平成 30 年度 文部科学省 ポスト「京」で重点的に取り組むべき社会的・科学的課題に関する アプリケーション開発・研究開発

# 平成 30 年度

# 「個別化・予防医療を支援する統合計算生命科学」

# 成果報告書

令和元年 5 月 31 日 国立大学法人東京大学 宮野 悟

本報告書は、文部科学省の科学技術試験研究委託 事業による委託業務として、国立大学法人東京大学 が実施した平成30年度「個別化・予防医療を支援す る統合計算生命科学」の成果を取りまとめたもので す。

1. 委託業務の題目1
2. 実施機関(代表機関)1
3. 委託業務の目的1
4. 平成 30 年度(報告年度)の実施内容2
4-1. 実施計画2
4-2. 実施内容(成果)
1. 大量シーケンスによるがんの個性と時間的・空間的多様性・起源の解明(サブ課題 A)5
2.データ同化生体シミュレーションによる個別化医療支援(サブ課題 B)
3. 心臓シミュレーションと分子シミュレーションの融合による基礎医学と臨床医学の架橋(サブ課題
C)
4. プロジェクトの総合的推進143
4-3. 活動(研究会・受賞・書籍等)149
4-4. 実施体制
別添1 学会等発表実績
別添2 実施計画

#### 1. 委託業務の題目

「個別化・予防医療を支援する統合計算生命科学」

#### 2. 実施機関(代表機関)

	機関名			国立大学法人東京大学					
	所在地		〒113-8654 東京都文京区本郷七丁目3番1号						
		ふりがな		みやの	さとる	生年	西暦 1954 年 12 月 5 日(64 歳)		
	⇒田 日早	氏	名	宮野	悟	月日	₩2019 4	年4月1日現在	
代	味 想	所属部署	罾名	医科学研究所			役職	教授	
表	貝仕伯	連絡分	七	Tel. 03-5449-5615 Fax. 03-5449-5442			449-5442		
機				E-mail: icls-office@hgc.jp, miyano@ims.u-tokyo.ac.jp					
関	事務連絡	ふりがな		まえだ ゆきこ					
		氏	名	前	田 幸子				
		事務 正尾如異々	毘夕	医科学研究所	〒 研究支援	課	小啦 ナバ		
		川周即有石		外部資金戦略	<b>小部資金戦略チーム</b>		上上		
	1534			Tel. 03-5449-5138 Fax. 03-6409-2017			09-2017		
		連絡プ	'L	E-mail: t-gs	shikin@ims	.u-tokyo	o.ac.jp		

3. 委託業務の目的

病気は、臓器群の変調という現象として現れるが、その背景には生命の設計図とも呼ばれるゲノムがあ り、オミクスと呼ばれるエピゲノム、RNA、タンパク質など多彩な分子が細胞を制御・構成している。 また、細胞には環境や加齢により長い時間をかけて変化していく個々人で異なる細胞コンテクストがあ り、そのもとで構成されている臓器の状態も多様である。さらに、その繋がりは人智を超えた複雑さを有 している。その理解には、画像や生理データなどを含む高精度臨床データとともに環境・生体・時空間的 にゲノムから全身を捉える必要がある。現在、ゲノム解析技術の劇的な革新と高精度計測機器の急速な発 展は、健康・医療ビッグデータを生み出そうとしており、この傾向は急激に加速している。こうした背景 のもと、本研究の目的は、ポスト「京」によって初めて実現できる「情報の技術」と「物理の原理」の融 合により、がんをはじめとして全脳・循環・代謝系など、全身の疾患に対して、ビッグデータを活用し、 高度の生体階層統合シミュレーションに個体データを同化させる技術、並びに、ライフサイエンスにおい てかつてない大規模なデータ解析技術を開発・応用することにより、また、大規模データに基づくアプロ ーチと並行して、分子細胞レベルの研究と臓器個体レベルの研究を融合させ、ミクロとマクロのメカニク スとを関連させて定量的に捉えたシミュレーションモデルを構築することにより、病態の理解と効果的 な治療法の探索を行い、その成果を個別化・予防医療へ返す基盤となる統合計算生命科学を確立すること を目的とする。

このため、国立大学法人東京大学を中核機関として、分担機関である国立大学法人京都大学、国立大学 法人大阪大学、株式会社 UT-Heart 研究所、学校法人自治医科大学と密接に連携し、研究開発を実施す る。 4. 平成 30 年度(報告年度)の実施内容

4-1. 実施計画

平成 30 年度は、本格実施フェーズの3年目であり、平成26~27年度に実施した調査研究・準備研究 および平成28~29年度の本格実施フェーズ開発結果に基づき、以下に示す重点課題2に関するサブ課題 (A~C)(1.~3.)について以下の目標を定め、研究開発を実施する。また、プロジェクトの統合的推進 4.を行う。

1. 大量シーケンスによるがんの個性と時間的・空間的多様性・起源の解明(サブ課題A)

がんはゲノムの変異によって生ずる疾患で、個体別、空間的(腫瘍の場所)、時間的多様性 を持っている。がんの起源とその多様性の理解は、がん治療戦略、がんの予防法と超早期発見 にイノベーションを起こし、副作用に優しく個人ごとに効き目のよい薬を創出するための戦 略上大変重要である。そのため、次の目標を設定し研究を実施する。

(1) 世界の追随を許さない大規模がんオミクスデータ解析システムの開発

大規模がんオミクスデータ解析システムの継続発展(高速化・高感度化)を図り、Genomonに基づ くポスト「京」の詳細設計コデザインを実施しつつ、高感度解析において、700 検体/日のパフォー マンスを達成することを目標にして、各機能を確認してシステムを発展させることを実施内容とす る。ここで高感度解析とは、通常の SNV 検出や融合遺伝子検出に加え複雑な変異解析(中間的範囲 の欠失挿入やウイルス挿入検出などの構造変異検出等)を含む。

- (2) がんの起源と集団内多様性を理解するためのシステムの開発 単一細胞や単一細胞由来のコロニーから遺伝子変異および網羅的な遺伝子発現を解析する系を確立 および最適化する。遺伝子変異がどのような細胞にどのような順序・組合せで発生し、なぜそのよう な組合せが特段に選択されていくのかを解析し、腫瘍の起源と多様性構築機構のモデルを検証する。
- (3) 胚細胞バリアントの効率的な解析システムの開発

胚細胞変異同定手法として、発生初期に胚細胞に生じた変異を DNA シークエンスデータから高精度 に検出・同定する数理的手法の開発を進める。細胞分化の結果、組織ごとに変異アレル頻度は異なる が、頻度が低い場合でも同定できるよう、ハプロタイプ情報およびシークエンスエラー発生過程を反 映した情報などの複数情報を統合した統計的生成モデルの構成法を検討し開発を進める。

2. データ同化生体シミュレーションによる個別化医療支援(サブ課題B)

ポスト「京」を活用した大規模生体物理シミュレーションと生体計測データとを様々なレ ベルで同化・融合させることにより、実測データを重視する医療に受け入れられる計算機シ ミュレータを開発し、シミュレーションで得られる物理情報を活用した個別化医療支援の実 現を目指す。そのため、平成30年度は次の目標を設定し研究を実施する。

(1) 全脳循環代謝シミュレータの開発と高度個別化医療支援への応用

医用画像から取得できる主要脳動脈と脳静脈を接続する脳血管モデルを構築し、大規模ボクセル型 血流シミュレータを用いて、全脳レベルの血流動態を再現する。また、脳微小循環シミュレータによ り、脳毛細血管網内の赤血球の流動、分配、ガス交換の特性を調べ、全脳毛細血管に均一に血液が供 給されるメカニズムを明らかにしていく。さらに、得られたシミュレーション結果を臨床で計測され る MRI や CT 画像で表現し、脳血流データと同化させる手法の開発を進める。これにより、脳循環 系個別化医療支援に向けた基盤技術の確立を目指す。

(2) 脳機能障害に関わる個別化医療支援のための大規模生体シミュレータの開発

脳機能障害が引き起こす循環障害、運動障害、発話障害の診断および治療を支援するための大規模生 体シミュレータの開発を進める。具体的には、ヒト運動機能解析のための神経筋骨格系統合シミュレ ータ、構音機能分析のための流体音響連成シミュレータ、微小循環遊走細胞の動態解析プラットフォ ームの開発を進める。また、心原性脳塞栓の発生リスク評価のための左心房内血流シミュレータおよ び脳動脈瘤コイル塞栓シミュレータを臨床データの後ろ向き評価に適用し、個別化医療支援に対す る有効性を検討する。

3. 心臓シミュレーションと分子シミュレーションの融合による基礎医学と臨床医学の架橋(サブ課題 C)

タンパク機能(遺伝子)の変化が最終的に臓器レベルにどのような変化を及ぼすかを物理 的に解明・予測することができれば、その応用は無限に広がる。ポスト「京」のパワーを活か し、本サブ課題では、世界で初めて心臓シミュレーションと分子シミュレーションを融合さ せた真のマルチスケールシミュレーションを実現する。これにより計算科学の歴史に新たな マイルストーンを築くと共に、現実の医療・創薬にブレークスルーをもたらす。

(1) 分子モデルとリモデリング・慢性期予測、連成手法最終化

心臓病、特に「心不全」の根本的解明と最適治療を可能とするマルチスケール心臓シミュレータの開発を目的とする。モデル化は心不全研究において得られた遺伝子レベルの変化、発現調節、細胞内情報ネットワークに関する膨大な情報、臨床情報を効果的に取り入れて推進する。
 平成 30 年度は、サルコメアモデルとリモデリングの高度化および検証を行う。また連成手法を最終化する。

(2) チャネル分子と薬剤のドッキングシミュレーション・検証

目標 1 は分子レベルの力学と心臓のポンプ機能の関係に焦点を当てたものであるが、同様の技術を 致死性不整脈の病態解明と予防・治療に対して適用する。即ち、重点課題1において進められるイオ ンチャンネルの分子シミュレーションと UT Heart を融合することで、候補物質の分子構造と遺伝 子情報のみから不整脈のリスク評価を可能とするシステムの完成を目標とする。

平成 30 年度は、イオンチャネルと薬剤のドッキングシミュレーションを実施し、細胞実験により検証する。

4. プロジェクトの総合的推進

プロジェクト全体の連携を密としつつ円滑に運営していくため、運営委員会や研究会の開 催等、参画各機関の連携・調整にあたる。

特に、プロジェクト全体の進捗状況を確認しつつ計画の合理化を検討し、必要に応じて調 査あるいは外部有識者を招聘して意見を聞くなど、プロジェクトの推進に資する。またニュ ースレター(電子版)発行、ホームページ運営、シンポジウム開催、人材育成活動などを通じ て研究の進捗と連携を推進する。

プロジェクトで得られた成果については、積極的に公表し、今後の展開に資する。

4-2. 実施内容(成果)

