Vol.2014-MPS-98 No.33
Vol.2014-BIO-38 No.33
2014/6/27

エクソーム解析パイプラインの京コンピュータ上での大規模並列化

青山 健人[1,2,a)]　角田 将典[1,b)]　松崎 由理[2,c)]　石田 貴士[1,d)]　秋山 泰[1,2,e)]

概要：近年，全ゲノム配列のうちタンパク質を翻訳するエクソン領域のみを解析するエクソーム解析が可能となり，がんゲノム研究などに用いられている．また，シーケンシング技術の向上によってゲノム情報の蓄積は増加し続けており，さらに大規模な生命情報解析環境が求められている．本研究では汎用 PC クラスター上で動作するエクソーム解析パイプラインソフトウェア Genomon-exome を理化学研究所のスーパーコンピュータ「京」上に移植し，パイプライン内部の処理について MPI による Master-Worker モデルでタスク分散を行うシステムを実装することで，ジョブ投入数を抑えた大規模な生命情報解析環境を構築した．本報告では，スーパーコンピュータ「京」上に実装したパイプラインの実行性能評価を行った．

キーワード：スーパーコンピュータ「京」，エクソーム解析，パイプライン, Genomon-exome，MPI

Large-scale Parallelization of Exome Analysis Pipeline on K-computer

Kento Aoyama[1,2,a)]　Masanori Kakuta[1,b)]　Yuri Matsuzaki[2,c)]　Takashi Ishida[1,d)]
Yutaka Akiyama[1,2,e)]

Abstract: Exome analysis, which is a method to analyze only protein coding region, has been used in various research fields such as a cancer genome research. Because of the improvement of a high-speed sequencer, demands of effective sequence analysis on large computational environment have been increased. Genomon-exome is a useful pipeline software for analyzing exome data but executable on only general PC clusters. In this study, We attempted to implement the Genomon-exome on the K-computer using a Master-Worker model task distribution framework implemented MPI. We also evaluated the scalability of the pipeline on K-computer.

Keywords: K-computer，Exome analysis, Pipeline, Genomon-exome, MPI

1. 導入

ヒトの塩基配列には血液型や皮膚の色の違いなど形質に影響を及ぼす遺伝子の情報が含まれている．中でも DNA から mRNA に転写されてタンパク質に翻訳されるコーディング領域を含む領域は全塩基配列の約 2%未満にも関わらず機能的に重要な役割を担うと考えられており，エクソンまたは総称してエクソームと呼ばれる．全塩基配列の中からエクソン領域だけを抽出し網羅的に解析することで機能的に重要なエクソン領域上の変異を効率的に検出する手法をエクソーム解析と呼び，希少な遺伝性疾患の原因遺伝子

[1] 東京工業大学 大学院情報理工学研究科 計算工学専攻
Department of Computer Science,
Graduate School of Information Science and Engineering,
Tokyo Institute of Technology
[2] 東京工業大学
博士課程教育リーディングプログラム情報生命博士教育院
Education Academy of Computational Life Sciences,
Tokyo Institute of Technology
[a)] aoyama@bi.cs.titech.ac.jp
[b)] kakuta@bi.cs.titech.ac.jp
[c)] matsuzaki@acls.titech.ac.jp
[d)] t.ishida@bi.cs.titech.ac.jp
[e)] akiyama@cs.titech.ac.jp

の同定などの研究に用いられている [1],[2],[3].

一方，次世代シーケンサの登場によって配列読み取りのスループットが飛躍的に向上し，日々膨大な量のデータがデータベースに蓄積され続けているが，その膨大なデータの解析コストが研究上のボトルネックとなっていることから大規模な生命情報解析環境が必要とされている．そこで解析コストの低減のため，解析で行われる一連の作業をパイプライン化し，高速な計算環境を利用したゲノム解析を提供するパイプラインソフトウェアが多く開発されている [4],[5]．そのようなパイプラインソフトウェアは提供するゲノム解析の機能に加え，想定する計算環境に応じて実装が多様化している．例えば，SIMPLEX[4] はエクソームシーケンスデータのマッピングやアライメント，アノテーション等の解析機能を備え，パイプライン全体を仮想イメージとして提供することで煩雑なインストール作業を簡略化し，Amazon EC2 などのクラウド環境に対応した解析が可能である．Genomon-exome[5] はジョブ管理システムを備えた汎用 PC クラスタ向けに実装されており，基本的な解析機能を備えているほか，ジョブ管理システムを利用して大規模な計算資源を動的に利用した解析が可能であり，実際にスーパーコンピュータを利用した解析結果が研究成果として発表されている [2],[3].

