

「サブAB課題用格子QCDプログラム改良報告」

石川健一
金森逸作
(広大理)

取り組んでいること(H27-H28)

- コデザインとは別に以下のアルゴリズム、数値計算のための改良・作業に取り組んできた。
 - サブ課題A: ドメインウォール型フェルミオン作用のソルバー
 - 重いクォーク(ボトムやチャーム)に関する物理の格子QCDによる精密計算
 - 格子カイラル対称性があるフェルミオン作用が精密計算に向いている(理論的に良い性質)
 - ただし、計算量は下記のウィルソン型に比べ多い。
 - サブ課題B: ウィルソン型フェルミオン作用のソルバー

取り組んでいること(H27-H28)

- コデザインとは別に以下のアルゴリズム、数値計算のための改良・作業に取り組んできた。
 - サブ課題A: ドメインウォール型フェルミオン作用のソルバー
 - サブ課題B: ウィルソン型フェルミオン作用のソルバー
 - 格子QCDで長年使われてきた作用。
 - 大規模な計算に使えるが格子カイラル対称性が無いため繰り込みが面倒。
 - バリオン間相互作用の計算に必要な大体積 ($>(10\text{fm})^4$) での計算に使われる。

取り組んでいること(H27-H28)

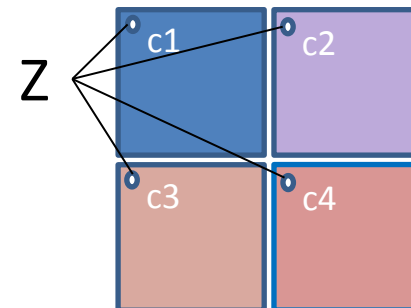
- コデザインとは別に以下のアルゴリズム、数値計算のための改良・作業に取り組んできた。
 - サブ課題A: ドメインウォール型フェルミオン作用のソルバー
 - 金森逸作氏と松古栄夫氏(KEK)らによって開発中。
 - Oakforest-PACS (KNL) 機向けのドメインウォール型フェルミオン作用のソルバーの開発を行っている。
 - フェルミオン場の線形代数に P.Boyle 氏(エジンバラ大)らの開発している Grid++ を援用して実装中
 - 分野5の際に作成した格子QCD汎用コード Brige++ にソルバーを組み込むことを予定している。

サブ課題A: KNL機 (OakForest PACS) でのQCD

ベクトル変数の有効活用が鍵



simd vector: 512 bit
(double complex が4個入る)



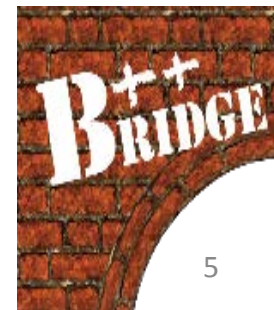
方針

- ベクトルの要素: 内部自由度ではなく、空間自由度に使う
 - 例: フェルミオン (spin × color = 12 自由度)
 - 12個のvector変数で4つの格子点上のフェルミオンを同時に扱う。
- 差分がベクトル要素間の演算になるのは極力避けたい
 - 各ノードが担当する格子を4つの領域に分け、そこにばら撒く

実装

- 低レベル
 - GRID (<https://github.com/paboyle/Grid> by P.Boyle et. al)
 - QCD simulation の開発中のフルコード、一部を借用:
 - simd 変数に対する文法糖衣、線形演算、データ構造 (格子の構造)
 - KNLノウハウ蓄積のために別の独自SIMD対応コードセットも準備中
- 高レベル **Bridge++** (<http://bridge.kek.jp/Lattice-code/>)
 - アルゴリズム、Dirac演算子の実装
 - Oak Forest PACS 向けの **Domain wall** フェルミオンコードを準備中

既存の simd コード? 要チューニング、ノウハウが蓄積できないでも 1 から simd 対応のコードセットを作るのは大変



取り組んでいること(H27-H28)