課題全体としての進捗は順調であり、一部、予定よりも進んでいるものがある。

## 1. 大量シーケンスによるがんの個性と時間的・空間的多様性・起源の解明(サブ課題A) 【概要】

ターゲットアプリケーション Genomon を「京」/ポスト「京」上でスムーズに実行するためのグリッ ドエンジン Virtual Grid Engine (VGE)を開発する中で、リスタート機能を開発し様々のテストを経て実 装した。そして大規模データ解析を実施して性能データを得た。成果は BIBM (2018)で発表した。 Genomon を用いた大規模シークエンスデータ解析では、139 人のがん患者ないし健常人から得られた 682 個の食道上皮検体についてゲノム解析を行った。その結果、NOTCH1 を主要な標的とするドライバ 一変異を獲得したクローンが幼少期から生じ、加齢とともに拡大すること、また、この過程は、高度の飲 酒や喫煙によって促進することが明らかとなった。70 歳という高齢者になると正常食道の大半がドライ バー変異を有するクローンに置換されているという驚くべき事実が判明した。これにより、がんの発生と 進化について世界初の知見がえられ、その成果を Nature (2019)に発表した。また、がんの進化を再現す るシミュレーションモデルを構築し、そのモデルのダイナミクスを、大規模並列にパラメータを変えなが ら精査する感受性解析を行った。その成果は昨年度までのものを含め Nature Communications (2018)及 び PLoS One (2019)に発表した。高精度にバリアント検出するために、これまで検討を行ってきたベイ ズモデル統合法を発展させ、進化系統樹の性質なども利用し、一患者由来の複数のシークエンスデータが 得られる場合に、バリアント検出を高精度化する方法を開発した。その成果は Bioinformatics 2019 に発 表した。

【詳細】

(1) 世界の追随を許さない大規模がんオミクスデータ解析システムの開発

A) Virtual Grid Engine の開発

(a) 概要

重点課題2のターゲットアプリケーションであるGenomonの「京」/ポスト「京」上での動作のために、Virtual Grid Engine (VGE)の開発を実施した。VGE は非グリッドエンジン環境において、MPI のプロセス空間を疑似的なグリッドエンジン環境としてユーザアプリケーションに提供するミドルウェアである。

ゲノムシーケンスや RNA シーケンスデータの解析ではパイプラインが一般的であり、本体および 内部利用の外部ソフトウェア等の更新が頻繁(~10回/年)なバイオインフォマティクス分野において、 ソフトウェアの個別 MPI 化は発見に至るまでの時間的競争という生命科学的優位性の観点から現実 的な対応ではない。グリッドエンジンはポスト「京」へ搭載しないことが決定しているが、「京」を含 めた HPCI 計算機群や他のペタフロップス級スーパーコンピュータ上での柔軟な運用のためには、必 要不可欠なソフトウェアである。平成 29 年度に実施したロバスト性向上および高速化に基づき、本年 度はリスタート機能の実装と大規模解析を実施した。次節以降で本年度実施内容を順次述べる。

(b) リスタート機能の開発

(ア) 設計と実装

スーパーコンピュータ「京」を始めとする大規模スーパーコンピュータでは、ユーザのジョブ実行 時間に制限をつけていることが通常である。「京」の場合、24 時間(small および large キュー)また は 8 時間(huge キュー)の経過でジョブの強制停止措置が実行される。重点課題 2 のターゲットアプ リケーション Genomon を含む、バイオインフォマティクス分野で一般的なパイプライン型ソフトウ ェアにおいて、任意の時刻における計算中断はソフトウェアの使用上想定されておらず、その状況か らの計算再開は容易には出来ない。何故なら、パイプライン中のソフトウェアには計算結果を逐次フ ァイル出力するものが多数含まれており、並列同時実行されたプロセスが途中終了したかどうかはシ ステムからの情報が無ければ判別できないからである。

グリッドエンジン環境下では、通常パイプライン中のサブプロセス(タスク)はメインジョブから 新たに投入されるジョブとして実行されるため、そのジョブが終了したかどうかを確認することによ ってタスクの完了/未完了を判断することが出来る。VGEにおいても各ジョブの情報を管理しており、 この管理情報を用いて計算を再開することは理論的には可能である。しかし、VGEの想定する多数検 体の同時解析においては、検体ごとに進行状況の異なる計算結果を逐一確認し、適切な再開箇所を設 定してパイプラインを個別に調整する必要が出てくるため、ユーザへの中断再開コストが甚大になり、 現実的な選択肢とは言えない。そこで我々は、任意状況における計算中断に対して自動的に復旧ポイ ントを設定し計算を再開可能にするリスタート機能を実装した。

リスタート機能の設計にあたり、最初に検討した項目は全自動リスタートが可能かどうかである。 前述のように、対象となるパイプラインの処理内容や利用ソフトウェアを考慮せずにあらゆる場合を 想定し、そのすべてに対応したリスタートは実装不可能であるとの結論に至った。これは、単一のソ フトウェアだけで考えても、中断から何もせずに再開可能な場合と一時ファイル等の除去作業などを 含む計算再開準備が必要なソフトウェアが想定され、VGEのシステムログだけではどちらを実行すべ きか判断できないことからも容易に理解できる。よって、本リスタート機能の設計においては、Fault Tolerant/Fault Resilience 分野で一般的なチェックポイントを用いたリスタート機能の実装について 述べる。

開発するリスタート機能の「京」コンピュータにおける各タスクのパイプラインの処理フローとチ ェックポイントデータの出力地点(青色の図形)を図1.1.1に示す。チェックポイントデータには、タ スク終了時におけるアウトプットディレクトリ以下のファイル情報を書き出す。パイプラインタスク は、次のタスクがアウトプットディレクトリ内のファイルを入力ファイルとして利用するようになっ ているため、タスク終了時のアウトプットディレクトリ以下のファイルが正しく揃っていればタスク は正しく実行されるはずである。したがって、アウトプットディレクトリのファイルが正しく揃って いれば、パイプラインの原理的にはどのタスクからでもスタートできる。

チェックポイントデータに書き出す構造情報は以下のとおりである。

- ① アウトプットディレクトリの絶対パス
- ② 各アウトプットファイルの絶対パス
- ③ 各アウトプットファイルの(アウトプットディレクトリから見た)相対パス
- ④ 各アウトプットファイルのサイズ

#### GenomonPipelineパイプラインフロー



図 1.1.1 各タスクのパイプラインのフロー

開発するリスタート機能の「京」コンピュータにおける通常実行およびリスタート実行の処理フロ ーを図 1.1.2 に示す。



図 1.1.2 リスタート機能を追加した Genomon の処理フロー

通常実行時は、「京」コンピュータ用の実行バッチファイルをジョブ投入することによって、入力デ ータをステージインし、入力データから Genomon の各タスクを実行する。リスタートに備えて各タ スク終了時にチェックポイントデータを作成する。チェックポイントデータにはアウトプットディレ クトリのファイル情報を書き込む。タスクが全て終了すると、出力データが生成される。また、「京」 コンピュータの時間制限により途中終了しリスタートする場合に備え、各アウトプットファイルとチ ェックポイントデータをアウトプットディレクトリごとステージアウトする。

リスタート時は、ユーザが通常実行時の実行バッチファイルを修正することにより、リスタート用 の実行バッチファイルを作成する。また、通常実行時のアウトプットファイルとチェックポイントデ ータを用意する。オリジナルの入力データは、全てのタスクを起動させるため再度ステージインする 必要がある。ステージイン後、チェックポイントマスタファイル(チェックポイントデータの作成情 報を管理するファイル)を用いて、タスクが前回の実行でどこまで進んでいたかをチェックし、どの タスクからリスタートするべきかどうかを判定する。チェックポイントデータや入力ファイルをチェ ックし、不足、または破損がある場合には、リスタートが不可能であると判断し、パイプラインを終 了する。これらのチェックに問題がなければ、入力データを再構築する。

入力データの再構築には、通常実行時のアウトプットディレクトリをそのまま用いるが、Genomon のアウトプットディレクトリには、入力データに対するシンボリックリンクが含まれている。「京」コ ンピュータのステージングの特殊性により、計算ノードのディレクトリが実行のたびに変化する。何 もしなければシンボリックリンクの元のファイルへの参照ができず、リスタート時のシンボリックリ ンクへのアクセス時に異常終了する。このような「京」コンピュータのステージング処理の特殊性に 対応するため、入力データのシンボリックリンクを張り直す必要がある。

パイプラインは各タスクの入出力データが後のタスクで保存されないため、そのままリスタートす ると最初のタスクのデータがないために、最初のタスクからリスタートされる。これを回避してリス タートしたい位置から正常に実行させるためには、各タスクのパイプラインの適切な定義が必要とな る。このため、タスクの ruffus デコレータ設定の設計を変更する必要がある。ruffus デコレータの入 力設定では、入力を表す第一引数に指定するタスクが実行されない場合、異常終了になることが調査 で判明した。このため、ruffus デコレータの入力部を変更する。パイプラインで用いられる ruffus デ コレータは、タスク関数に直接作用するためデコレータのオン/オフを関数定義上から直接は変更でき ない。そのため、ruffus デコレータを変える前と同様のタスク処理を行う関数を別に呼び出す。タス クに対して別の ruffus デコレータの処理を実施する場合は、関数定義部分においてリスタート位置チ ェックのフラグに基づいて関数定義を切り分けたうえで ruffus デコレータ処理を切り分ける。以上の 処理により、Genomon のリスタートを実現する。

【用語】

デコレータ:関数を引数にとり、関数を戻り値に設定する関数。 ruffus:ワークフローマネージメントシステムの一つ。Pyhton でパイプライン型ソフトウェアの記述を 容易にするためのライブラリー。 開発したリスタート機能に対応するためのパイプラインの構築方法について、図 1.1.3 に示すダイ アモンド型のパイプラインを例として記述する。このパイプラインにはルートが 3 つ存在し、ルート A の Task1A、Task2A が処理された後、ルート A の Task3A とルート B の Task3B が処理される。 その後、ルート C の Task4C で合流するパイプライン構成になっている。

ここで、タスクとはパイプラインの関数のことであり、図の Task1A、Task2A 等を指す。ルートと はパイプラインの関数が所属するグループであり、分岐や合流するとルートが増える。なお、各ルー トの命名規則は以下の通りである。

- 1. 分岐、合流しない限りは、同一のルート名とする。
- 2. 分岐した場合は、1つのルートを分岐前のルート名と同一にし、それ以外のルートは重複 しない新規のルート名とする。
- 3. 合流した場合は、重複しない新規のルート名とする。
- 4. 一度終了したルートのルート名を再度使用することはできない。