また，大規模な並列計算機の例として理化学研究所に設置されたスーパーコンピュータ「京」が挙げられる．「京」は 2011 年に計算性能の指標である LINPACK ベンチマークで 10.51PFLOPS を記録した実績を持つ，日本で最大規模の並列計算機である．現在，「京」を中心とした全国のスーパーコンピュータを活用して特定の研究分野に戦略的に取り組む HPCI 戦略プログラムで「予測する生命科学・医療および創薬基盤」が採択されており，その中で「大規模生命データ解析」が課題として取り組まれている．そのため，「京」上での生命情報解析に対する期待が高まっており，大規模な生命情報解析環境の構築が急務とされている．

しかし，「京」における生命情報解析には，消費メモリ量が増大しがちな生命情報解析に対して 1 ノードあたりのメモリ容量が少ないことや，解析パイプラインが内包する様々なソフトウェアに環境が対応していないなどの障害が多数存在している．特にジョブ管理システムを通した計算資源割り当ての待ち時間が大きいことは，ジョブ投入数が他用途のスーパーコンピュータ利用の数千倍になることもある生命情報解析で大きな問題となっている．

本研究ではエクソーム解析に対する大規模な生命情報解析を実現するため，エクソーム解析のワークフロー全体を有するパイプラインソフトウェア Genomon-exome を理化学研究所のスーパーコンピュータ「京」に移植し，ジョブ管理システムに依存していたパイプライン内部のタスク分散処理を MPI(Message Passing Interface) を用いて Master-Worker モデルで実装した．また，実装したパイプラインソフトウェアを用いてエクソームシーケンスデータの解析を行い，大規模並列環境における実行性能の評価を行った．

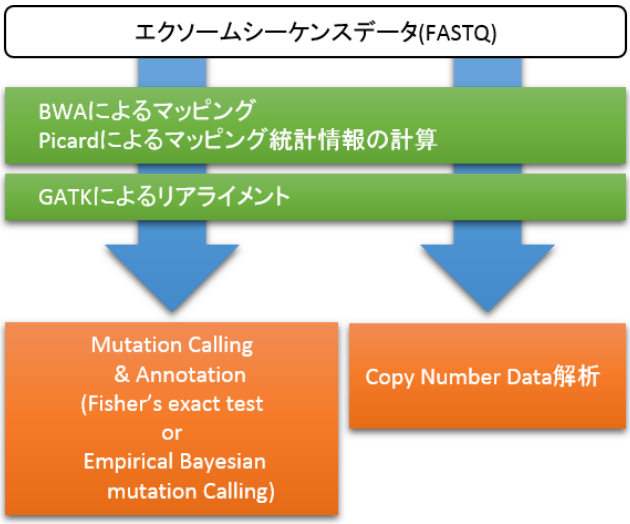

図 1 Genomon-exome エクソーム解析ワークフロー
Fig. 1 Genomon-exome, Exome analyzing work flow

2. Genomon-exome

Genomon-exome は東京大学医科学研究所ヒトゲノム解析センター宮野研究室および京都大学大学院医学研究科腫瘍生物学講座小川研究室の共同研究で 2012 年に開発されたエクソーム解析のパイプラインソフトウェアである．様々なオープンソースソフトウェアを組み合わせることで，エクソームシーケンスの結果である FASTQ ファイルのヒトゲノム (hg19) へのマッピングやデータ解析を行い，変異の候補一覧を出力することができる．Genomon-exome は既に実際に生物学の研究に利用されている．[2],[3].

Genomon-exome の解析ワークフローを図 1，使用ソフトウェアを図 2 に示す．まず，入力として与えられた FASTQ ファイルに対して BWA[6] を用いて断片配列の参照配列へのマッピングを行う．次に GATK[7] を用いてリアライメントを行い，アライメントミスを軽減する．そして得られた BAM ファイルに対して SAMtools[8] やスクリプトによるフィルタリングを行いながら塩基多型を検出し，ベイズ推定もしくはフィッシャー検定を適用後に Annovar[9] を用いてデータベースへのアノテーションを行う．また，Picard[10] を用いたマッピング結果の統計情報も出力する．

現在，Genomon-exome は東京大学医科学研究所ヒトゲノム解析センターのスーパーコンピュータのみに対応しており，ジョブ管理システムを利用してパイプラインの各処理を実行している．特に BWA によるマッピングや変異に対する検定など，並列化可能な部分を複数のジョブに分割して処理することで実行効率を向上させており，動的な計算資源の確保による効率的な解析を可能としている．

