

大規模 ALM シミュレーションへの HPC 技術の適用

Application of High Performance Computing methods to the Large-Scale Portfolio Simulation for Asset Liability Management

松尾 拓真、鳥居 秀行
Takuma Matsuo, Hideyuki Torii

ニューメリカルテクノロジーズ株式会社
Numerical Technologies Incorporated
<http://www.numtech.com/>

金融機関のリスクエクスポージャーの規模は自己資本比率規制 (Basel II) に基づいて制御されている。2007-08 年に起きた金融危機の結果、バーゼル銀行監督委員会 (BCBS) は従来型の市場リスクと信用リスクに加えてカウンターパーティ・クレジットリスク (CCR) を含む新しい規制を課そうとしている。次世代の資産負債管理 (ALM) は、こうした制約条件下での最大収益を目指さねばならない。現在の ALM に内在する欠陥を避けるには次の条件が必須である。(1) 現実の会計ルールに即した方法での個々の取引の勘定処理、(2) 時間経過に伴って進行する市場価格変動と個別企業の信用格付け変化のシナリオパス、(3) 各シナリオパスに対する数年間に及ぶ日次の洗い替え。そのような計算には大規模なシミュレーションが必要であるため、計算コストが高すぎてこれまでは現実的ではなかった。本研究では最新の HPC によってたとえ現在最大規模の銀行が対象であってもこうしたシミュレーションが実用化可能であることを示した。

The Basel II framework for regulatory capital sets the size limit of risk exposure. The 2007-08 financial crisis spotted counterparty credit risk (CCR) as well as traditional market and credit risk, which persuaded the Basel Committee of Banking Supervision (BCBS) to issue new proposals concerning the regulatory framework. Asset Liability Management (ALM) in the next generation should maximize profits within this limitation. To avoid potential failures of the current ALM process, the following must be achieved: 1) realistic accounting by each transaction, 2) market rates and corporate credit rating scenario paths proceed with time, 3) evaluation along each scenario for at least several years at daily resolution. Such calculations require large-scale simulations which have not been realistic so far, because they are computationally expensive. This research contribution showed that the latest high performance computing (HPC) makes such simulations feasible, even for the largest banks in today's world.

Keywords: ALM (Asset Liability Management), VaR (Value at Risk), EaR (Earnings at Risk), EDF (Expected Default Frequency), TP (Transfer Pricing), Basel II, CCR (counterparty credit risk), CVA (Credit Valuation Adjustment)



Fig.1 "Blind monks examining an elephant"

[Source: Wikipedia]

背景と目的

今日、金融機関が行っているさまざまなリスク管理の手法は古代インドの説話「盲人と象」(“The Blind Men and an Elephant”)に似ている。象 (= 保有ポートフォリオ) の全体を目で見ることができないので、象の一部だけを見て全体を想像するという隘路に陥っている。象を鳥の視点から見るのはあまりに高コストあるいは不可能なのだ。この高コスト仮説はあまりに広く信じられていて、規制当局ばかりでなく主流の経済学者さえも疑ってはいない。そこで複数の盲人 (= リスク管理手法) が目的に応じて活躍している。それはたとえば以下のようなものである。

(様式第 20)

(1) マチュリティーラダー

保有ポジションを maturity 軸に沿った一次元の残高表にする。maturity としては資金満期をあてる。残高としては名目元本 (notional amount) をあてる。これで金融機関にとって最も基本的な財務オペレーションであるファンディング用としては十分。つまり流動性リスクの管理になっている。しかしそれ以外の目的にはまったく不十分である。

(2) 静的 ALM

そこでマチュリティーラダーをさらに金利満期あるいはアセットクラスに応じて区分し、複数の表にまとめてみる。するとたとえば金利感応部分と非感応部分、安全性の高い資産とハイリスクな資産などに区分できる。(1)と(2)は商業銀行 (commercial bank) のすべてが採用している管理手法であり、ほぼすべての会計士が理解可能な管理手法であり、BIS 規制の基本を成している。しかし投資銀行 (investment bank) やリスク管理にはまったく不十分である。

(3) 動的 ALM

静的 ALM まで行った後、一定の期間毎に収益 (return) を残高で割ってリスクリターンを計算してみる。これで EaR (earnings at risk) を掌握できる。また将来の市場シナリオを複数用意してシミュレーションする。するとシナリオパス毎に将来の財務諸表ができあがる。こうして出来たパラレルワールドを確率で考えるならば期待収益あるいは期待 EaR である。この管理手法は非常に強力でビジネススクール出の管理職が好んで用いており、ボーナス金額の算出根拠などに使われる。しかしバンキング勘定 (banking account) はそれでよいが、トレーディング勘定 (trading account) 相手では意味をなさない。また大半の会計士にとって、確率論を含む数学はハイスクール以来避けてきた話題なので、ハードルが高すぎるのも問題である。