- コデザインとは別に以下のアルゴリズム、数値計算のための改良・作業に取り組んできた。
 - サブ課題B: ウィルソン型フェルミオン作用のソルバー
 - 京用HMCコードのクオーク型ソルバーのより高速化をしたい
 - 京上ではチューニングはしつくされていて
 - Mixed precision + D-D SAP + SSOR + inner/outer BiCGStab + HPC-ACE-intrinsics
 - 効率20%
 - さらなる高速化を行うためには、アルゴリズム的な変更や、解きたい問題に合わせた変更が必要。
 - ゲージ配位生成でなく、測定計算の時に必要なクオーク伝搬関数計算への高速化要求がある。
 - ゲージ場を固定したまま(=係数行列固定)、ひたすら右辺ベクトルを取り換えて解く。(統計精度向上)
- 測定用コードに使われるソルバー部のアルゴリズムの検討

サブ課題B:線形ソルバー(ウィルソクローバーフェルミオン)

石川健一, 金森逸作, 「格子QCDにおける軽いモードを取り除いた前処理付きクォークソルバーの開発」
日本物理学会@2016年秋の学会、2016年年会(春)

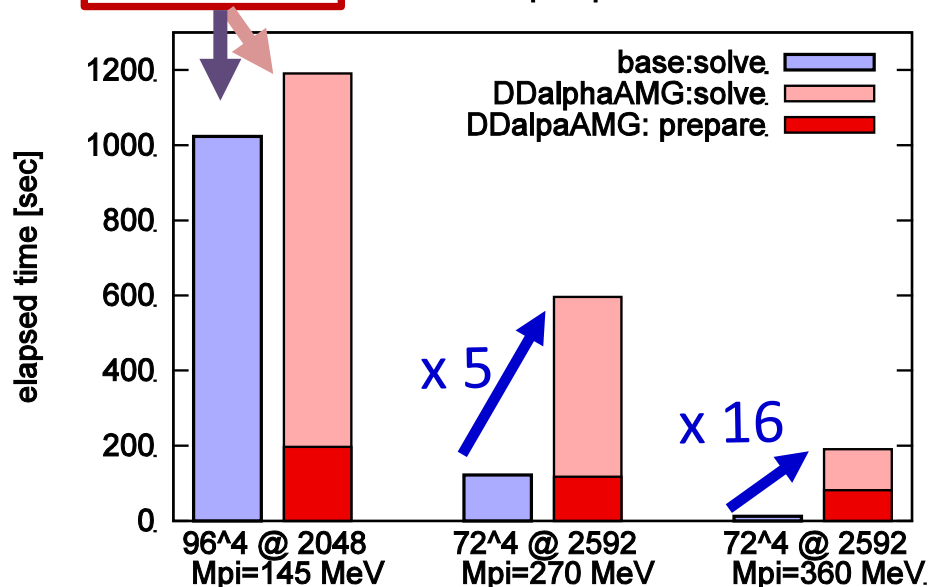
- 最近の流行 : Algebraic Multi Grid (代数的マルチグリッド法、AMG)
 - 適当な基底で小さな演算子を作って前処理に利用
 - 軽いモードを多く含むよう、基底の取り方を工夫
 - 再帰的にさらに小さな演算子も作れる("Multi" grid)

実装例 : DD α AMG <https://github.com/DDalphaAMG/DDalphaAMG>

M.Rottmann et. al

ほぼ同じ時間
(少し早い)

24 solves + prepare time



これを京用のコード(LDDHMC)から呼べるようにしました

軽いクォーク(Mpi=145MeV)でほぼ同じ時間で解けている(最左棒グラフ)。

solve (疎格子)部の実行効率がまだ悪い

Mixed Prec. BiCGstab(base): ~20%

DD α AMG: ~2-3%

x 1/7 ~ 1/10

チューニングすれば既存(base)より速くなる可能性がある。

要チューニング

ベンチマーク(24 solves)の結果

2017/2/17

素粒子・原子核・宇宙「京からポスト京に向けて」シンポジウム

そのほか

- JLDGクライアント(広島サイト)再立ち上げ(金森さん)
- Twisted Gradient Flow を用いた純SU(3)ゲージ理論の Λ パラメータの導出

Ken-Ichi Ishikawa, Issaku Kanamori, Yuko Murakami, Ayaka Nakamura, Masanori Okawa, Ryoichiro Ueno,
PoS(LATTICE2016) 185