ソフトウェア名	利用内容	使用言語
BWA	FASTQデータのヒトゲノム(hg19)に対するマッピング	C, Perl
GATK	bamファイルのリアライメント	Java
SAMtools	sam(bam)ファイルに対する操作	C
Picard	マッピング率やカバレージなどの統計情報の出力 sam(bam)ファイルに対する操作	Java
bedtools	BEDファイル（エクソンのターゲット情報を示す）の操作	C++
cutadapt	PCRアーティファクト（アダプタシーケンス）の除去	Python
ANNOVAR	変異の候補の一覧にアノテーションを付与	Perl
maq	マッピングとアセンブリ	C++
bioconductor	Copy Number data解析	R

図 2 Genomon-exome の使用ソフトウェア
Fig. 2 Genomon-exome, using softwares

2.1 BWA によるマッピング処理のプロセスと並列度

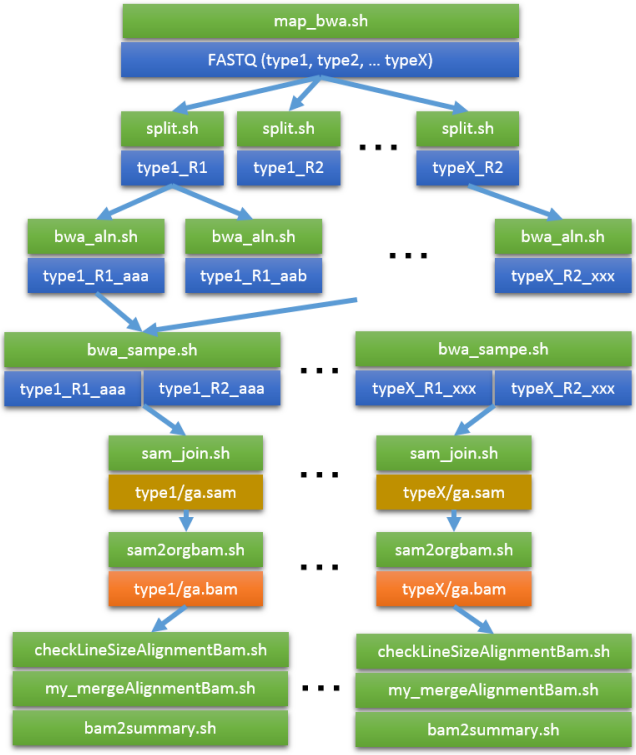

図 3 マッピング処理のプロセス
Fig. 3 Process of mapping

Genomon-exome のマッピング処理のプロセスの例を図 3 に示す．ユーザーが map_bwa.sh をジョブとして投入するとマッピング処理を開始する．まず，split.sh において入力の FASTQ ファイルをノード内で計算可能なサイズに分割する．次に，bwa_aln.sh で分割したファイルにそれぞれアライメントを実施し，bwa_sampe.sh でペアエンド配列から SAM ファイルを出力する．そのあと sam_join.sh で分割された sam ファイルを結合，sam2orgbam.sh で SAM ファイルから BAM ファイルへ変換・複製リードのマーク・インデックス作成を行い，checkLineSizeAlignmentBam.sh で BAM ファイルの出力サイズのチェック，my_mergeAlignmentBam.sh で Type 毎に BAM ファイルをマージし．最後に bam2summary.sh でマッピング統計情報を出力する流れとなる．

以上のマッピング処理を 1 サンプルに対して行った際の並列度（ジョブ投入数）を表 1 に示す．ペアエンド配列を仮定して，サンプルの解析対象の Type の数を T，FASTQ ファイル 1 つに対する分割数を S とした．

このように，Genomon-exome では並列度の高い処理を分割してジョブ管理システムに投入することで計算資源を効率的に利用した処理を行うが，「京」のようなジョブ投入から実行までの待ち時間が大きいシステムでは各ステップ毎の待ち時間が蓄積し，多大な待ち時間が発生することが予測される．

表 1 マッピング処理の並列度
Table 1 Concurrency on mapping process

ジョブ名	並列度
map_bwa.sh	1
split.sh	$2T$
bwa_aln.sh	$2TS$
bwa_sampe.sh	TS
sam_join.sh	T
sam2orgbam.sh	T
checkLineSizeAlignmentBam.sh	T
my_mergeAlignmentBam.sh	T
bam2summary.sh	T