(4) ポジションシート

トレーダーたちはマチュリティーラダーの変種を使っている。たとえば確定利付債券 (fixed income bond) のトレーダーは資金満期の代わりに金利満期あるいはデュレーション (duration) を maturity

軸にとり、残高としては現在価値 (present value) や PV01 (ベースポイントあたりの価値変化) をあてる。単純な債券や株式から複雑なデリバティブに至るまでこの方式が拡張され広く応用されている。しかしこの方法の明白な欠点は異種商品との一体管理ができないことである。つまり一人のトレーダーのリスク管理はできるが金融機関全体のリスク管理としては不十分である。また時価を重視するあまり時間概念を欠いていて手法(1)の単純な機能すら果たさない。トレーダーたちは「それは repo desk の仕事だ」として無視している。

(5) VaR

個々のリスクファクターの間にある相関構造をモデル化することによって複数のポジションシートを統合する。こうすればどんなに複雑なエクスポージャーを抱えた金融機関が対象であっても単一の数値 VaR (value at risk) でリスク量を定量化できる。これが現行 Basel II framework の根幹たる「第 1 の柱」(Pillar I) である。当初の VaR は市場リスクのみ対象とする市場 VaR (market VaR) として定義されたが[1]、現在では信用リスクを対象とする信用 VaR (credit VaR) へと拡張された[2]。ところが明白な欠点として時価がなければ使えない管理手法なので、市場取引に基づいた現在価値 (mark to market) を得られない商品に対しては、価格モデルを使って計算した現在価値 (mark to model) を使わねばならない。ここが大間違いの温床になっている。それで 1998 年の LTCM 事件から 2008 年のベアスターンズ、AIG に至るまで、金融業界最高の頭脳とノーベル賞学者を巻き込んだ事故が繰り返り起きている。加えて流動性リスクの管理にはまったく不十分であり、会計士が理解できる数字でもない。したがって金融機関の財務責任者 (CFO) さえ今や VaR を単なる Basel Capital 引当用の数値とみなす傾向がある。

(6) CVA

デリバティブのトレーダーたちが従来行っているポジションシートの拡張仕様としてポテンシャルエクスポージャー (potential exposure) がある。これは将来の市場シナリオを複数用意してシミュレーシ

ョンし、シナリオパス毎に現在価値を計算し、その含み益側の期待値あるいは信頼区間に応じた値を計る手法である。将来の含み益は取引相手 (counterparty) に対する与信行為であるから取引相手の信用力が十分でなければ取引停止するなり担保 (collateral) を徴求する。ここに信用 VaR を持ち込んだのが CVA (credit value adjustment) である。取引毎に取引相手と自身の将来の信用格付けを信用 VaR と同じ手法を使って計算する。これを取引の maturity まで十分に密な時間間隔で計算する。そうして計算した取引相手毎の未来の信用 VaR を元にした規制資本付加が現在 BCBS で検討されている。2007-08 年金融危機への対策として主にデリバティブ業界が推している考え方である。したがって当然ながら商業銀行の管理手法とは整合しておらず、さらに言えば現在議論されている CVA 手法はトレーダー向けにアレンジされるあまり信用リスクの相関構造を無視する傾向にある。デリバティブトレーダーや格付け会社の多くは高等数学には熱心でも、信用リスクについてはローンオフィサー並みの常識さえ備えていないことが多い。モデルリスク (model risk) も大きい。まさにこれはサブプライム危機を生んだ CDO 価格モデルの状況と似ている。

立場が違えば人間が必要とするリスクの理解度にも差がある以上、このようにリスク管理手法が乱立する状態は将来も変わらないだろう。しかし取締役会あるいは規制当局の立場からすれば金融機関全体のひいては金融システム全体のリスク量を鳥瞰する手法が必要である。しかし象の全体を把握するのが非現実的なほど高コストであるからこれまでは行われてこなかった。しかし最新の HPC を使えばどうだろうか。そこで我々はマチュリティーラダーから CVA に至るリスク管理手法を内包するモデルを構築した。

計算モデル

我々が構築したモデルのあらまはは次の通りである。

- (1) 個々の取引を現実の会計方法で処理した。特に非線形なリスク特性を持つデリバティブや、大数の法

則が働きにくい法人向け取引や大口取引については個別件別をひとつずつ評価した。会計原則に基づいたことによりひとつの金融取引は単一の時価ではなく複数の勘定科目に仕訳される。Fig.2 に単一の取引が将来にわたる勘定仕訳に分解された様子を示したが、単純なキャッシュフロー表に比べて 1 桁大きな情報量になっていることが理解できる。ストラクチャード取引は、もちろんケースバイケースであるが、基本的にバックアセットに分解して名目元本を使って価格を追跡する。投資銀行が抱えがちな過剰な mark to model 方式による弊害はここで排除できる。