図 1.1.3 ダイアモンド型のパイプライン

まず、関数 checkpoint\_restart と、その戻り値 restart\_files\_list\_dict の使用例を記述する。関数 checkpoint\_restart は、パイプライン処理に対する前処理の最初に呼ばれ、前回実行時に生成された チェックポイントデータがあれば、そこからパイプライン処理をリスタートするための情報を戻り値 として取得する。この時、戻り値 flag\_skiptask には実行しないタスクを設定したスキップフラグ、戻 り値 flag\_starttask には開始タスクを設定したリスタートフラグ、戻り値 restart\_files\_list\_dict に は、各タスクのリスタートに必要なアウトプットファイルの情報が格納されており、複数の処理ルートがある場合には、そのルートごとに用意したリストに情報を格納する。使用例は図 1.1.4 の通りで ある。



pipeline\_run([merge\_bam], multithread=5)

図 1.1.4 関数 checkpoint\_restart と戻り値 restart\_files\_list\_dict の使用例

続けて、パイプラインの各タスクの記述法と、それに伴う関数 write\_checkpoint の使用例、リスタートするための ruffus デコレータ設定について記述する。ruffus デコレータは、タスクの処理の種類 によって異なるものが用いられる。代表的な ruffus デコレータは以下の通りである。

• originate

このデコレータが付いたタスクは、パイプラインの最初のタスクになる。pipeline\_run を呼ん だ時に、最初に実行するタスクに指定する。

• follows

このデコレータが付いたタスクより前に実行しておくタスクの関数名を指定し、タスクの順序 依存性を持たせる。

subdivide

複数(1つのファイル群)から複数のファイルを出力するタスクに付ける。

• collate

複数の入力ファイルを、同一の出力をするグループに分けるタスクに付ける。

• transform

1つの入力ファイルから別の1つのファイルを出力するタスクに付ける。

• merge

複数の入力ファイルから1つのファイルを出力するタスクに付ける。

リスタートする際に正しくパイプラインを構築するための、タスク記述の要点は以下の5点である。 ただし、ここでは合流タスクについては考慮しない。

フラグによってそのタスクを実行するか判定する if 文を用意する。
 一つのタスクの実行に対して、考えられるケースは以下の3つが挙げられる:
 ケース1:リスタート時の最初のタスクとして実行する

ケース2:そのタスクを実行しないでスキップする

ケース3:リスタート時は、通常のパイプライン実行と同じように実行する これらのうちどのケースであるかによって、ruffus デコレータの記述、またはタスクを実行す るか否かが変わるため、if 文により処理を決定する。

- リスタート時のデコレータから@follows を除去する。
   例えば Task2A は、通常実行時は Task1A の後に実行するため@follows が設定されているが、 Task2A からリスタートする際には Task1A が実行されない。そのため、@follows を除去する 必要がある。
- リスタート時のデコレータの第一引数をルートごとに変更する。
   入力ファイルのリストをルートごとに用意し、ruffus デコレータ(transform、merge 等)の第 一引数に指定する。
- 既存のタスクの間に、チェックポイントデータ書き出しのための関数を追加する。
   チェックポイント機能の追加により、既存のタスクの間にチェックポイントを書き出す処理が 加えられる。リスタート時に用いるチェックポイントデータを書き出すため、各タスクの終了 後にチェックポイントデータ書き出しのための関数を追加する。
- 通常実行時の関数に対して、デコレータ@follows(checkpoint\_task○○)を追加する。
   チェックポイント書き出しタスクの終了を待つ必要があるため、例えば Task2A は、Task1A 終了後のチェックポイントデータ書き出しタスク(checkpoint\_task1A)の後に処理されるよう
   にデコレータを設定する。

例として、上記の規則に従って Task2A の記述をリスタート機能未対応のものからリスタート機能 対応のものに書き換える。書き換え前を図 1.1.5 に、書き換え後を図 1.1.6 に示す。なお、関数 write\_checkpoint の第1引数はチェックポイントを書き出すタスク名、第2引数は次のタスク名のリ スト、第3引数は次のタスクのルート名のリスト、第4引数は各アウトプットファイルの絶対パスの リストを表す。

<pre>@follows( run_pre )</pre>
@transform(run_pre , suffix(".fastq"),".bam1")
def run_bwa1(input_file, output_file1):
print "test input %s to output 1 -> %s " % (input_file, output_file1)
fw=open(output_file1,"w")
fi=open(input_file,"r")
for row in fi:
fw.write(row.strip())
fw.write('¥n')
print('test 1')
fw.write('run bwa1¥n')
print('test 1-2')
fw.close
fi close

#### 図 1.1.5 Task2A の構築方法の書き換え前



図 1.1.6 Task2A の構築方法の書き換え後

続けて、合流タスクである Task4C の記述を書き換える。書き換え前を図 1.1.7、書き換え後を図 1.1.8 に示す。合流には、merge デコレータを使用する。通常実行時以外は、merge デコレータの第 1 引数には、合流タスクのインプットとなるファイルの絶対パスのリスト restart\_files\_list\_C を指定す る。restart\_files\_list\_C には、関数 pipeline\_run を呼ぶ前に、Task4C への入力として与えるファイ ルの絶対パスのリストを予め格納する。また、表 1.1.1 に示す通り、合流タスク直前のタスクの状態 によって、Task4C に設定するデコレータが変わってくる。そのため、合流タスク直前のタスクの数 が増えると、書き換え後の if 文の分岐数も増える。

パターン	Task3A の状態	Task3B の状態	Task4C のデコレータ
1	終了	終了	@merge(restart_files_list_C, "merge.bam")
2	終了	未完了	<pre>@follows(checkpoint_task3b) @merge([restart_files_list_C], "merge.bam")</pre>
3	未完了	終了	<pre>@follows(checkpoint_task3a) @merge([restart_files_list_C], "merge.bam")</pre>
4	未完了	未完了	@follows(checkpoint_task3a) @follows(checkpoint_task3b) @merge([run_bwa2, run_bwa3], "merge.bam")

表 1.1.1 Task4C の if 文のパターン

@merge([run\_bwa2,run\_bwa3], "merge.bam")
def merge\_bam(inputfiles, output\_file4):
 fw=open(output\_file4,"w")
 for input\_file\_name in inputfiles:
 fi=open(input\_file\_name,"r")
 for row in fi:
 fw.write(row.strip())
 fw.write('¥n')
 print "merge input %s " % ( input\_file\_name)
 fi.close
 print "merge output %s " % ( output\_file4)
 fw.close

図 1.1.7 Task4C の構築方法の書き換え

flag_skiptask, flag_starttask, restart_files_list_dict = checkpoint_restart( restart_files_list_A = [] restart_files_list_B = [] restart_files_list_C = [] if restart_files_list_dict.get("Route_A") is not None: restart_files_list_dict.get("Route_B") is not None: restart_files_list_dict.get("Route_B") is not None: restart_files_list_dict.get("Route_C") is not None: restart_files_list_dict.get("Route_C") is not None: restart_files_list_dict.get("Route_C") is not None: restart_files_list_dict.get("Route_C") is not None: restart_files_list_C = restart_files_list_dict.get("Route_C") for outputfile_path in 合流タスクのインプットとなるアウトプットファ たリスト:	target_dir) ァイルの絶対パスを要素とし
if outputfile_path が存在する: continue	
restart_files_list_C に outputfile_path を追加する 絶	対パスを予め格納
~~~中略~~~	
if flag_starttask.get("Task4C") and not flag_skiptask.get("Task3B"): @follows(checkpoint_task3b) @merge(restart_files_list_C, "outputdir/test/merge.bam") def merge_bam(inputfiles, output_file4): ~~~中略~~~	パターン 2
elif flag_starttask.get("Task4C") and not flag_skiptask.get("Task3A"): @follows(checkpoint_task3a) @merge(restart_files_list_C, "outputdir/test/merge.bam") def merge_bam(inputfiles, output_file4):	パターン3
~~~中略~~~	パターン 1
elif flag_starttask.get("Task4C"): @merge(restart_files_list_C, "outputdir/test/merge.bam") def merge_bam(inputfiles, output_file4): ~~~中略~~~	
	パターン 4
@follows(checkpoint_task3a) @follows(checkpoint_task3b) @merge([run_bwa2,run_bwa3], "outputdir/test/merge.bam") def merge_bam(inputfiles, output_file4): ~~~中略~~~	

図 1.1.8 Task4C の構築方法の書き換え後

図 1.1.9 に示す赤丸で囲われたタスクであり、合流タスクの直前のタスクおよび、パイプラインの 分岐先の最後のタスクである。それぞれの場合の関数 write\_checkpointの使用例を図 1.1.10、図 1.1.11 に示す。また、ルートが1つしか存在しなかった場合は、最後のタスクが終点タスクとなる。なお、 終点タスクの関数 write\_checkpointの第2引数は["final"]、第3引数はそのタスクのルート名を要素 とするリストとする。

最後に、タスクが終点タスクだった時の関数 write\_checkpoint の使用例を記述する。終点タスクは



図 1.1.9 終点タスクの例



図 1.1.10 合流タスクの直前かつ合流タスクと異なるルートのタスクの関数 write\_checkpoint の使用例

#### 

図 1.1.11 分岐先の終点タスクの関数 write\_checkpoint の使用例

(イ) リスタート機能のテスト

以上の VGE 提供機能を Genomon が利用できるよう、Genomon 側にリスタート関数を挿入し、本 リスタート機能のテストを実施した。以降にテスト仕様、および結果を記述する。

リスタート機能は、チェックポイントデータに基づいて各タスクの実行に必要なデータを再構築し、 Genomon の中断したタスクから再度計算を実施する機能である。パイプラインタスク実行前のアウ トプットディレクトリの状態によって通常実行/リスタートを自動的に判断する。そのため、まず一検 体データに対する通常実行、リスタートのテストを実施する。また、Genomonは複数検体の解析を可能としている。このため、リスタート機能のテストは1つの検体データを用いたテストおよび複数の検体データを用いたテストを実施する。

チェックポイントデータに問題がある場合には Genomon を強制終了する。したがって、チェック ポイントデータに問題がある場合に強制終了するかどうかテストを実施する。

以上の考えに基づき、選定したテストパターンを表 1.1.2 に示す。テストパターンは全て、図 1.1.1 のフローで実施する。各テストパターンでの詳細な条件、確認項目、すなわちテストケースについて は、各テストの項で説明する。なお、No.1~4 については、整合性判定を行わない、ファイル名で行 う、ファイルサイズで行う、の 3 通りそれぞれについて確認を行う。テストは「京」上で実施した。

テストパターン	テストパターンの内容
No.1	ー検体データ入力時の Genomon 通常実行
No.2	ー検体データ入力時の Genomon リスタート実行
No.3	複数検体データ入力時の Genomon 通常実行
No.4	複数検体データ入力時の Genomon リスタート実行
No.5	チェックポイントデータに問題がある場合の Genomon リスタート実行