3. 提案手法

本研究では，Genomon-exome をそのまま「京」上に移植した際に予測される，多数のジョブ投入による実行待ち時間の問題に対して，MPI を用いた Master-Worker モデルでタスク分散を行う MPIDP を利用することで解決を試みた．

3.1 MPI を用いた Master-Worker モデルによるタスク分散処理

MPIDP は 2012 年に東京工業大学大学院情報理工学研究科秋山研究室で開発された，MPI ライブラリを用いた Master-Worker モデルによるタスク並列分散処理フレームワークである．MPIDP のタスク分散の仕組みを図 4 に示す．MPIDP は，MPI におけるランク 0 のプロセスをタスク管理を担う Master に選び，残りのプロセスを実際にタスクを実行する Worker とする．Master は入力として与えられるタスクが記述されたリストを読み込み，Point-to-Point 通信により Worker にタスクを振り分ける．Worker は処理が終了するとそのことを Master に通知し，通知を受け取った Master は次の処理を Worker に与える．これを繰り返し実行することでタスクを並列分散処理することが可能である．

MPIDP は並列計算機環境で多数のクエリファイルに対

する高速な相同性検索を行う GHOST-MP[11] にも使用されており，「京」における実装の実績がある．本研究においては，Genomon-exome のパイプラインの各処理の並列分散に MPIDP を利用することで，実行待ち時間の原因であるジョブ投入数を抑えつつタスクの並列処理が可能なシステムの実装を目指した．

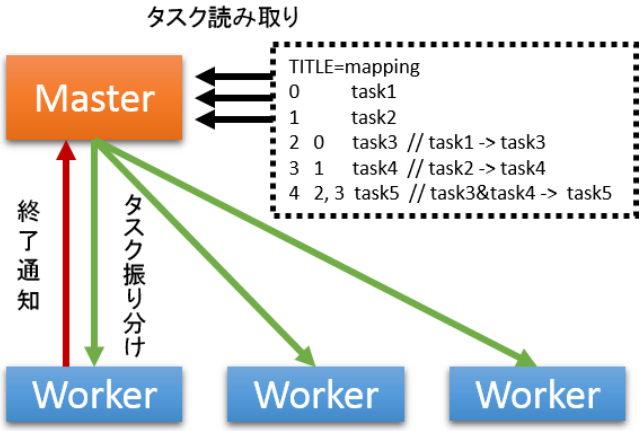

図 4 MPIDP によるタスク分散の仕組み
Fig. 4 MPIDP: Parallel task distribution

3.2 スーパーコンピュータ「京」上の Genomon-exome の実装

実装したシステムの処理フローをマッピング処理を例として図 5 に示す．ログインノードのデータ領域に解析対象のエクソームシーケンスデータ（FASTQ ファイル），必要なプログラム群（BWA 等）があるとする．まず，ログインノード上でスクリプトを実行してマッピング処理を行うために必要なタスクリストを出力する．「京」ではログインノードと計算ノードはディスク領域が異なるため，ジョブ投入時のデータ転送（ステージング）が記述されたスクリプトが同時に出力される．次に，出力されたスクリプトを用いてジョブ管理システムを通して計算ノードを確保し，必要なファイルを計算ノードからアクセス可能なワーク領域に転送する（ステージイン）．計算ノード上では MPIDP により，Master がタスクリストを元に Worker にパイプラインの各処理を割り当てる．全てのタスクの処理が終了すると，出力されたファイルをログインノード側のデータ領域に転送する（ステージアウト）．以上が「京」上に実装した MPIDP による Genomon-exome の処理の流れである．これにより，マッピングの一連の処理を 1 つの MPI プロセスとして実行することで，ジョブ投入を初めの 1 回に抑えることが可能である．