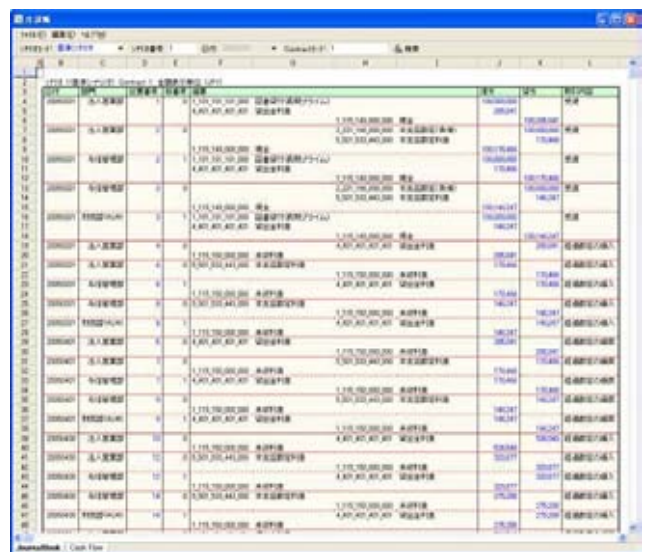


Fig.2 勘定仕訳に分解された取引

- (2) 市場レートと個別企業の信用格付けのシナリオパスをモンテカルロ法で生成した。

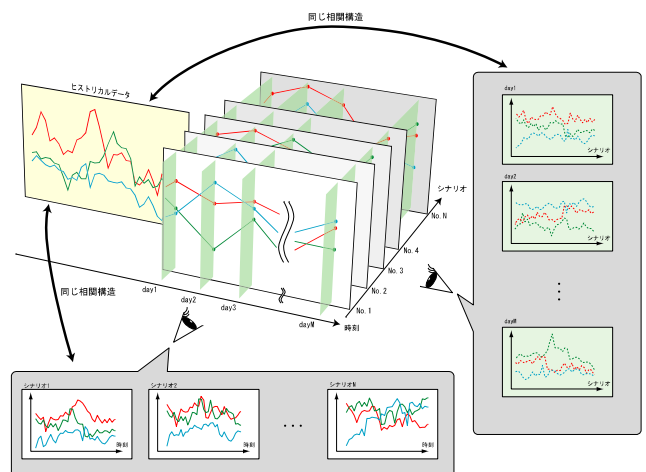


Fig.3 将来シナリオの 3 次元構造

ポートフォリオ全体を評価するためには市場リスクと信用リスクの相関構造を反映しなければならない。ここで VaR モデルとの大きな相違点は、タイムホライズンが将来の 1 時点なのではなく、連続した時間軸になっている点である。つまり VaR モデルに比べて次元数が 1 大きい。その様子を Fig.3 に示す。市場シナリオの生成においては主成分分析によって緩く制約をかけるに留めた。つまりデリバティブの価格モデルでよく使われるイールドカーブモデルとは異なる。これはリスク管理では標準的なイールドカーブモデルが破綻するような周辺事象をシナリオに加える必要があるからである。信用リスクは単なるデフォルト事象だけではなく日次の EDP(expected default probability)としてシナリオパス上で追跡し、Merton[1974]モデルを使って格付けを追跡した。Fig.4 にその様子を示す。つまり事実上すべての取引組み合わせにおける CVA を内部計算した。

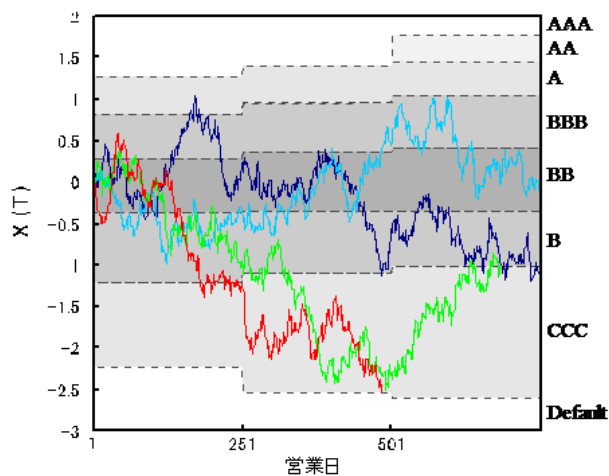


Fig.4 EDP の日次変化と格付け判定

- (3) 個々のシナリオパスに対して日次で数年間に及ぶ評価を行った。ここでモデルのアウトプットとして市場 VaR や信用 VaR を計算するだけならばタイムホライズン時点の市場レートと信用格付けを計算すればよい。また CVA を考慮するならば取引の満期まで計算を実行しなければならない。ALM が目的ならば数年間でよいがモデルのアウトプットは将来の present value ばかりでなく、Fig.5 に示すように将来の財務諸表となる。