表 1.1.2 テストパターン一覧

#### テストパターンNo.1:一検体入力のGenomon 通常実行

テストパターン No.1 のステージング設定と入力設定を以下に示す。テストフローは図 1.1.1。

#PJM --stg-transfiles all #PJM --stgin ~./public.tar ./~ #PJM --stgin ~./database.tar ./~ #PJM --stgin ~./sample.tgz ./~ #PJM --stgin ~./genomon.cfg ./~ #PJM --stgin ~./dna\_task\_param.cfg ./~ #PJM --stgin ~./vge.cfg ./~ #PJM --stgin ~./vge\_output.tgz ./~ #PJM --stgout ~./vge\_output.tgz ./~ #PJM --stgout ~./output.tgz ./~

genomon\_pipeline dna ./sample/sample.csv ./output ./genomon.cfg ./dna\_task\_param.cfg > stdout1.log 2> stderr1.log

Genomon の通常実行時に異常や不具合がないかテストする。テストの確認項目を表 1.1.3 に示す。

確認対象	確認ファイル	確認内容
変異ルートの最終 出力ファイルデー タ	output1/mutation/sample_tumo r/sample_tumor_genomon_mut ations.result.txt	オリジナルの変異ルートの最終出力ファイルデ ータと比較し、同じか確認する。
SV ルートの最終 出力ファイルデー タ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルのSVルートの最終出力ファイルデー タと比較し、同じか確認する。
変異ルートのチェ ックポイントアウ トプットディレク ト リ の bam 、 fastq、mutation 下	output1/checkpoint/backup_Ro ute_A/bam、 output1/checkpoint/backup_Ro ute_A/fastq、 output1/checkpoint/backup_Ro ute_A/mutation	オリジナルのアウトプットディレクトリの bam、 fastq、mutation 下を比較し、ファイル名/ディレ クトリ名が同じか確認する。
SV ルートのチェ ックポイントアウ トプットディレク トリの sv	output1/checkpoint/backup_Ro ute_B/sv	オリジナルのアウトプットディレクトリの sv 下 を比較し、ファイル名/ディレクトリ名が同じか 確認する。
変異ルートのチェ ックポイントマス タファイル	output1/checkpoint/checkpoint _Route_A.mst	以下のデータが書き込まれているか確認する。 Task3A,checkpoint_Task1A.chkの絶対パス Task4A,checkpoint_Task3A.chkの絶対パス Task5A,checkpoint_Task4A.chkの絶対パス Task6A,checkpoint_Task5A.chkの絶対パス Task7A,checkpoint_Task6A.chkの絶対パス
SV ルートのチェ ックポイントマス タファイル	output1/checkpoint/checkpoint _Route_B.mst	以下のデータが書き込まれているか確認する。 Task2B,checkpoint_Task5A.chk の絶対パス Task3B,checkpoint_Task2B.chk の絶対パス Task4B,checkpoint_Task3B.chk の絶対パス
Task1A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task1A.chk であるか
ポイントデータ	_Task1A.chk	を確認する。
Task3A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task3A.chk であるか
ポイントデータ	_Task3A.chk	を確認する。
Task4A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task4A.chk であるか
ポイントデータ	_Task4A.chk	を確認する。
Task5A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task5A.chk であるか
ポイントデータ	_Task5A.chk	を確認する。
Task6A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task6A.chk であるか
ポイントデータ	_Task6A.chk	を確認する。
Task7A チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task7A.chk であるか
ポイントデータ	_Task7A.chk	を確認する。
Task2B チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task2B.chk であるか
ポイントデータ	_Task2B.chk	を確認する。
Task3B チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task3B.chk であるか
ポイントデータ	_Task3B.chk	を確認する。
Task4B チェック	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task4B.chk であるか
ポイントデータ	_Task4B.chk	を確認する。

表 1.1.3 テストケース No.1-1 のテスト確認項目

#### テストパターン No.2:一検体入力の Genomon リスタート実行

テストケースとそのときの各タスクの状態を表に示す。テストパターン No.2 のリスタートのテストは、表に示すテストケースの数だけ行う。テストフローは図 1.1.1。

テストケ ース	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
No.2-1	リスター ト	未完了	未完了	未完了	未完了	未完了	未完了	未完了
No.2-2	終了	終了	終了	リスター ト	未完了	終了	終了	終了
No.2-3	終了	終了	終了	終了	リスター ト	終了	終了	終了
No.2-4	終了	終了	終了	終了	終了	リスター ト	未完了	未完了
No.2-5	終了	終了	終了	終了	終了	終了	リスター ト	未完了
No.2-6	終了	終了	終了	リスター ト	未完了	リスター ト	未完了	未完了
No.2-7	終了	終了	終了	リスター ト	未完了	終了	リスター ト	未完了
No.2-8	終了	終了	終了	終了	リスター ト	リスター ト	未完了	未完了
No.2-9	終了	終了	終了	終了	リスタート	終了	リスタート	未完了

表 1.1.4 テストパターン No.2 におけるテストケース一覧

Task2A、Task1Bからのリスタートは、Task1A、Task2Bがいずれも入力の設定あるいはシンボリ ックリンクの設定であり、チェックポイントの書き出しはしないためテストしない。Task1Aからの リスタートはTask1AがGenomoのタスク開始位置であるためテストしない。Task4A、Task5Aか らのリスタートはTask3Aからのリスタートとリスタート条件(Task3Aはルートが分かれない共通 中間タスクであり、Task3Aの入力となるTask1Aはルートが分かれない共通中間タスク)が同一で あるためテストしない。Task4BからのリスタートはTask3Bからのリスタートとリスタート条件 (Task3Bはルートが分かれる分岐中間タスクであり、Task3Bの入力となるTask2Bはルートが分か れる分岐中間タスク)が同一であるためテストしない。テストパターン No.2のステージング設定と 入力設定を以下に示す。 #PJM --stg-transfiles all #PJM --stgin "../public.tar ./" #PJM --stgin "../database.tar ./" #PJM --stgin "../sample.tgz ./" #PJM --stgin "../genomon.cfg ./" #PJM --stgin "../dna\_task\_param.cfg ./" #PJM --stgin "./vge.cfg ./" #PJM --stgin "./output.tgz ./" #PJM --stgout "./vge\_output.tgz ./"

genomon\_pipeline dna ./sample/sample.csv ./output ./genomon.cfg ./dna\_task\_param.cfg > stdout1.log 2> stderr1.log

## テストケース No.2-1: Task3A からリスタート

Genomon の Task3A からのリスタート時に異常や不具合がないかテストする。テストの確認項目 を表 1.1.5 に示す。

確認対象	確認ファイル	確認内容
変異ルートの 最終出力ファ イルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイルデータ と比較し、同じか確認する。
SV ルートの 最終出力ファ イルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイルデータと 比較し、同じか確認する。
Task1A のス キップ分岐通 過	stdout1.log	Task1A のスキップ分岐を通過したかデータを確認する。
Task3A のリ スタート分岐 通過	stdout1.log	Task3A のリスタート分岐を通過したかデータを確認 する。
Task4A の通 常分岐通過	stdout1.log	Task4Aの通常分岐を通過したかデータを確認する。
Task5A の通 常分岐通過	stdout1.log	Task5A の通常分岐を通過したかデータを確認する。
Task6A の通 常分岐通過	stdout1.log	Task6A の通常分岐を通過したかデータを確認する。
Task7A の通 常分岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認する。
Task2B の通 常分岐通過	stdout1.log	Task2Bの通常分岐を通過したかデータを確認する。
Task3B の通 常分岐通過	stdout1.log	Task3Bの通常分岐を通過したかデータを確認する。
Task4B の通 常分岐通過	stdout1.log	Task4Bの通常分岐を通過したかデータを確認する。

表 1.1.5 テストケース No.2-1 のテスト確認項目

## テストケース No.2-2: Task6A からリスタート

Genomon の Task6A からのリスタート時に異常や不具合がないかテストする。テストの確認項目 を表 1.1.6 に示す。

確認対象	確認ファイル	確認内容
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイルデ ータと比較し、同じか確認する。
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータを 確認する。
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータを 確認する。
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータを 確認する。
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータを 確認する。
Task6A のリスタ ート分岐通過	stdout1.log	Task6A のリスタート分岐を通過したかデータ を確認する。
Task7A の通常分 岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認 する。
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のスキップ分岐を通過したかデータを 確認する。
Task3B のスキッ プ分岐通過	stdout1.log	Task3B のスキップ分岐を通過したかデータを 確認する。
Task4B のスキッ プ分岐通過	stdout1.log	Task4B のスキップ分岐を通過したかデータを 確認する。

表 1.1.6 No.2-2 のテスト確認項目

## テストケース No.2-3: Task7A からリスタート

Genomon の Task7A からのリスタート時に異常や不具合がないかテストする。テストの確認項目 を表 1.1.7 に示す。

確認対象	確認ファイル	確認内容
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。
Task6A のスキッ プ分岐通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。
Task7A のリスタ ート分岐通過	stdout1.log	Task7A のリスタート分岐を通過したかデー タを確認する。
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。
Task3B のスキッ プ分岐通過	stdout1.log	Task3B のスキップ分岐を通過したかデータ を確認する。
Task4B のスキッ プ分岐通過	stdout1.log	Task4B のスキップ分岐を通過したかデータ を確認する。

表 1.1.7 テストケース No.2-3 のテスト確認項目

## テストケース No.2-4: Task2B からリスタート

Genomon の Task2B からのリスタート時に異常や不具合がないかテストする。テストの確認項目を表 1.1.8 に示す。

確認対象	確認ファイル	確認内容
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumor /sample_tumor_genomon_muta tions.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sampl e_tumor.genomonSV.result.txt	オリジナルの SV ルートの最終出力ファイルデ ータと比較し、同じか確認する。
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータを 確認する。
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータを 確認する。
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータを 確認する。
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータを 確認する。
Task6A のスキッ プ分岐通過	stdout1.log	Task6A のスキップ分岐を通過したかデータを 確認する。
Task7A のリスタ ート分岐通過	stdout1.log	Task7A のスキップ分岐を通過したかデータを 確認する。
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のリスタート分岐を通過したかデータ を確認する。
Task3B のスキッ プ分岐通過	stdout1.log	Task3B の通常分岐を通過したかデータを確認 する。
Task4B のスキッ プ分岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。

表 1.1.8 テストケース No.2-4 のテスト確認項目

## テストケース No.2-5: Task3B からリスタート

Genomon の Task3B からのリスタート時に異常や不具合がないかテストする。テストの確認項目 を表 1.1.9 に示す。

確認対象	確認ファイル	確認内容		
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。		
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。		
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。		
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。		
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。		
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。		
Task6A のスキッ プ分岐通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。		
Task7A のスキッ プ分岐通過	stdout1.log	Task7A のスキップ分岐を通過したかデータ を確認する。		
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。		
Task3B のリスタ ート分岐通過	stdout1.log	Task3B のリスタート分岐を通過したかデー タを確認する。		
Task4B の通常分 岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。		