3.3 「京」のプログラム言語環境上の問題

Genomon-exome の提供する機能をそのまま「京」に実装

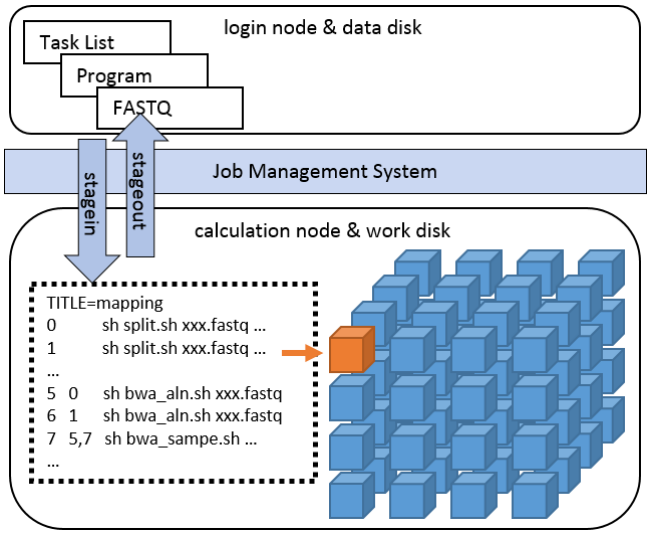

図 5 スーパーコンピュータ「京」における Genomon-exome のタスク分散処理の流れ
Fig. 5 Genomon-exome: Process work flow on K-computer

するにはプログラム言語環境上の問題がある．Genomon-exome に要求されるプログラム言語環境と現在の「京」の言語環境を表 2 に示す．

表 2 Genomon-exome の要求するプログラミング言語環境と現在のスーパーコンピュータ「京」の言語環境
Table 2 Requierd programming languages and K-computer environment

必要な言語環境	K-computer
C compiler	fcc ver. 1.2.0
Java	N/A
Python ver.2.6 以上	python ver. 2.6.2
Perl	perl ver. 5.10.0
R	R ver. 3.0.1[*1]

特に問題となるのが，Java の言語環境が「京」の計算ノードに存在しないことである．Java はリアライメントに用いる GATK やマッピング統計情報を出力する Picard に必要な他，SAM(BAM) ファイルに対する機能が様々なタスクで用いられている．「京」にはログインノードとネットワークで繋がれたプリポスト処理ノードと呼ばれる環境が存在しており，そこで Java を利用することは可能であるが，その場合には転送コストの問題やプリポスト処理ノードの低い計算性能を考慮する必要がある．

今回，我々は GATK によるリアライメント処理はマッピングと同程度の計算時間がかかるものの解析結果の精度への寄与が小さく，リアライメントを行わない利用者が多いなどの理由により「京」における実装を見送った．Picard によるマッピング統計情報出力については，その後の解析工程に対する依存関係がないことから，計算ノードによる

[*1] R は研究開始当初は言語環境は存在しなかったが，2013 年末に HPCI 運用事務局ヘルプデスクにより環境が提供された．

マッピング終了後にプリポスト処理ノードで計算することを検討している．また，様々なタスクで利用される Picard の SAM(BAM) ファイルに対する機能は，C 言語で動作する SAMtools の機能で代用した．

4. パイプラインの評価実験

実装したパイプラインの評価のため，エクソームシーケンスデータに対して「京」上に実装した Genomon-exome で解析を行う．今回の報告では特にマッピング処理に焦点を当て，並列度に対する実行時間の評価を行う．

4.1 実験データ

実験では 2 つのエクソームシーケンスデータに対して解析を行った．1 つ目は，Genomon-exome 公式サイト内の小規模な動作確認用人工データである [12]（small sample）．このデータに対する Genomon-exome による解析結果は公式サイトからダウンロード可能であり，実装したパイプラインの解析結果が一致することは確認済みである．2 つ目は，実際の肺がんに関する研究で用いられた肺がんのデータセット [13] から任意に 1 つずつ健常体 (normal:ERX135969, ERR160121) と罹患者 (tumor: ERX142229, ERR166339) のランデータ（lung cancer sample）を選んだデータを用いる．以下の表 3 に使用したデータの概要を示す．

表 3 実験に使用するエクソームシーケンスデータ
Table 3 Exome dataset used in evaluation

	small sample	lung cancer sample
Type	normal/tumor	normal/tumor
Length	100[bp]	100[bp]
Paired	yes	yes
Platform	Simulated	Illumina HiSeq 2000
TotalDataSize	3,548[MB]	74,100[MB]

4.2 実験方法・実験環境

実験では前述のデータに対して本研究で実装した Genomon-exome を用いて「京」上でエクソーム解析のマッピング処理を行い，ノード数を変化させながら実行時間の計測を行った．時間の計測にはジョブ管理システムによる実行情報に加えて，各タスクの実行時間を UNIX の “date” コマンドを使用して取得した時間情報を用いる．また，解析時の「京」の実験環境について表 4 に示す．