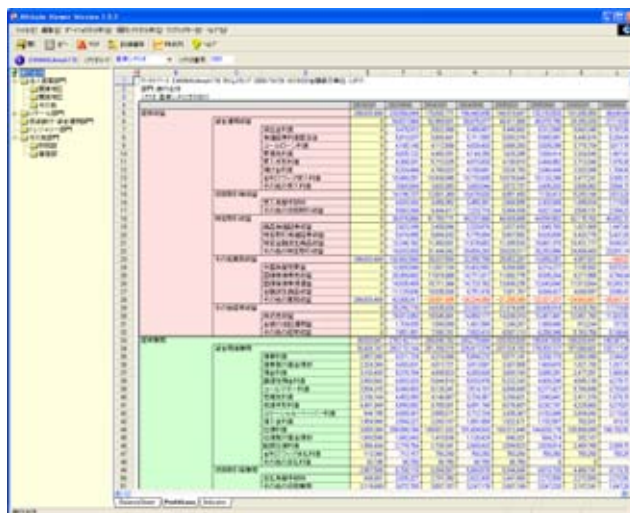


Fig.5 本モデルが出力した未来 Income Statement

入力データ

我が国最大のメガバンクを想定した場合はシミュレーションモデルに入力する取引規模が約 300 万件、シミュレーション日数は仮に 3 年間ならば約 1000 日、さらに実際の金融機関ではモンテカルロ法を 1 万回～10 万回程度実施した。総計算量はこれらの積になるため膨大な計算量となった。経営判断に要求される時間を考えればこの計算量を最大でも 24 時間以内に処理することが要求されるだろう。本研究では以上を想定して次の Table 1 の通りパラメータを決定した。

取引件数	約 340 万件
キャッシュフロー総数	約 5 億件
勘定処理数	約 44 億件/シナリオ
取引先相手数	約 20 万件
シミュレーション期間	3 年
モンテカルロ法の回数	1000 回～10 万回
使用ノード数(最大)	135 ノード/2160 コア
会計処理ルール	時価法 +アモチ・アキュム処理

Table.1 入力データ諸元

Table 1 がどれほど大きな計算量なのか理解を助けるために現時点で非常に計算コストがかかると言われる CVA と対比してみる。Morgan Stanley の Canabarro[3]によると、典型的な CVA システムにおいては、2 百万-1 千万件の取引に対して(勘定仕訳を考

えていないので本研究よりも 1 桁以上オーダーが小さい)、1000-2000 の市場パス(本研究比 1/10 規模)、タイムホライズンは 100 日分(本研究比 1/10 規模)、である[3]。これらすべてのファクターが乗算で寄与するので Morgan Stanley の総計算量は本研究比 1/1000 のオーダーとみなしてよいだろう。Morgan Stanley はこの計算のために大規模なグリッドコンピューティング方式で CVA システムを構築したと公言している。

初期の研究

本研究は平成 19 年度に実施した新規利用拡大の課題「銀行業・保険業における ALM (Asset Liability Management) システムの開発」として始まった。この段階では弊社アプリケーションである Numerical Technologies Altitude®を東京工業大学の大規模ハイパフォーマンスコンピューティング(HPC)環境である TSUBAME に移植することから始まった。Altitude システムは先述した計算モデルを内蔵しており、Windows と UNIX に対応したグリッドコンピューティング機能を備えている。したがって最初の作業は OS 環境の変更に伴う修正作業であった。

平成 19 年度の課題では、弊社の製品として出荷しているほぼそのままのコードを使用し、1000 回~1 万回のモンテカルロシミュレーションを実施した。シミュレーション回数 1000 回の計算で、計算ノード数を 128 コア(8 ノード)から 1536 コア(96 ノード)まで(実際には管理用ノードとしてさらにもう 1 ノード使用)変化させて得たクラスタ性能を Fig.6 に示す。

