

コヒーレント X 線によるスペックル像の高速計算

物質系専攻 47-116042 太田崇士

指導教員：雨宮慶幸(教授)

キーワード：コヒーレント X 線 スペックル像 XPCS 並列計算

緒言

コヒーレントな X 線が試料に入射すると、試料から生じる散乱波が干渉してスペックル像と呼ばれる散乱像が得られる[1]。これは小角 X 線散乱法で得られるような空間的に平均化されたなだらかな像と異なり、試料中の粒子配置などといった構造情報を強く反映している。こうして得られるスペックル像の時間変化を用いてサンプル中の粒子運動などのダイナミクスを測定する手法として XPCS がある。XPCS はスペックル像の時間揺らぎから散乱強度の時間相関関数をサンプルのダイナミクスと結びつけることで構造解析を行うが、複雑な粒子運動とスペックル像の関連については未解明の部分が多い。

近年、コンピュータ技術の発展により、これまで困難であった大規模な物理シミュレーションが可能になりつつある。かつてはスーパーコンピュータに頼らざるを得なかった計算でも、近年普及が進む GPGPU[2]やメニーコアプロセッサ[3]といった並列計算機を用いることで、研究室レベルでも容易に高速な計算が行われるようになった。並列計算とはプロセッサを複数用意することで複数の計算を一度に行う手法であり、スーパーコンピュータの高速化にも用いられている高速化手法である。一方で並列計算は全ての計算に対し適用できるわけではなく、計算機に適したアルゴリズム選択や最適化が不可欠である。

XPCS で測定されるデータは、サンプルのある瞬間の配置から得られるスペックル像を長時間にわたって連続撮影したものであり、これをコンピュータで計算するためには「撮影時間間隔ごとの構造変化のシミュレーション」「サンプル情報を用いたスペックル像の生成」の二つの過程が必要になる。将来的には前者について MD 計算などの手法を取り入れることで、複雑な粒子運動の系で XPCS の解析が可能になると期待できる。本研究では後者のスペックル像計算について、並列計算機による高速計算の可能性について検討を行った。サンプル規模が大きくなるほどスペックル像計算にかかる時間は増大することが予想され、これを高速に行うことは今後の XPCS の発展に重要な役割を果たすと考えられる。なお、本研究において用いるサンプルは粒子分散系を仮定している。

スペックル像計算プログラムの作成と考察

本研究では計算機として Fujitsu Server PRIMERGY TX300 S8 を用いた。これは CPU である Intel Xeon E5-2650 を 2 つ持つのとは別に、並列計算用のマルチコアプロセッサである Intel XeonPhi 5110p を内蔵する。コプロセッサはコンピュータ等に接続して補助的に動作する計算機である。XeonPhi 5110p は 60 個のコアによる並列計算を特徴とする。倍精度浮動小数演算における理論ピーク性能は 1.011 TFLOPS である。

XPCS で用いるスペックル像は Q 空間上で定義される散乱像であり、散乱振幅 $E(\mathbf{q})$ から求めることができる。光が完全コヒーレントであると仮定した場合、 $E(\mathbf{q})$ は電荷密度分布 $\rho(\mathbf{r})$

から

$$E(\mathbf{q}) = \int \rho(\mathbf{r}) \exp(-i\mathbf{q} \cdot \mathbf{r}) d\mathbf{r}$$

の式で求めることができる。小角 X 線散乱の場合と異なるのは積分範囲がビーム照射範囲全体に及ぶことであり、小角 X 線散乱では構造因子として考慮されるかまたは無視されていた粒子配置の情報が散乱像に反映されることになる。ここから粒子の散乱による影響を取り除くと、粒子配置に依存する部分として次の式を得る。

$$E(\mathbf{q}) = \sum_n^N \exp(-i\mathbf{q} \cdot \mathbf{r}_n)$$

この式を用いると、計算対象とする全粒子、全 \mathbf{Q} 空間座標に対し指数関数計算を行うことでスペックル像を求めることができる。

上式を基にスペックル像を計算するプログラムを作成し、XeonPhi 5110p による計算を行った。例として波長 1\AA 、ビームサイズ $5\ \mu\text{m} \times 5\ \mu\text{m}$ の正方形ビームを用いて、直径 $100\ \text{nm}$ の球が体積分率 10% でランダムに分布したサンプルを厚さ $100\ \mu\text{m}$ で測定すると仮定した場合のスペックル像について計算を行った。この場合の粒子数は 4.8×10^5 である。 \mathbf{Q} 空間座標はピクセルサイズを $2 \times 10^5\ \text{m}^{-1}$ として 500×500 ピクセルに設定した。以上に示した系の場合、スペックル像生成にかかる時間は 1 枚あたり $22.3\ \text{sec}$ となった。同様の計算を PRIMERGY TX300 S8 上で XeonPhi を使わずに行う場合は $100\ \text{sec}$ ほどかかることから、XeonPhi の利用は計算時間短縮に有効であることがわかる。計算時間は Fig.1-2 に示す通り粒子数と \mathbf{Q} 空間座標数に比例した。