4.3 実験結果

本節では，small sample，lung cancer sample のそれぞれのサンプル解析時のマッピング処理の実行時間について述べる．

表 4 実験環境: スーパーコンピュータ「京」
Table 4 Environment: Riken, K-computer

# nodes	82944
CPU	SPARC64 VIIIfx [2.0GHz](8cores)
Memory	16[GB]
OS	Linux version 2.6.25.8
C compiler	fcc ver. 1.2.0
Python	python ver. 2.6.2
Java	N/A
Perl	perl ver. 5.10
MPI	OpenMPI ver. 1.4.3 FUJITSU MPI Library ver. 1.2.0
JMS	Parallel Job Manager

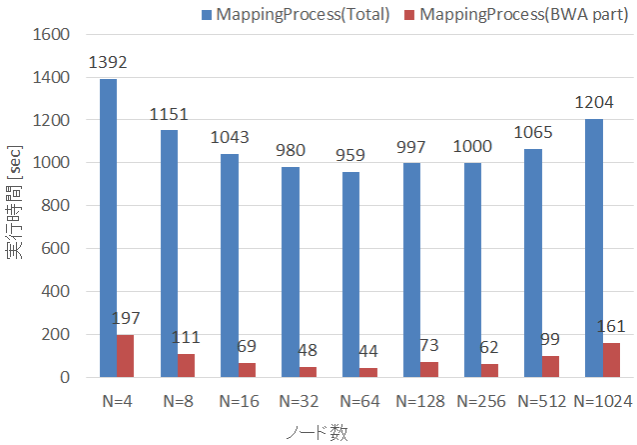

図 6 マッピング処理の実行時間（small sample）
Fig. 6 Execution time on mapping process(small sample)

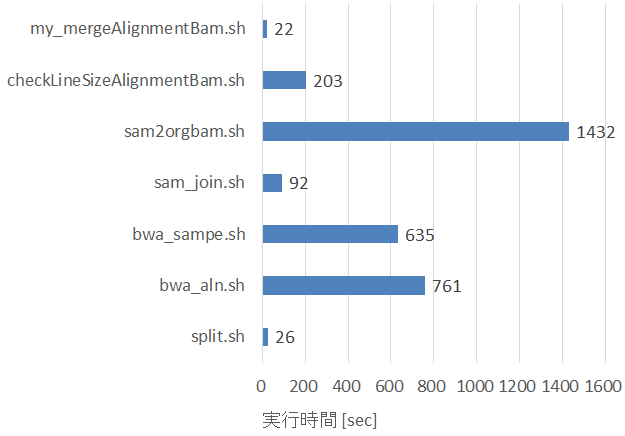

図 7 マッピング処理のタスク別合計実行時間（small sample）
Fig. 7 Summation time on mapping process(small sample)

4.3.1 small sample の解析時の実行結果

図 6 は small sample の解析におけるマッピング処理の合計実行時間と，その中の BWA を利用したマッピング部分のタスク（bwa_aln.sh，bwa_sampe.sh）の実行時間である．双方とも $N = 64$ のときに最も実行時間が短縮されており，ノード数 $N = 4$ の場合に対して BWA によるマッピ

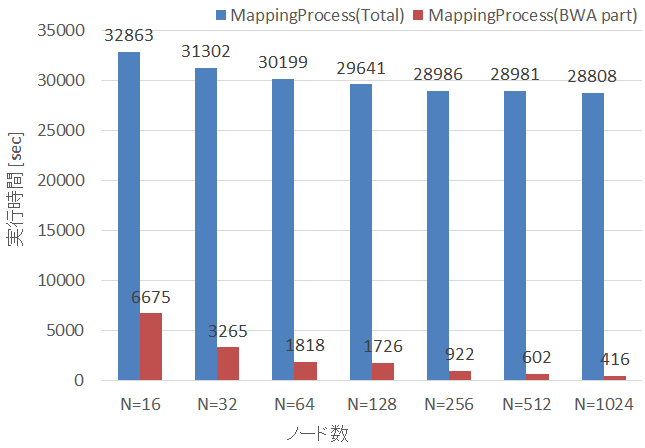

図 8 マッピング処理の実行時間（lung cancer sample）
Fig. 8 Execution time on mapping process(lung cancer sample)