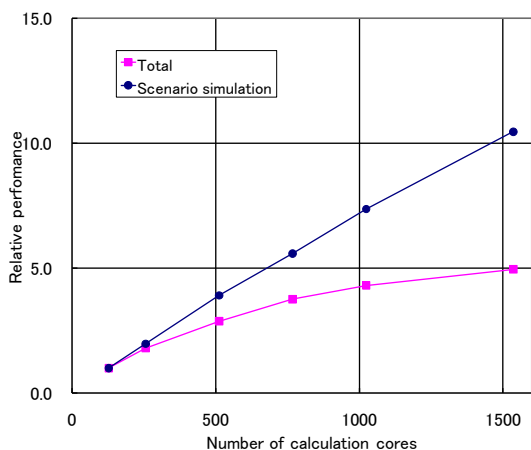


Fig.6 クラスタ性能曲線(信用シナリオ生成部改良前)

我々はあえて TSUBAME 環境向けの特別なコーディング等はしていない。弊社の顧客でも既にいくつかの小規模なグリッド環境が稼働しているが、そうした商用環境は必ずしも TSUBAME のような高性能なネットワーク環境を備えてはいないからである。HPC ユーザーとは必ずしも言えない金融機関を相手にする商用アプリケーションである以上、リッチな HPC 環境以外でも十分な性能と信頼性で稼働できるように設計する必要がある。

ホットスポットの発見

Altitude は Fig.7 の処理ステップで構成されている。このうちロード、データ配布、OLAP とストアに関してはシミュレーション回数に依らず処理時間に大きな差異はないため、信用シナリオ生成部とシナリオシミュレーション部が並列化処理の候補となる。本研究を始める以前からシナリオシミュレーション部分がボトルネックになることがわかっていたので当該個所の並列化対応は済ませてあった。Fig.6 の青線に示すようにこの部分は 1500 並列に至ってもなお綺麗にスケールしている。

初期の研究段階では Table 1 に示した入力データを入力した段階でデータ配布フェーズに問題が出た。すなわち通常ならば短時間で済む約 2Gbytes もしくは 2GB のデータ配布に要する時間が計算サーバーのノード数増大により無視できないものになったのである。この問題は Fig.8 に示すパイプライン方式を使って解決した。Fig.9 に改善効果を示す。この作業を行った後で KEK 真鍋 篤氏が同様の方法による Dolly+アプリケーション[4]を作成していることを知った。本研究では性能が Dolly+論文に示された性能を上回った。

ここまで作業した結果得た総合性能は Fig.6 の赤線の通りとなった。並列度を上げるに従い性能は上昇するが、アムダール則が働いて性能曲線が徐々に寝ていき 1536 並列で並列化効率が 50%となった。著名な商用リスク管理製品である米 NumeriX 社の製品や汎用統計ソフトベンダーである米 SAS 社の製品がわずかに 100 並列程度で並列化効率が 50%以下に落ちるのに比べれば[5]、Altitude は優秀と言えるだろう。しかしこの性能曲線を見れば大規模並列化対応が必要なことも明らかであった。