高速計算にあたって「並列化」「ベクトル化」「演算精度の制御」の 3 点を特に重視した。並列化とベクトル化はともに複数の計算を同時に実行する手法であり、採用するアルゴリズムにより適用可能かどうか異なる。本研究で採用したアルゴリズムでは、粒子ごとに計算する値 $\exp(-i\mathbf{q} \cdot \mathbf{r})$ が別の粒子や座標に依存せず計算可能であるため、並列化もベクトル化も適用可能である。これを用いておよそ 100 倍の高速化が達成された。精度については、計算精度をあえ

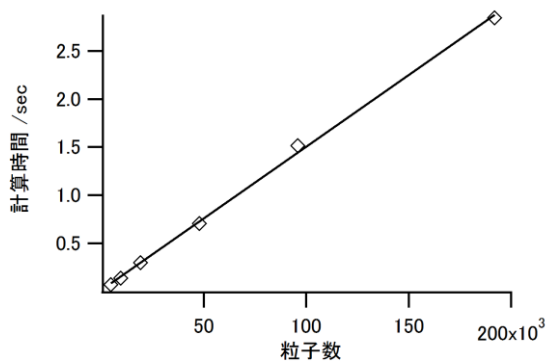


Fig.1 粒子数を変化させた場合の実行時間変化。 \mathbf{Q} 空間座標数は 2.5×10^5 に固定した。

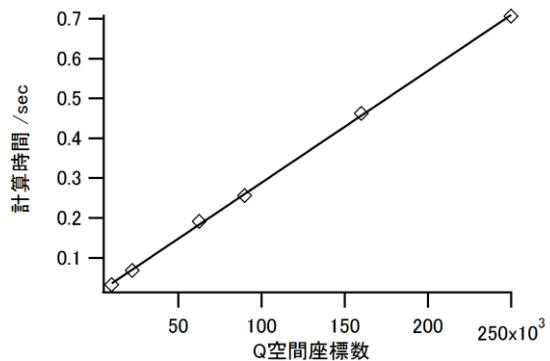


Fig.2 粒子数と \mathbf{Q} 空間座標数を変化させた場合の実行時間変化。粒子数は 4.8×10^4 に固定した。

て低下させることにより計算速度を向上させることが可能であり、この手法の採用の可否について検討した。上記の計算例で倍精度浮動小数点から単精度浮動小数点へ変更すると有効数字は4桁程度となり、2倍程度の高速化が得られる。採用するかどうかはパラメータに応じて判断する必要がある。

本プログラムは並列計算を用いた高速化を行っているため、さらなる高速化が必要な場合、適切な言語による書き換えを行うことでスーパーコンピュータ上でも実行可能と考えられる。その場合の処理速度と実行時間の関係は Amdahl の法則[4]に従い、全体に占める並列化可能な処理の割合が大きいほど効率よく高速化できる。本プログラムの中でスペックル像を計算する部分は並列化可能であり、上で示した例ではデータ保存等の並列化不可能な処理時間が全体の 0.02% 程度である。粒子運動計算が並列化可能かつ計算時間が無視できると仮定した場合、理論性能 10PFLOPS のスーパーコンピュータ上で実行する場合は 1×10^4 倍の処理速度により 3×10^3 倍の高速化が見込まれる。

本研究で作成したプログラムがどの程度の規模の計算を行えるかについても検証・考察を行った。プログラムの実行範囲を決めている要素は主に「コプロセッサのメモリサイズの上限」「整数型の上限」がある。前者については、コプロセッサで計算を行う前にあらかじめ問題を分割し、それぞれを計算した後に和を取るという手順を経ることで回避することができる。後者については、より大きな数を扱える整数型を採用することで解決する。

現状、作成したプログラム上では粒子運動について非常に簡単な運動を仮定して計算を行っている。この部分は将来的により複雑な運動を再現するよう書き換えることもできる。今後の研究によってゴム中粒子をはじめとした未解明のダイナミクスについて XPCS により解明することが目標となる。

最後に理論ピーク性能と実行性能の比較について述べる。文献[5]の手法を用いて実効性能を評価すると、粒子数 4.8×10^5 、Q 空間座標数 500×500 のとき 0.032 TFLOPS となり、理論ピーク性能の 3.2% に留まる。これは並列化のオーバーヘッド等が主な原因として考えられる。

まとめ

XPCS に用いられる、コヒーレント光によるスペックル像を並列計算機を用いて高速に計算する手法を確立し、計算を行った。また作成したプログラムの演算精度や適用限界についても検証した。

参考文献

- [1]M. Sutton *et al.*, *Nature*, **352**(6336), 608-610 (1991).
- [2]J. Fung *et al.*, *Proceedings of the Pattern Recognition 17th International Conference on (ICPR'04) Volume 1*, 805-808 (2004).
- [3]L. Seiler *et al.*, *ACM Transactions on Graphics*, **27**(3), 18:1-18:15(2008).
- [4]G.M. Amdahl, *Proceedings of the April 18-20, 1967, Spring Joint Computer Conference*, 483-485 (1967).
- [5]V. Favre-Nicolin *et al.*, *Journal of Applied Crystallography*, **44**(3), 635-640 (2011).