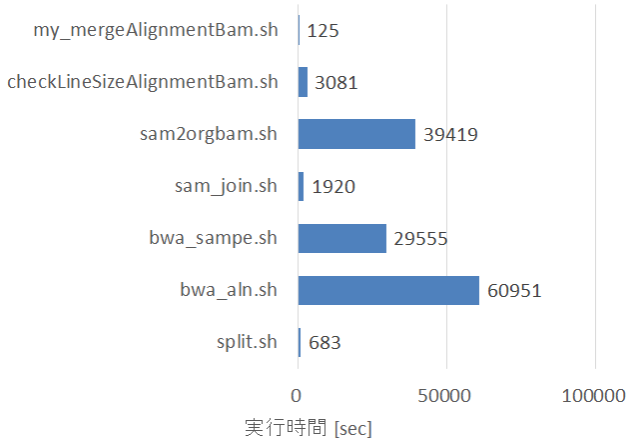

図 9 マッピング処理のタスク別合計実行時間（lung cancer sample）
Fig. 9 Summation time on mapping process(lung cancer sample)

ング部分は約 4.48 倍，合計実行時間は約 1.42 倍の高速化を達成した．しかし，これ以上のノード数を用いても実行時間の短縮は見られなかった．

一方，図 7 は small sample を $N = 4$ で解析した際に全ノードで並列実行される全タスクの実行時間を合算したタスク別合計実行時間である．並列実行される処理の実行時間も合算されるため，対象サンプル解析にかかるタスク量を見積もることが可能である．図 7 より，マッピング処理の全タスクの合計実行時間のうち約 45%が並列度の低い sam2orgbam.sh の実行時間で占められていたことがわかる．並列度が低く，ノード数増加による高速化が見込めないタスクの占める割合が大きいため，並列化による高速化が鈍化したものと考えられる．また，その他の要因として，小規模なサンプルをノード数に合わせて分割した結果，各ノード内の実行時間のバラつきやタスク数の増加によるオーバーヘッドが蓄積したことにより実行時間が増加したと考えられる．

4.3.2 lung cancer sample の解析時の実行結果

図 8 は lung cancer sample の解析におけるマッピング処理の合計実行時間と，その中の BWA を利用したマッピング部分のタスク（bwa_aln.sh，bwa_sampe.sh）の実行時間である．ノード数 $N = 16$ に対して $N = 1024$ で BWA によるマッピング部分は約 16.0 倍，合計実行時間は約 1.14 倍の高速化を達成している．

また，図 9 は lung cancer sample を $N = 16$ で解析した際のタスク別合計実行時間である．依然として合計実行時間における sam2orgbam.sh の占める割合が約 29%と大きいものの，並列度の高いタスクである bwa_aln.sh と bwa_sampe.sh が全体の 66%と大部分を占めていることから，small sample の解析時よりも良好な結果になったと考えられる．

最後に，図 10 に $N = 16$ をベースとしたノード数変化に

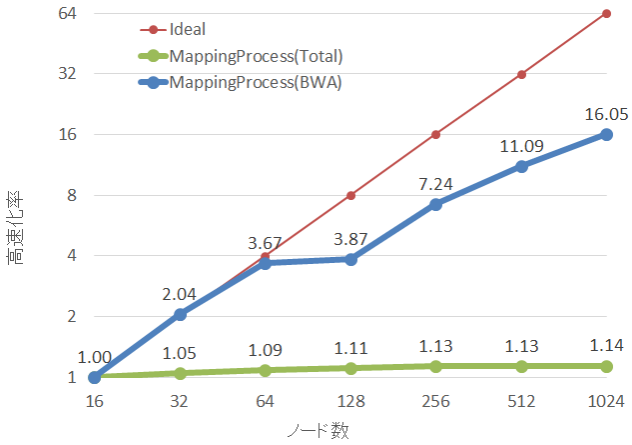

図 10 ノード数変化に対する高速化のスケーラビリティ（lung cancer sample）
Fig. 10 Speedup scalability on mapping process (lung cancer sample)

対する実行時間の高速化のスケーラビリティを示す．BWA によるマッピング部分は $N = 64$ まで順調にスケールしているものの，$N = 128$ 以上のノード数では鈍化している．これはノード数の増加による I/O 負荷の増加などの要因の他，実行時間の長さにより実験データの計測回数が少ないことによる影響と考えられる．一方で合計実行時間はノード数変化に対して高速化率の変化が少なく，ほぼノード数に対してスケールしない結果となった．

4.4 今後の課題

「京」上に実装した Genomon-exome によるマッピング処理の実行時間をノード数を増加させることによって短縮できたが，並列度の低いタスクの合計実行時間に占める割合も大きいことから良好な結果を得るには至らなかった．