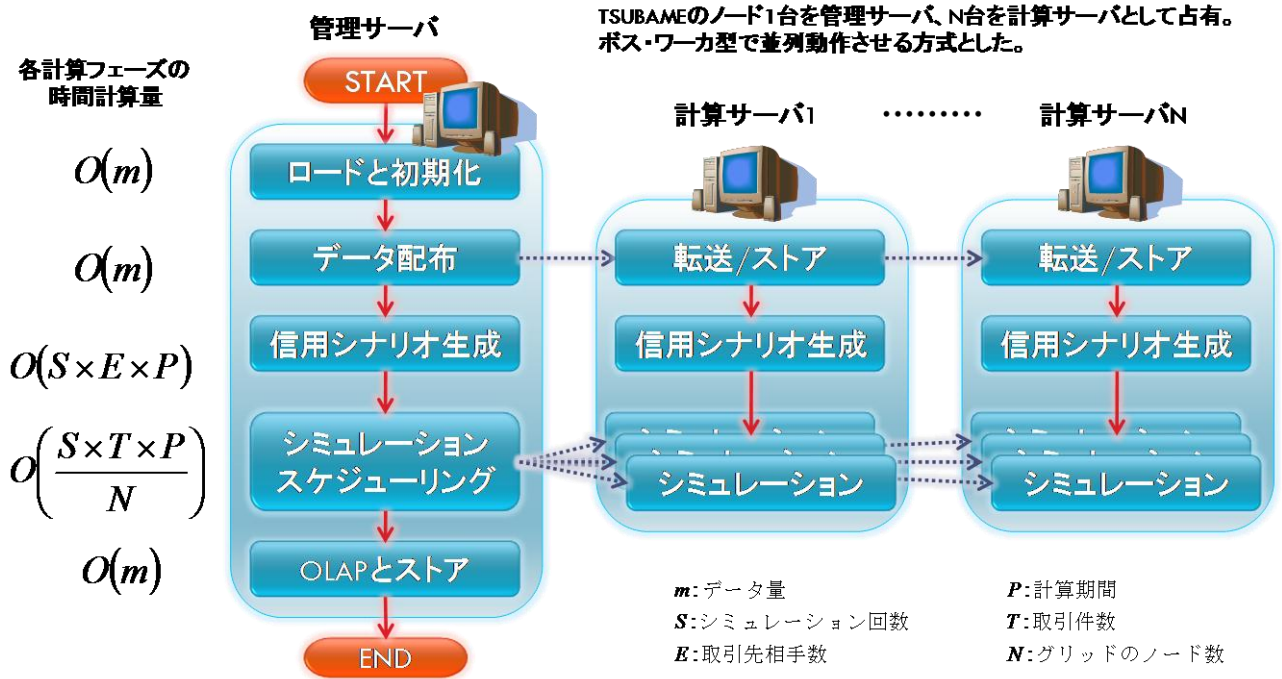


Fig.7 Altitude の処理ステップ

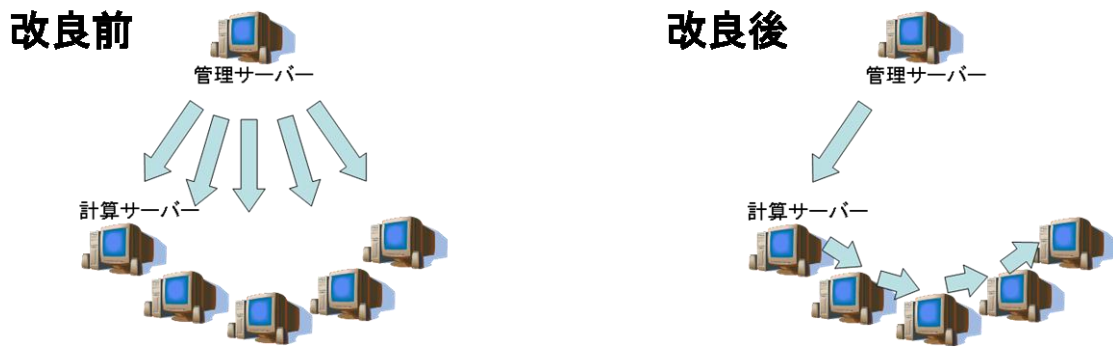


Fig.8 データ配布フェーズの改良

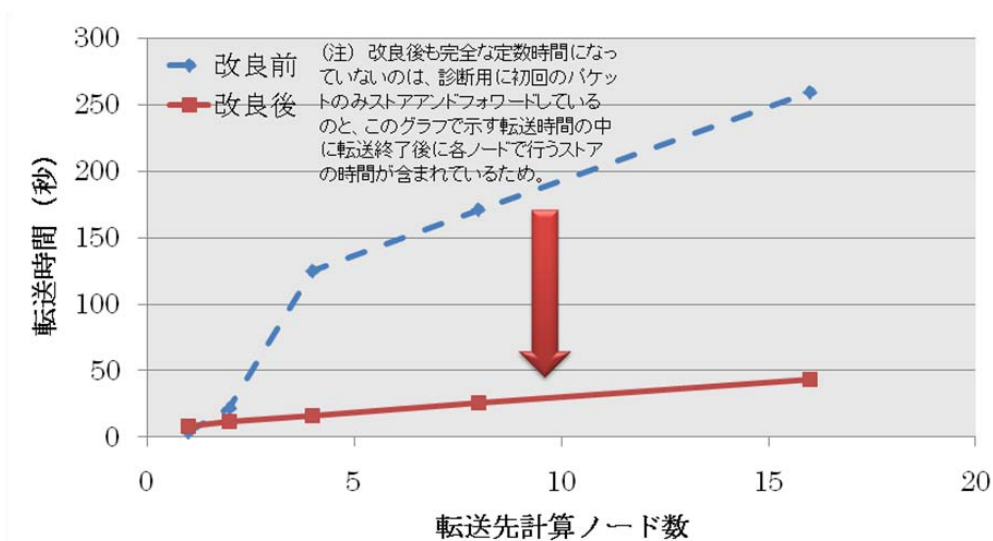


Fig.9 データ配布フェーズの改良による効果

信用シナリオ生成部の大規模並列化対応

我々はそれまで信用シナリオ生成部はシナリオシミュレーション部に比べれば相対的な計算量が少ないと考えてきた。ところが TSUBAME の利用によって並列度が上がったためにシナリオシミュレーション部は相対的に計算時間が短縮されてしまった。さらにデータ配布部もパイプライン方式で高速化された。こうして Fig.10 に示すように今度は信用シナリオ生成部が新たなホットスポットとして顕在化してきた。