表 1.1.9 テストケース No.2-5 のテスト確認項目

## テストケース No.2-6: Task6A と Task2B からリスタート

Genomon の Task6A と Task2B からのリスタート時に異常や不具合がないかテストする。テストの確認項目を表 1.1.10 に示す。

確認対象	確認ファイル	確認内容		
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。		
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。		
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。		
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。		
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。		
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。		
Task6A のリスタ ート分岐通過	stdout1.log	Task6A のリスタート分岐を通過したかデー タを確認する。		
Task7A の通常分 岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認する。		
Task2B のリスタ ート分岐通過	stdout1.log	Task2B のリスタート分岐を通過したかデー タを確認する。		
Task3B の通常分 岐通過	stdout1.log	Task3B の通常分岐を通過したかデータを確 認する。		
Task4B の通常分 岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。		

表 1.1.10 テストケース No.2.6 のテスト確認項目

## テストケース No.2-7: Task6A と Task3B からリスタート

Genomon の Task6A と Task3B からのリスタート時に異常や不具合がないかテストする。テストの確認項目を表 1.1.11 に示す。

確認対象	確認ファイル	確認内容		
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。		
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。		
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。		
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。		
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。		
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。		
Task6A のリスタ ート分岐通過	stdout1.log	Task6A のリスタート分岐を通過したかデー タを確認する。		
Task7A の通常分 岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認する。		
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。		
Task3B のリスタ ート分岐通過	stdout1.log	Task3B のリスタート分岐を通過したかデー タを確認する。		
Task4B の通常分 岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。		

表 1.1.11 テストケース No.2-7 のテスト確認項目

## テストケース No.2-8: Task7A と Task2B からリスタート

Genomon の Task7A と Task2B からのリスタート時に異常や不具合がないかテストする。テストの確認項目を表 1.1.12 に示す。

確認対象	確認ファイル	確認内容		
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。		
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。		
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。		
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。		
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。		
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。		
Task6A のスキッ プ分岐通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。		
Task7A のリスタ ート分岐通過	stdout1.log	Task7A のリスタート分岐を通過したかデー タを確認する。		
Task2B のリスタ ート分岐通過	stdout1.log	Task2B のリスタート分岐を通過したかデー タを確認する。		
Task3B の通常分 岐通過	stdout1.log	Task3B の通常分岐を通過したかデータを確認する。		
Task4B の通常分 岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。		

表 1.1.12 テストケース No.2-8 のテスト確認項目

## テストケース No.2-9: Task7A と Task3B からリスタート

Genomon の Task7A と Task3B からのリスタート時に異常や不具合がないかテストする。テストの確認項目を表 1.1.13 に示す。

確認対象	確認ファイル	確認内容
変異ルートの最 終出力ファイル データ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
SV ルートの最終 出力ファイルデ ータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.tx t	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
Task1A のスキッ プ分岐通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。
Task3A のスキッ プ分岐通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。
Task4A のスキッ プ分岐通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。
Task5A のスキッ プ分岐通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。
Task6A のスキッ プ分岐通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。
Task7A のリスタ ート分岐通過	stdout1.log	Task7A のリスタート分岐を通過したかデー タを確認する。
Task2B のスキッ プ分岐通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。
Task3B のリスタ ート分岐通過	stdout1.log	Task3B のリスタート分岐を通過したかデー タを確認する。
Task4B の通常分 岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。

表 1.1.13 テストケース No.2-9 のテスト確認項目

## テストパターン No.3: 複数検体入力の Genomon 通常実行

テストパターン No.3 のステージング設定と入力設定を以下に示す。テストフローは図 1.1.1。

#PJMstg-transfiles all
#PJMstgin "/public.tar ./"
#PJMstgin "/database.tar ./"
#PJMstgin "/sample.tgz ./"
#PJMstgin "/genomon.cfg ./"
#PJMstgin "/dna_task_param.cfg ./"
#PJMstgin "./vge.cfg ./"
#PJMstgout "./vge_output.tgz ./"
#PJMstgout "./output.1tgz ./"
#PJMstgout "./output.2tgz ./"
genomon_pipeline
dna ./sample/sample.csv ./output1 ./genomon.cfg ./dna_task_param.cfg > stdout1.log 2>
stderr1.log
genomon_pipeline
dna ./sample/sample.csv ./output2 ./genomon.cfg ./dna_task_param.cfg > stdout1.log 2>
stderr1.log

## テストケース No.3-1:通常実行

Genomonの通常実行時に異常や不具合がないかテストする。テストの確認項目を表1.1.14に示す。

確認項目	確認対象	確認内容		
検体1の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイルデ ータと比較し、同じか確認する。		
検体1の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイルデ ータと比較し、同じか確認する。		
検体1の変異ル ートのチェック ポイントアウト プットディレク トリの bam、 fastq、mutation 下	output1/checkpoint/backup_Ro ute_A/bam、 output1/checkpoint/backup_Ro ute_A/fastq、 output1/checkpoint/backup_Ro ute_A/mutation	オリジナルのアウトプットディレクトリのbam、 fastq、mutation 下を比較し、ファイル名/ディレ クトリ名が同じか確認する。		
検体1の SV ル ートのチェック ポイントアウト プットディレク トリの sv	output1/checkpoint/backup_Ro ute_B/sv	オリジナルのアウトプットディレクトリの sv 下 を比較し、ファイル名/ディレクトリ名が同じか 確認する。		

表 1.1.14 テストク	ース No.3-1	のテスト	、確認項目
---------------	-----------	------	-------

検体 1 の変異ル ートのチェック ポイントマスタ ファイル	output1/checkpoint/checkpoint _Route_A.mst	以下のファイルデータが書かれているか確認す る。 Task3A,checkpoint_Task1A.chkの絶対パス Task4A,checkpoint_Task3A.chkの絶対パス Task5A,checkpoint_Task4A.chkの絶対パス Task6A,checkpoint_Task5A.chkの絶対パス Task7A,checkpoint_Task6A.chkの絶対パス				
検体1の SV ル ートのチェック ポイントマスタ ファイル	output1/checkpoint/checkpoint _Route_B.mst	以下のファイルデータが書かれているか確認す る Task2B,checkpoint_Task5A.chkの絶対パス Task3B,checkpoint_Task2B.chkの絶対パス Task4B,checkpoint_Task3B.chkの絶対パス				
検体1のTask1A チェックポイン トデータ	output1/checkpoint/checkpoint _Task1A.chk	ファイル名は checkpoint_Task1A.chk であるか を確認する				
検体1のTask3A チェックポイン トデータ	output1/checkpoint/checkpoint _Task3A.chk	ファイル名は checkpoint_Task3A.chk であるか を確認する				
検体1のTask4A チェックポイン トデータ	output1/checkpoint/checkpoint _Task4A.chk	ファイル名は checkpoint_Task4A.chk であるか を確認する				
検体1のTask5A チェックポイン トデータ	output1/checkpoint/checkpoint _Task5A.chk	ファイル名は checkpoint_Task5A.chk であるか を確認する				
検体1のTask6A チェックポイン トデータ	output1/checkpoint/checkpoint _Task6A.chk	ファイル名は checkpoint_Task6A.chk であるか を確認する				
検体1のTask7A チェックポイン トデータ	output1/checkpoint/checkpoint _Task7A.chk	ファイル名は checkpoint_Task7A.chk であるか を確認する				
検体1のTask2B チェックポイン トデータ	output1/checkpoint/checkpoint _Task2B.chk	ファイル名は checkpoint_Task2B.chk であるか を確認する				
検体1のTask3B チェックポイン トデータ	output1/checkpoint/checkpoint _Task3B.chk	ファイル名は checkpoint_Task3B.chk であるか を確認する				
検体1のTask4B チェックポイン トデータ	output1/checkpoint/checkpoint _Task4B.chk	ファイル名は checkpoint_Task4B.chk であるか を確認する				
検体2の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイルデ ータと比較し、同じか確認する。				
検体2の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイルデ ータと比較し、同じか確認する。				

検体2の変異ル ートのチェック ポイントアウト プットディレク トリの bam、 fastq、mutation 下	output1/checkpoint/backup_Ro ute_A/bam, fastq, mutation	オリジナルのアウトプットディレクトリのbam、 fastq、mutation 下を比較し、ファイル名/ディレ クトリ名が同じか確認する。		
検体2の SV ル ートのチェック ポイントアウト プットディレク トリの sv	output1/checkpoint/backup_Ro ute_B/sv	オリジナルのアウトプットディレクトリの sv 下 を比較し、ファイル名/ディレクトリ名が同じか 確認する。		
検体2の変異ル ートのチェック ポイントマスタ ファイル	output1/checkpoint/checkpoint _Route_A.mst	以下のデータが書かれているか確認する。 Task3A,/checkpoint_Task1A.chk Task4A,/checkpoint_Task3A.chk Task5A,/checkpoint_Task4A.chk Task6A,/checkpoint_Task5A.chk Task7A /checkpoint_Task6A chk		
検体2の SV ル ートのチェック ポイントマスタ ファイル	output1/checkpoint/checkpoint _Route_B.mst	以下のデータが書かれているか確認する。 Task2B,/checkpoint_Task5A.chk Task3B,/checkpoint_Task2B.chk Task4B,/checkpoint_Task3B.chk		
検体2のTask1A チェックポイン トデータ	output1/checkpoint/checkpoint _Task1A.chk	ファイル名は checkpoint_Task1A.chk であるか を確認する		
検体2のTask3A チェックポイン トデータ	output1/checkpoint/checkpoint _Task3A.chk	 ファイル名は checkpoint_Task3A.chk であるた を確認する		
検体2のTask4A チェックポイン トデータ	output1/checkpoint/checkpoint _Task4A.chk	ファイル名は checkpoint_Task4A.chk であるか を確認する		
検体2のTask5A チェックポイン トデータ	output1/checkpoint/checkpoint _Task5A.chk	ファイル名は checkpoint_Task5A.chk であるか を確認する		
検体2のTask6A チェックポイン トデータ	output1/checkpoint/checkpoint _Task6A.chk	ファイル名は checkpoint_Task6A.chk であるか を確認する		
検体2のTask7A チェックポイン トデータ	output1/checkpoint/checkpoint _Task7A.chk	ファイル名は checkpoint_Task7A.chk であるか を確認する		
検体2のTask2B チェックポイン トデータ	output1/checkpoint/checkpoint _Task2B.chk	ーーーー ファイル名は checkpoint_Task2B.chk であるか を確認する		
検体2のTask3B チェックポイン トデータ	output1/checkpoint/checkpoint _Task3B.chk	ファイル名は checkpoint_Task3B.chk であるか を確認する		

検体2の		
Task4B チェッ	output1/checkpoint/checkpoint	ファイル名は checkpoint_Task4B.chk であるか
クポイントデー	_Task4B.chk	を確認する
タ		