今後の課題として，まず，並列度の低いタスクの終了までノードを確保し続けることは計算資源の節約という観点

からも改善が必要である．具体的な対処としては，並列度の低いタスクの実行を「京」の計算ノードではなく，ログインノードとネットワークで繋がるプリポスト処理ノードで行うことや，マルチサンプルの解析を前提として並列度の低いタスクの実行時間を隠蔽してスループットを向上させることが考えられる．

また，GATK によるリアライメントや Picard によるマッピング統計情報の出力など，現在対応していない機能の実装を検討する必要がある．プリポスト処理ノードは Java 環境を備えているため機能的に利点があるものの，計算性能の問題があるため大規模なサンプルの処理には不向きであり，実行性能等を含めて調査が必要である．

5. 結論

本研究では，エクソームシーケンスデータの解析パイプラインソフトウェアである Genomon-exome について，理化学研究所のスーパーコンピュータ「京」上への実装を行い，従来環境よりも多数のノードを用いたエクソーム解析が可能であることを示した．Genomon-exome の特性であるジョブ数が膨大になることを考慮し，スーパーコンピュータ「京」上の実装においてはパイプライン処理中に投入されるジョブ数を抑えるために MPI を用いて Master-Worker モデルによるタスク分散処理を行い，1 つの MPI プロセスとして実行することでジョブ数を大きく低減することに成功した．

本研究によって，日本最大規模の並列計算機であるスーパーコンピュータ「京」を利用した，より大きな規模のエクソームシーケンスデータの解析が可能となった．この成果はゲノム情報の解析コストがボトルネックとなっている現状に対して大規模な生命情報解析環境を提供し，ゲノム情報に基づいた個別化医療などの実現に貢献するものであるが，未だ実用上の課題は多く残っており，今後のシステム改善が必要である．

謝辞 本研究で使用した Genomon-exome を御提供頂き，また研究に関する様々な助言を賜った，東京大学医科学研究所 宮野 悟 教授，井元 清哉 准教授，白石 友一 助教，玉田 嘉紀 助教，伊東 聰氏 博士，千葉 健一 博士に厚く御礼申し上げる．本研究の結果は，理化学研究所のスーパーコンピュータ「京」を利用して得られたものである（課題番号:hp130017，hp140230）．また，文部科学省 博士課程教育リーディングプログラム東京工業大学「情報生命博士教育院」の支援を受けて行われた．

参考文献

[1] Bamshad, M., and Ng, S. “Exome sequencing as a tool for Mendelian disease gene discovery”, Nature Reviews, Vol.12, No.11, pp.745-755 (2011).
[2] Sakaguchi, H., Okuno, Y., Muramatsu, H. et al. “Exome sequencing identifies secondary mutations of SETBP1 and JAK3 in juvenile myelomonocytic leukemia, Nature Genetics”, Vol.45, No.8, pp.937-941 (2013).
[3] Shiraishi, Y., Sato, Y., Chiba, K. et al. “An empirical Bayesian framework for somatic mutation detection from cancer genome sequencing data”, Nucleic Acids Research, Vol.41, No.7, P.e89 (2013).
[4] Fischer, M., Snajder, R., Pabinger, S. “SIMPLEX: cloud-enabled pipeline for the comprehensive analysis of exome sequencing data”, PloS One, Vol.7, No.8, P.e41948 (2012).
[5] Genomon-exome: http://genomon.hgc.jp/exome (2014.05.19)
[6] Li, H., and Durbin, R. “Fast and accurate short read alignment with Burrows-Wheeler transform”, Bioinformatics, Vol.25, No.14, p.p1754-1760 (2009).
[7] McKenna, A., Hanna, M., Banks, E. et al. “The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data”. Genome Res, pp.1297-1303 (2010).
[8] Li, H., Handsaker, B., Wysoker, A.et al. “The Sequence Alignment/Map format and SAMtools”, Bioinformatics, Vol.25, No.16, pp.2078-2079 (2009).
[9] Wang, K., Li, M., Hakonarson, H. “ANNOVAR: functional annotation of genetic variants from high-throughput sequencing data”, Nucleic Acids Research, Vol.38, No.16, P.e164 (2010).
[10] Picard: http://picard.sourceforge.net/ (2014.05.19)
[11] GHOST-MP: http://www.bi.cs.titech.ac.jp/ghostmp/ (2014.05.19)
[12] http://genomon.hgc.jp/exome/faq.html
[13] Lung Cancer Sequencing Project Exome sequencing of lung adenocarcinomas and their normal counterparts: http://trace.ddbj.nig.ac.jp/DRASearch/study?acc=ERP001575 (2014.05.24)