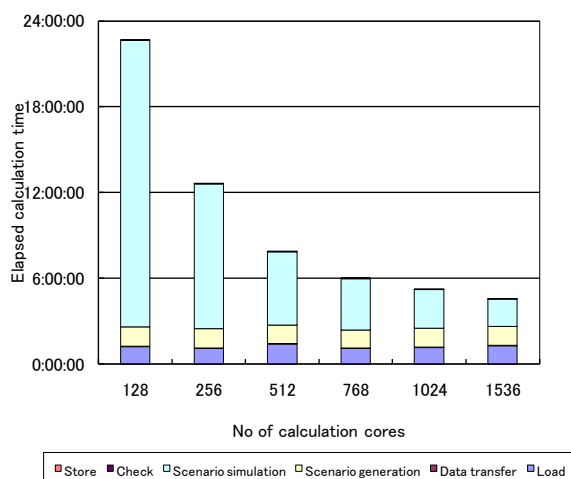


Fig.10 処理ステップ毎の計算時間
(信用シナリオ生成部改良前)

信用シナリオ生成部では、個別企業ごとに信用シナリオを生成しながらその格付けに応じた倒産や格付け変化の処理を行う。取引先間には一般的に関連があり、ある取引先の状況が別の取引先にも影響を与え得るため単純に取引先ごとにノード間で分担して並列処理することはできない。そのため今回は取引先間の関連をあらかじめ調べて処理ステップを分け、その処理ステップごとにノード間で同期を取り情報を分配することで並列化を行った。その結果 Fig.11 に示すように大幅に性能が改善した。

この改善手法では、各ノードで計算された信用シナリオ情報を全ノードに配る必要があり、これまで以上の大量データ転送が必要とされる。TSUBAME のようなリッチなネットワーク環境では全く問題にならないが実際の顧客環境では、Infiniband クラスの高速ネットワーク環境を導入すべきかについては費用対効果なども含めた検討が必要となるだろう。

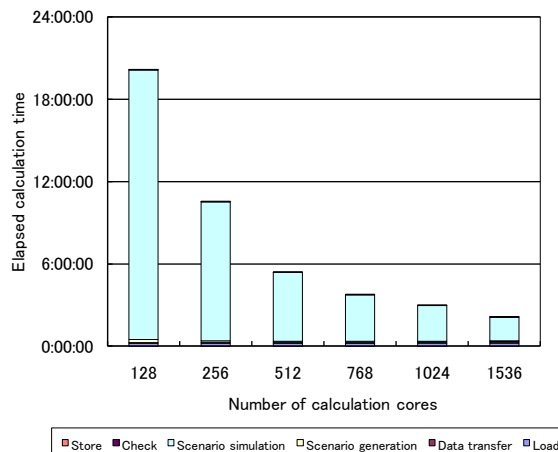


Fig.11 処理ステップ毎の計算時間
(信用シナリオ生成部改良後)

次の Fig.12 に信用シナリオ生成部を改善した後のクラスタ性能を示す。1500 並列規模でも優秀にスケールしていることがわかる。

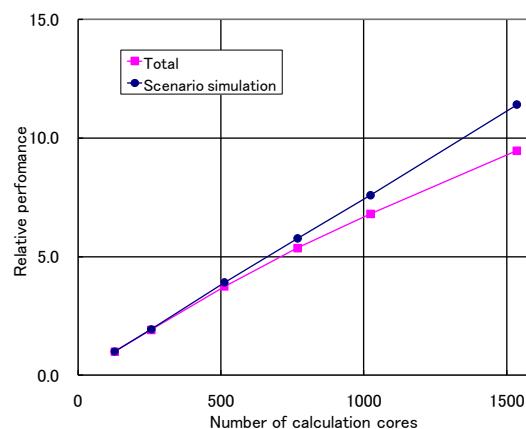


Fig.12 クラスタ性能曲線
(信用シナリオ生成部改良後)

全カテスト

最終的に我々は Table 1 の我が国最大のメガバンク規模を想定した全数シミュレーションを全カテストとして実施した。計算出力として VaR、EaR 等のリスク指標とともに TP 割り当て済み予想財務諸表をシナリオ毎に得た。10 万回規模の ALM シミュレーションを、135 ノード(2160 コア)を用いて 118.5 時間(約 5 日弱)で実行できることを確認し、メガバンク規模の ALM シミュレーションを現実的な規模の大規模並列環境において実務で要求される時間内に計算可能であることが示され