#### テストパターン No.4:複数検体入力の Genomon リスタート実行

テストケースとそのときの各タスクの状態を表 1.1.15 に示す。テストパターン No.4 のリスタート のテストは、表 1.1.15 に示すテストケースの数だけ行う。なお、テストパターン No.4 では二つの検 体の同時リスタートをテストするが、各テストケースで二検体のタスク進行状況は同一であるものと する。テストフローは図 1.1.1。

テストケ	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
ース								
No.4-1	リスター ト	未完了	未完了	未完了	未完了	未完了	未完了	未完了
No.4-2	終了	終了	終了	リスター ト	未完了	終了	終了	終了
No.4-3	終了	終了	終了	終了	リスター ト	終了	終了	終了
No.4-4	終了	終了	終了	終了	終了	リスター ト	未完了	未完了
No.4-5	終了	終了	終了	終了	終了	終了	リスター ト	未完了

表 1.1.15 テストパターン No.4 におけるテストケース一覧

Task2A、Task1Bからのリスタートは、Task1A、Task2Bがいずれも入力の設定あるいはシンボリ ックリンクの設定であり、チェックポイントを書き出ししないためテストしない。Task1Aからのリ スタートはTask1AがGenomoのタスク開始位置であるためテストしない。Task4A、Task5Aから のリスタートはTask3Aからのリスタートとリスタート条件(Task3Aはルートが分かれない共通中 間タスクであり、Task3Aの入力となるTask1Aはルートが分かれない共通中間タスク)が同一であ るためテストしない。Task4BからのリスタートはTask3Bからのリスタートと条件(Task3Bはルー トが分かれる分岐中間タスクであり、Task3Bの入力となるTask2Bはルートが分かれる分岐中間タ スク)が同一であるためテストしない。また、複数タスクからのリスタートは、テストパターン No.2 でテストしているためテストしない。

テストパターン No.4 のステージング設定と入力設定を以下に示す。

#PJMstg-transfiles all
#PJMstgin ~/public.tar ./"
#PJMstgin "/database.tar ./"
#PJMstgin "/sample.tgz ./"
#PJMstgin "/genomon.cfg ./"
#PJMstgin "/dna_task_param.cfg ./"
#PJMstgin "./vge.cfg ./"
#PJMstgin <sup>‴</sup> ./output.1tgz ./ <sup>″</sup>
#PJMstgin <sup>%</sup> ./output.2tgz ./ <sup>″</sup>
#PJMstgout "./vge_output.tgz ./"
#PJMstgout "./output.1tgz ./"
#PJMstgout "./output.2tgz ./"
genomon_pipeline
dna ./sample/sample.csv ./output1 ./genomon.cfg ./dna_task_param.cfg > stdout1.log 2>
stderr1.log
genomon_pipeline
dna ./sample/sample.csv ./output2 ./genomon.cfg ./dna_task_param.cfg > stdout1.log 2>
stderr1.log

## テストケース No.4-1: Task3A からリスタート

Genomon の検体1と検体2の Task3A からのリスタート時に異常や不具合がないかテストする。 テストの確認項目を表 1.1.16 に示す。

確認対象	確認ファイル	確認内容	
検体 1 の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。	
検体 1 の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。	
検体1のTask1A のスキップ分岐 通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。	
検体1のTask3A のリスタート分 岐通過	stdout1.log	Task3A のリスタート分岐を通過したかデー タを確認する。	
検体1のTask4A の通常分岐通過	stdout1.log	Task4A の通常分岐を通過したかデータを確認する。	
検体1のTask5A の通常分岐通過	stdout1.log	Task5A の通常分岐を通過したかデータを確認する。	
検体1のTask6A の通常分岐通過	stdout1.log	Task6A の通常分岐を通過したかデータを確認する。	
検体1のTask7A の通常分岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認する。	

表 1 1 16	テストケース	No 1-1 のテス	ト確認値日
$\propto 1.1.10$	ノストクース	$10.4^{-1}07$	下推砣項日
検体1のTask2B	stdout1.log	Task2B の通常分岐を通過したかデータを確	
-------------	------------------------------	-------------------------	
の通常分岐通過		認する。 	
検体1のTask3B	stdout1 log	Task3B の通常分岐を通過したかデータを確	
の通常分岐通過		認する。	
検体1のTask4B	atdout1 log	Task4B の通常分岐を通過したかデータを確	
の通常分岐通過	stdout1.log	認する。	
検体2の変異ル	output2/mutation/sample_tumo		
ートの最終出力	r/sample_tumor_genomon_mu	オリシノルの変共ルートの取終面力ノアイル	
ファイルデータ	tations.result.txt	テータと比較し、回しか唯福する。	
検体 2 の SV ル	output2/sv/sample_tumor/sam		
ートの最終出力	ple_tumor.genomonSV.result.t	オリンノルの SV ルートの取終出力ノアイル	
ファイルデータ	xt	テーダと比較し、回しか確認9る。 	
検体2のTask1A			
のスキップ分岐	stdout2.log	Iask1A のスキップ分岐を通過したかテータ	
通過	5	を確認りる。	
検体2のTask3A			
のリスタート分	stdout2.log	Task3A のリスタート分岐を通過したかテー	
岐通過		タを確認する。	
検体2のTask4A		Task4A の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask5A		Task5A の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask6A		Task6A の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask7A		Task7A の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask2B		Task2B の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask3B		Task3B の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	
検体2のTask4B		Task4B の通常分岐を通過したかデータを確	
の通常分岐通過	stdout2.log	認する。	

# テストケース No.4-2: Task6A からリスタート

Genomon の検体1と検体2の Task6A からのリスタート時に異常や不具合がないかテストする。 テストの確認項目を表 1.1.17 に示す。

確認対象	確認ファイル	確認内容
検体 1 の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 1 の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。

表 1.1.17 テストケース No.4-2 のテスト確認項目

検体 1 の Task1A のスキップ分岐 通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体 1 の Task3A のスキップ分岐 通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体 1 の Task4A のスキップ分岐 通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体1のTask5A のスキップ分岐 通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体1のTask6A のリスタート分 岐通過	stdout1.log	Task6A のリスタート分岐を通過したかデー タを確認する。
検体1のTask7A の通常分岐通過	stdout1.log	Task7A の通常分岐を通過したかデータを確認する。
検体 1 の Task2B のスキップ分岐 通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体 1 の Task3B のスキップ分岐 通過	stdout1.log	Task3B のスキップ分岐を通過したかデータ を確認する。
検体1のTask4B のスキップ分岐 通過	stdout1.log	Task4B のスキップ分岐を通過したかデータ を確認する。
検体 2 の変異ル ートの最終出力 ファイルデータ	output2/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 2 の SV ル ートの最終出力 ファイルデータ	output2/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体2のTask1A のスキップ分岐 通過	stdout2.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体2のTask3A のスキップ分岐 通過	stdout2.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体2のTask4A のスキップ分岐 通過	stdout2.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体2のTask5A のスキップ分岐 通過	stdout2.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体2のTask6A のリスタート分 岐通過	stdout2.log	Task6A のリスタート分岐を通過したかデー タを確認する。
検体2のTask7A の通常分岐通過	stdout2.log	Task7A の通常分岐を通過したかデータを確認する。

検体2のTask2B のスキップ分岐 通過	stdout2.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体2のTask3B のスキップ分岐 通過	stdout2.log	Task3B のスキップ分岐を通過したかデータ を確認する。
検体2のTask4B のスキップ分岐 通過	stdout2.log	Task4B のスキップ分岐を通過したかデータ を確認する。

# テストケース No.4-3: Task7A からリスタート

Genomon の検体1と検体2の Task7A からのリスタート時に異常や不具合がないかテストする。 テストの確認項目を表 1.1.18 に示す。

確認対象	確認ファイル	確認内容
検体 1 の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 1 の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体1のTask1A のスキップ分岐 通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体1のTask3A のスキップ分岐 通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体1のTask4A のスキップ分岐 通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体1のTask5A のスキップ分岐 通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体1のTask6A のスキップ分岐 通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。
検体1のTask7A のリスタート分 岐通過	stdout1.log	Task7A のリスタート分岐を通過したかデー タを確認する。
検体 1 の Task2B のスキップ分岐 通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体1のTask3B のスキップ分岐 通過	stdout1.log	Task3B のスキップ分岐を通過したかデータ を確認する。

表 1.1.18 テストケース No.4-3 のテスト確認項目

検体1のTask4B のスキップ分岐 通過	stdout1.log	Task4B のスキップ分岐を通過したかデータ を確認する。
検体 2 の変異ル ートの最終出力 ファイルデータ	output2/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 2 の SV ル ートの最終出力 ファイルデータ	output2/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体2のTask1A のスキップ分岐 通過	stdout2.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体2のTask3A のスキップ分岐 通過	stdout2.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体2のTask4A のスキップ分岐 通過	stdout2.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体2のTask5A のスキップ分岐 通過	stdout2.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体2のTask6A のスキップ分岐 通過	stdout2.log	Task6A のスキップ分岐を通過したかデータ を確認する。
検体2のTask7A のリスタート分 岐通過	stdout2.log	Task7A のリスタート分岐を通過したかデー タを確認する。
検体2のTask2B のスキップ分岐 通過	stdout2.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体2のTask3B のスキップ分岐 通過	stdout2.log	Task3B のスキップ分岐を通過したかデータ を確認する。
検体2のTask4B のスキップ分岐 通過	stdout2.log	Task4B のスキップ分岐を通過したかデータ を確認する。

