hybrid Monte Carlo”, *Physics Letters B* **195**, 216 (1987).
5. PACS-CS Collaboration: S.Aoki, K.-I.Ishikawa, N.Ishizuka, T.Izubuchi, D.Kadoh, K.Kanaya, Y.Kuramashi, Y.Namekawa, M.Okawa, Y.Taniguchi, A.Ukawa, N.Ukita, and T.Yoshie, “2+1 flavor lattice QCD toward the physical point”, *Physical Review D* **79** 034503 (2009); PACS-CS Collaboration: S.Aoki, K.-I.Ishikawa, N.Ishizuka, T.Izubuchi, D.Kadoh, K.Kanaya, Y.Kuramashi, Y.Namekawa, M.Okawa, Y.Taniguchi, A.Ukawa, N.Ukita, T.Yamazaki, and T.Yoshie, “Physical point simulation in 2+1 flavor lattice QCD”, *Physical Review D* **81** 074503 (2010).
6. M.Lüscher, “Lattice QCD and the Schwarz alternating procedure”, *High Energy Physics* **0305**, 052 (2003); “Schwarz-preconditioned HMC algorithm for two-flavor lattice QCD” *Computer Physics Communication* **165**, 199 (2005).
7. J.C.Sexton and D.H.Weingarten, “Hamiltonian evolution for the hybrid Monte Carlo algorithm”, *Nuclear Physics B* **380**, **665** (1992).
8. I.P.Omelyan, I.M.Mryglod, and R.Folk, “Symplectic analytically integrable decomposition algorithms: classification, derivation, and application to molecular dynamics, quantum and celestial mechanics simulations”, *Computer Physics Communication* **151**, 272 (2003).
9. PACS Collaboration: E.Shintani, K.-I.Ishikawa, Y.Kuramashi, S.Sasaki, and T.Yamazaki, “Nucleon form factors on a large volume lattice near the physical point in 2+1 flavor QCD”, *Physical Review D* **98**, 074510 (2018).
10. L.Meadows, S.J.Pennycook, A.Duran, T.Wilmarth, J.Cownie, “Workstealing and Nested Parallelism in SMP Systems”, In: *OpenMP: Memory, Devices, and Tasks. IWOMP 2016*, N.Maruyama, B.de Supinski, M.Wahib (eds), *Lecture Notes in Computer Science*, vol 9903. Springer, Cham.
11. T.Boku, K.-I.Ishikawa, Y.Kuramashi and L.Meadows, “Mixed Precision Solver Scalable to 16000 MPI Processes for Lattice Quantum Chromodynamics Simulations on the Oakforest-PACS System,” 2017 Fifth International Symposium on Computing and Networking (CANDAR), Aomori, 2017, pp. 362-368. doi: 10.1109/CANDAR.2017.69.
12. PACS Collaboration: E.Shintani, K.-I.Ishikawa, Y.Kuramashi, S.Sasaki, and T.Yamazaki, “Nucleon and root-mean-square radii on a  $(10.8 \text{ fm})^4$  lattice at the physical point”, in preparation.
13. R.Pohl *et al.*, “The size of the proton”, *Nature* **466**, 213 (2010).

14. M.Constantinou, “Hadron structure” , Proceedings of Science (LATTICE2014) 001 (2014).