(様式第 20)

た。

HPC の威力の一端として、シミュレーション開始日付から 3 年間の経常収益の PDF(probability density function)を Fig.13(シミュレーション回数 1000 回)、Fig.14(シミュレーション回数 1 万回)、Fig.15(シミュレーション回数 10 万回)に示した。シミュレーション回数の増加にしたがって度数分布のグラフがなめらかになっている。実務で指標として使用されるパーセンタイル値は 99%、99.9%、99.97%といった周辺確率であるから、ある程度のシミュレーション回数の実施がなければ計算できない。実務においても近年では 10 万回を超えるようなモンテカルロシミュレーションも普通に行われるようになってきており、さらに多いシミュレーション回数の実施を求める声も上がっている。

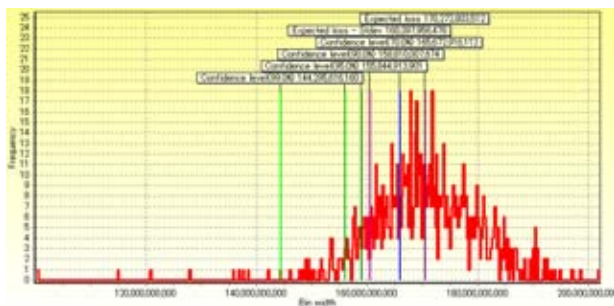


Fig.13 PDF (Number of simulations: 1000)

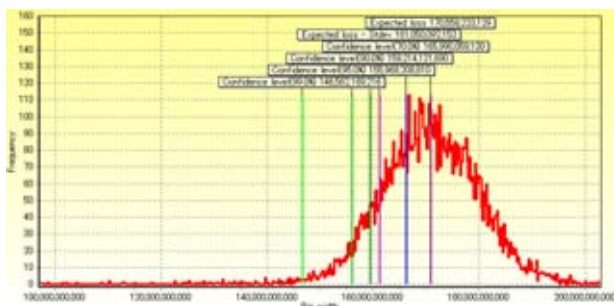


Fig.14 PDF (Number of simulations: 10000)

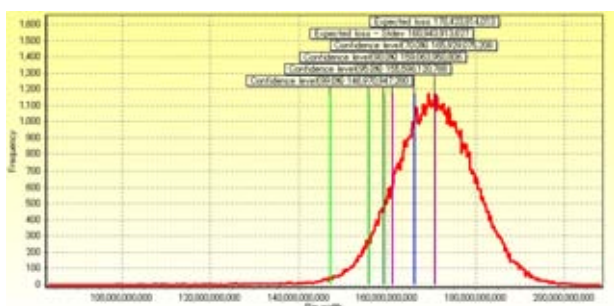


Fig.15 (Number of simulations: 100000)

結論

今回確認されたノード数の増加に対する性能向上特性とシミュレーション回数 10 万回での計算時間から 1 日で計算を終了させるために必要なコア数を単純に見積ると 11200 コアが必要となる。1 万コアの計算機クラスタとなると商用としてはかなり大規模ではあるが、近年の研究 HPC 環境としてはコモディティに接近している。また近年の CPU 性能自体の大幅な向上もあり(本研究で TSUBAME のノードで使用している CPU は今や 3 年以上も前の CPU である)、現時点ならばここで想定したコア数の半分以下で所望の性能を得ることができる。

つまり規制当局や金融機関が何と言おうが金融リスクの全体像を把握するシステムを構築できない理由として計算量の壁を言い訳には最早できないのである。

最新の HPC を使ってシステム規模を小さくできれば構築コストを削減できる。民間企業での導入も進むだろう。そうした意味で 2010 年中の稼働が予定される世界的にも先進的かつユニークな GPGPU クラスタ TSUBAME 2.0 への期待は大である。

謝辞

本研究は先端研究施設共用イノベーション創出事業【産業戦略利用】および先端研究施設共用促進事業による支援のもとに行われた。ここに記して感謝の意を表したい。

参考文献

- [1] Philippe Jorion (1996), "Value at Risk: The New Benchmark for Controlling Derivative Risk," Irwin Professional Pub
- [2] J.P. Morgan & Co. Incorporated (1997), "CreditMetrics™ - Technical Document."
- [3] Eduardo Canabarro (2010), "Counterparty Credit Risk," Risk Books
- [4] Atsushi Manabe (2001), "Disk cloning program 'Dolly+' for system management of PC Linux cluster," KEK, Tsukuba, Japan
- [5] 塩田雅之 (2010), "ポートフォリオ管理パッケージの並列化性能検証," Market Solutions Review Vol.3 No.2