# テストケース No.4-4 : Task2B からリスタート

Genomon の検体1と検体2の Task2B からのリスタート時に異常や不具合がないかテストする。 テストの確認項目を表 1.1.19 に示す。

確認対象	確認ファイル	確認内容
検体 1 の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。

表 1.1.19 テストケース No.4-4 のテスト確認項目

検体 1 の SV ル	output1/sv/sample_tumor/sam	オロジナルの SV ルートの最終出力ファイル
ートの最終出力	ple_tumor.genomonSV.result.t	「オウシアルのるマルートの最終ロカシアール」
ファイルデータ	xt	
検体1のTask1A		   Task1A のスキップ分岐を通過したかデータ
のスキップ分岐	stdout1.log	を確認する。
通過		
検体1のTask3A		Task3A のスキップ分岐を通過したかデータ
のスキッノ分岐	stdout1.log	を確認する。
通過		
検体10 lask4A		Task4A のスキップ分岐を通過したかデータ
のスキッノ分岐	stdout1.log	を確認する。
」 地回		
一 使 体 T の フ キ い プ 公 は		Task5A のスキップ分岐を通過したかデータ
のスキッフカ岐	stdout1.log	を確認する。
<sup>  </sup> 地旭                            		
夜体 TO TaskoA	stdout1 log	Task6A のスキップ分岐を通過したかデータ
通過	studuttilog	を確認する。
<sup>施起</sup> 検体1のTask7A		
のスキップ分岐	stdout1 log	Task7A のスキップ分岐を通過したかデータ
通過	studuttilog	を確認する。
 検体1のTask2B		
のリスタート分	stdout1.log	Task2B のリスタート分岐を通過したかテー
岐通過	5	タを確認する。
検体1のTask3B		Task3B の通常分岐を通過したかデータを確
の通常分岐通過	stdout1.log	認する。
検体1のTask4B		Task4B の通常分岐を通過したかデータを確
の通常分岐通過	staout 1.log	認する。
検体2の変異ル	output2/mutation/sample_tumo	   オロジナルの恋卑ルートの最終出力ファイル。
ートの最終出力	r/sample_tumor_genomon_mu	オークと比較し、同じか確認する
ファイルデータ	tations.result.txt	
検体 2 の SV ル	output2/sv/sample_tumor/sam	   オリジナルの SV ルートの最終出力ファイル
ートの最終出力	ple_tumor.genomonSV.result.t	「データと比較し、同じか確認する。
ファイルデータ	xt	
検体2のTask1A		Task1A のスキップ分岐を通過したかデータ
のスキッノ分岐	stdout2.log	を確認する。
検体200 lask3A		Task3A のスキップ分岐を通過したかデータ
のスキッノ分岐	stdout2.log	を確認する。
」 地回		
快体 Z の Task4A   のフナップ公岐		Task4A のスキップ分岐を通過したかデータ
のスキッノカ岐	stdout2.log	を確認する。
<sup>- </sup> 地迎 検休 2 の Tack 5 A		
使体Z00 Task3A のフェップ分岐	atdout2 log	Task5A のスキップ分岐を通過したかデータ
のスイック力岐	studutz.log	を確認する。
<sup>通過</sup>		
のスキップ分岐	stdout2 log	Task6A のスキップ分岐を通過したかデータ
	sidout2.log	を確認する。

検体2のTask7A のスキップ分岐 通過	stdout2.log	Task7A のスキップ分岐を通過したかデータ を確認する。
検体2のTask2B のリスタート分 岐通過	stdout2.log	Task2B のリスタート分岐を通過したかデー タを確認する。
検体2のTask3B の通常分岐通過	stdout2.log	Task3B の通常分岐を通過したかデータを確認する。
検体2のTask4B の通常分岐通過	stdout2.log	Task4B の通常分岐を通過したかデータを確認する。

# テストケース No.4-5: Task3B からリスタート

Genomon の検体1と検体2の Task3B からのリスタート時に異常や不具合がないかテストする。 テストの確認項目を表 1.1.20 に示す。

確認対象	確認ファイル	確認内容
検体 1 の変異ル ートの最終出力 ファイルデータ	output1/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 1 の SV ル ートの最終出力 ファイルデータ	output1/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体 1 の Task1A のスキップ分岐 通過	stdout1.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体1のTask3A のスキップ分岐 通過	stdout1.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体1のTask4A のスキップ分岐 通過	stdout1.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体1のTask5A のスキップ分岐 通過	stdout1.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体1のTask6A のスキップ分岐 通過	stdout1.log	Task6A のスキップ分岐を通過したかデータ を確認する。
検体1のTask7A のスキップ分岐 通過	stdout1.log	Task7A のスキップ分岐を通過したかデータ を確認する。
検体 1 の Task2B のスキップ分岐 通過	stdout1.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体1のTask3B のリスタート分 岐通過	stdout1.log	Task3B のリスタート分岐を通過したかデー タを確認する。

表 1.1.20 テストケース No.4-5 のテスト確認項目

検体1のTask4B の通常分岐通過	stdout1.log	Task4B の通常分岐を通過したかデータを確認する。
検体 2 の変異ル ートの最終出力 ファイルデータ	output2/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 2 の SV ル ートの最終出力 ファイルデータ	output2/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体2のTask1A のスキップ分岐 通過	stdout2.log	Task1A のスキップ分岐を通過したかデータ を確認する。
検体 2 の Task3A のスキップ分岐 通過	stdout2.log	Task3A のスキップ分岐を通過したかデータ を確認する。
検体2のTask4A のスキップ分岐 通過	stdout2.log	Task4A のスキップ分岐を通過したかデータ を確認する。
検体2のTask5A のスキップ分岐 通過	stdout2.log	Task5A のスキップ分岐を通過したかデータ を確認する。
検体2のTask6A のスキップ分岐 通過	stdout2.log	Task6A のスキップ分岐を通過したかデータ を確認する。
検体2のTask7A のスキップ分岐 通過	stdout2.log	Task7A のスキップ分岐を通過したかデータ を確認する。
検体2のTask2B のスキップ分岐 通過	stdout2.log	Task2B のスキップ分岐を通過したかデータ を確認する。
検体2のTask3B のリスタート分 岐通過	stdout2.log	Task3B のリスタート分岐を通過したかデー タを確認する。
検体2のTask4B の通常分岐通過	stdout2.log	Task4B の通常分岐を通過したかデータを確認する。

# テストパターン No.5:チェックポイントデータに問題がある場合の Genomon リスタ ート実行

テストパターン No.5 のリスタート時のエラーメッセージのテストは、表 1.1.21 に示すテストケースの数だけ行う。テストフローは図 1.1.1。

テストケース	テストケースの内容
No.5-1	Task1A アウトプットファイルの中身に変更がある
No.5-2	Task1A アウトプットファイルが存在しない
No.5-3	Task1A チェックポイントデータが存在しない
No.5-4	検体1パイプラインエラー

表 1.1.21 テストパターン No.5 におけるテストケース一覧

テストパターン No.5 では Genomon の中間タスクのうち、Task3A、Task6A、Task7A、Task2B、 Task3B からのリスタート時に、チェックポイントデータに問題がある場合にエラーが起きるかをテ ストする。ひとつ前のタスクのチェックポイントマスタファイルおよびチェックポイントデータおよ びアウトプットディレクトリを参照してリスタートするため、Task1A、Task5A、Task6A、Task2B のデータ状況を変更してテストする。

テストパターン No.5 のステージング設定と入力設定を以下に示す。

#PJMstg-transfiles all
#PJMstgin "/public.tar ./"
#PJMstgin "/database.tar ./"
#PJMstgin "/sample.tgz ./"
#PJMstgin "/genomon.cfg ./"
#PJMstgin "/dna_task_param.cfg ./"
#PJMstgin "./vge.cfg ./"
#PJMstgin "./output.tgz ./"
#PJMstgout "./vge_output.tgz ./"
#PJMstgout "./output.tgz ./"
genomon_pipeline dna ./sample/sample.csv ./output ./genomon.cfg ./dna_task_param.cfg

> stdout1.log 2> stderr1.log

# テストケース No.5-1: Task1A アウトプットファイルの中身に変更がある

Task3A からのリスタート時に用いる Task1A アウトプットファイルの中身に変更があった時にエ ラーになるかをテストする。テストの確認項目を表 1.1.22 に示す。

表 1.1.22 テストケース No.5-1 のテスト確認項目

確認対象	確認ファイル	確認内容
エラーログ	stderr1.log	整合性判定で問題が見つかった時のエラーメッセージを確認する。

## テストケース No.5-2: Task1A アウトプットファイルが存在しない

Task3A からのリスタート時に用いる Task1A アウトプットファイルが存在しない時にエラーになるかをテストする。テストの確認項目を表 1.1.23 に示す。

確認対象
確認ファイル
確認内容
エラーログ
stderr1.log
アウトプットファイルが存在しない時のエラーメッセージを確認する。

表 1.1.23 テストケース No.5-2 のテスト確認項目

# テストケース No.5-3: Task1A チェックポイントデータが存在しない

Task3A からのリスタート時に用いる Task1A チェックポイントデータが存在しない時にエラーに なるかをテストする。テストの確認項目を表 1.1.24 に示す。

表 1.1.24 テストケース No.5-3 のテスト確認項目

確認対象	確認ファイル	確認内容
エラーログ	stderr1.log	チェックポイントデータが存在しない時のエラーメッセージを確認 する。

# テストケース No.5-4: 検体1パイプラインエラー

リスタート実行前に Task1A チェックポイントデータが存在しない場合に、エラーになるかをテストする。テストの確認項目を表 1.1.25 に示す。

確認対象	確認ファイル	確認内容
検体 1 のエラー ログ	stderr1.log	チェックポイントデータが存在しない時のエ ラーメッセージを確認する
検体 2 の変異ル ートの最終出力 ファイルデータ	output2/mutation/sample_tumo r/sample_tumor_genomon_mu tations.result.txt	オリジナルの変異ルートの最終出力ファイル データと比較し、同じか確認する。
検体 2 の SV ル ートの最終出力 ファイルデータ	output2/sv/sample_tumor/sam ple_tumor.genomonSV.result.t xt	オリジナルの SV ルートの最終出力ファイル データと比較し、同じか確認する。
検体 2 の変異ル ートのチェック ポイントアウト プットディレク トリの bam、 fastq、 mutation 下	output2/checkpoint/backup_Ro ute_A/bam、 output2/checkpoint/backup_Ro ute_A/fastq、 output2/checkpoint/backup_Ro ute_A/mutation	オリジナルのアウトプットディレクトリの bam、fastq、mutation 下を比較し、ファイル 名/ディレクトリ名が同じか確認する。
検体 2 の SV ル ートのチェック ポイントアウト プットディレク トリの sv	output2/checkpoint/backup_Ro ute_B/sv	オリジナルのアウトプットディレクトリの sv 下を比較し、ファイル名/ディレクトリ名が同 じか確認する。

表 1.1.25 テストケース No.5-4 のテスト確認項目

		以下のデータが書かれているか確認する。				
検体2の変異ル		Task3A,checkpoint_Task1A.chk の絶対パス				
ートのチェック	output2/checkpoint/checkpoint	Task4A,checkpoint_Task3A.chk の絶対パス				
ポイントマスタ	_Route_A.mst	Task5A,checkpoint_Task4A.chk の絶対パス				
ファイル		Task6A,checkpoint_Task5A.chk の絶対パス				
		Task7A,checkpoint_Task6A.chk の絶対パス				
検体 2 の SV ル		以下のデータが書かれているか確認する。				
ートのチェック	output2/checkpoint/checkpoint	Task2B,checkpoint_Task5A.chk の絶対パス				
ポイントマスタ	_Route_B.mst	Task3B,checkpoint_Task2B.chk の絶対パス				
ファイル		Task4B,checkpoint_Task3B.chk の絶対パス				
検体2のTask1A						
チェックポイン		ノアイル名は cneckpoint_laskTA.cnk じのる				
トデータ	_lask1A.chk	かを確認する				
検体2のTask3A						
チェックポイン		ノアイル名は checkpoint_Task3A.chk でのつ				
トデータ		とうは思うてき				
検体2のTask4A		フライルタは abadynaint Taak(AA abk でちろ				
チェックポイン		ノアイル名は cneckpoint_lask4A.cnk じのる				
トデータ	_lask4A.cnk	し、な確認する				
検体2のTask5A	evitevit2/ebe elve eint/ebe elve eint	コマイルタけ abaaknaint Taak54 abk である				
チェックポイン		ノアイル名は cneckpoint_lask5A.cnk じめる				
トデータ		いを確認する				
検体2のTask6A	output2/chackpoint/chackpoint	   ファイルタは checknoint Task6A chk である				
チェックポイン						
トデータ						
検体2のTask7A	output2/chackpoint/chackpoint	   ファイルタは checknoint Task7A chk であろ				
チェックポイン						
トデータ						
検体2のTask2B	output2/checkpoint/checkpoint	   ファイル名は checknoint Task2B chk であろ				
チェックポイン	Task2B chk	かち確認する				
トデータ						
検体2のTask3B	output2/checkpoint/checkpoint	   ファイルタは checknoint Task3B chk であろ				
チェックポイン						
トデータ						
検体2のTask4B	output2/checkpoint/checkpoint	   ファイル名は checknoint Task4B chk であろ				
チェックポイン						
トデータ						

# テスト結果

リスタート対応版 Genomon のテスト結果を以下に記す。ただし、異常系以外のテストは、整合性 判定の方法3種類それぞれを別の項目としてカウントしている。

テストケース		テスト項目数	障害検出数
一検体入力	通常実行	45	0
	リスタート実行	297	0
複数検体入力	通常実行	90	0
	リスタート実行	330	0
異常系		19	0

リスタート対応版 Genomon は障害が検出されなかった。以上のことから、本機能の動作は問題がないと判断する。

# リスタート機能性能評価

本件で開発したリスタート機能について性能評価を行った。性能評価を行った項目としては以下の通りである。

- リスタート機能組み込み前後の Genomon の実行時間
- チェックポイント書き出しに要する時間
- リスタートに要する時間

実行環境は「京」コンピュータ1ノード1プロセス会話型ジョブで、24 プロセス(23 ワーカー) 実行による結果である。検体解析の入力データとしては全ての検体で標準入力データを用いた。

## リスタート機能追加による Genomon の通常実行への影響

リスタート機能追加前後の Genomon 通常実行時間(リスタートデータを使わず Task1A から実行) を計測することで、リスタート機能追加による Genomon への影響を評価する。リスタート機能追加 前後の、3 回実行したときの Genomon の実行時間を表 1.1.26 に示す。

	Genomon 実行時間[秒]						
実行回数	機能追加前	機能追加後					
1 🗆 🖯	305.70	314.67					
2 🗆 🗏	291.68	330.10					
3 🗆 🗏	238.65	243.19					

表 1.1.26 リスタート機能追加前後の実行時間

リスタート機能追加により平均で6%実行時間が増加している。

次に、リスタート機能追加前後の、3回実行したときの各タスクの実行時間を表 1.1.27 に示す。

			各タスクの実行時間[秒]							
実行回数	機能追加前後	Task1A	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	TasK3B	Task4B
	機能追加前	4.49	3.77	38.15	61.61	167.26	2.16	157.54	7.15	8.11
108	機能追加後	10.64	5.01	35.45	62.07	150.72	2.21	145.66	8.19	7.20
	機能追加前	5.67	5.74	37.41	60.68	127.29	9.20	119.23	16.14	7.13
208	機能追加後	5.66	4.74	33.50	64.21	123.98	2.23	116.66	8.20	13.18
3 🗆 🗏	機能追加前	5.77	4.86	34.23	43.11	83.41	3.26	81.59	7.26	7.18
	機能追加後	48.78	4.86	33.62	40.05	80.30	2.31	80.38	7.17	8.16

表 1.1.27 リスタート機能追加前後の各タスクの実行時間

各タスクの実行時間について機能追加前後で確認すると機能追加前後で性能が大きく変わるケース は少ない。表 1.1.27 では、Task1A の 1 回目と 3 回目の機能追加後、Task7A、Task3B の 2 回目の機 能追加前が他の実行に比べ倍以上性能が遅くなっているが、Task4A、Task5A、Task6A、Task2B は 10%未満の差である。大きく性能劣化する場合があるが IO 性能によるブレの影響と考える。

# チェックポイント機能有効時の実行時間への影響評価

リスタート機能を組み込んだ Genomon を通常実行させ、各タスク終了時に実施する関数 write\_checkpoint の実行時間を評価する。各タスクの実行時間を表 1.1.28 に示す。

write_checkpoint 実行時間[秒]									
Task1A	Task3A	Task4A	task5A	Task6A	Task7A	Task2B	Task3B	Task4B	
0.88	0.38	1.10	9.21	0.58	0.79	1.40	0.92	0.51	

表 1.1.28 write\_checkpoint 実行時間

次に、各タスクの write\_checkpoint の処理ごとの実行時間を表 1.1.29 に示す。

		write_checkpoint 実行時間内訳[秒]							
	Task1A	Task3A	Task4A	task5A	Task6A	Task7A	Task2B	Task3B	Task4B
チェックポイントデータ作成	0.336	0.058	0.395	0.128	0.137	0.111	0.577	0.250	0.097
チェックポイントデータ書き 出し	0.012	0.007	0.010	0.028	0.008	0.011	0.010	0.023	0.007
退避ディレクトリの作成	0.002	0.001	0.000	0.002	0.000	0.000	0.001	0.000	0.000
ハードリンクによる退避	0.527	0.305	0.690	9.050	0.435	0.660	0.810	0.639	0.402

表 1.1.29 write\_checkpoint 実行時間内訳

内訳を見るとどのタスクもチェックポイントデータとハードリンクによるデータ退避に時間を要しているのがわかる。

各タスクで作成されるファイル数とそのファイルサイズを表 1.1.30 に示す。

	Task1A	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
ディレクトリ数	24	24	24	24	36	36	36	36	36
ファイル数	36	41	214	51	157	133	125	133	133
ファイルサイズ合計[MB]	81.0	81.0	102.0	8.9	9.0	9.0	8.9	9.0	9.0

表 1.1.30 各タスク終了時のファイル数およびファイルサイズ

最もファイル数、ファイルサイズが多いのは Task4A だが、最も時間が長いタスクは Task5A である。このことから、ファイル数やサイズが write\_checkpoint の実行時間を決めているわけではないと考えられる。

最も実行時間が長かった Task5Aの、3回実行したときの時間を表 1.1.31 に示す。

	Task5A の実行時間				
	1 🗆 🗏	2 回目	3 🗆 🗏		
チェックポイントデータ作成	0.127	0.111	0.114		
チェックポイントデータ書き出し	0.028	0.035	0.027		
退避ディレクトリの作成	0.002	0.002	0.002		
ハードリンクによる退避	9.050	0.578	0.670		

表 1.1.31 Task5A 複数回実行時の時間

この結果から Task5A の遅延は性能ブレの影響が考えられる。「京」ではファイルシステム要因による性能ブレがしばしば発生する。Genomon ではファイル操作を多数行っているため、ファイルシステムの影響を受けたものと考えられる。

# リスタート機能使用時の実行時間への影響評価

リスタート機能を組み込んだ Genomon をリスタート実行させ、各タスクのリスタート時に実施する関数 checkpoint\_restart の実行時間を評価する。3 回実行したときの各タスクの checkpoint\_restart の実行時間を表 1.1.32 に示す。

	checkpoint_restart 実行時間[秒]										
実行回数	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B			
1 🗆 🗏	1.68	1.80	2.63	2.51	2.80	1.59	3.00	2.12			
2 🗆 🗏	1.61	1.59	1.78	2.37	2.51	1.54	2.96	1.80			
3 🗆 🗏	1.34	1.91	2.25	2.60	2.70	1.47	2.92	1.76			

表 1.1.32 各タスクの checkpoint\_restart の実行時間

次に、表 1.1.32 に示した 1 回目、2 回目、3 回目の各タスクの checkpoint\_restart の処理ごとの実行 時間を、それぞれ表 1.1.33、表 1.1.34、表 1.1.35 に示す。なお、リスタートファイルの再配置ではハー ドリンクの作成を行っている。

	checkpoint_restart 実行時間内訳[秒]							
	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
チェックポイントマスタファイ ルの存在確認	0.01	0.01	0.01	0.04	0.02	0.01	0.01	0.01
チェックポイントマスタファイ ルの読み込み	0.01	0.00	0.03	0.04	0.02	0.01	0.01	0.01
タスクフラグのセット	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
チェックポイントデータのパス 設定	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
チェックポイントデータのオー プン	0.01	0.01	0.03	0.02	0.03	0.01	0.04	0.02
整合性判定	0.40	0.34	1.25	0.77	1.10	0.34	1.48	1.01
outputfiles の設定	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00
<ul><li>バックアップディレクトリの再</li><li>配置</li></ul>	1.25	1.44	1.31	1.64	1.64	1.21	1.45	1.05
合計時間	1.68	1.80	2.63	2.51	2.80	1.59	3.00	2.12

表 1.1.33 1 回目の checkpoint\_restart 実行時間内訳

表 1.1.34 2 回目の checkpoint\_restart 実行時間内訳

	checkpoint_restart 実行時間内訳[秒]							
	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
チェックポイントマスタファイ ルの存在確認	0.01	0.01	0.01	0.09	0.02	0.01	0.01	0.02
チェックポイントマスタファイ ルの読み込み	0.01	0.01	0.00	0.02	0.02	0.01	0.01	0.07
タスクフラグのセット	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
チェックポイントデータのパス 設定	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
チェックポイントデータのオー プン	0.01	0.01	0.03	0.02	0.04	0.01	0.04	0.02
整合性判定	0.56	0.34	0.70	0.60	0.77	0.36	1.08	0.74
outputfiles の設定	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
バックアップディレクトリの再 配置	1.01	1.22	1.04	1.65	1.66	1.15	1.82	0.94
合計時間	1.61	1.59	1.78	2.37	2.51	1.54	2.96	1.80

	abackpoint_rootart 宝行哇問内印[11]								
		Checkpoint_lestalt 美门时间内趴[秒]							
	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B	
チェックポイントマスタファイ ルの存在確認	0.01	0.01	0.01	0.01	0.01	0.02	0.01	0.01	
チェックポイントマスタファイ ルの読み込み	0.01	0.01	0.01	0.01	0.02	0.02	0.01	0.02	
タスクフラグのセット	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
チェックポイントデータのパス 設定	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
チェックポイントデータのオー プン	0.01	0.02	0.04	0.01	0.05	0.01	0.05	0.02	
整合性判定	0.30	0.55	0.90	0.32	0.84	0.38	1.15	0.50	
outputfiles の設定	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	
バックアップディレクトリの再 配置	1.00	1.32	1.28	2.24	1.77	1.04	1.70	1.21	
合計時間	1.34	1.91	2.25	2.60	2.70	1.47	2.92	1.76	

表 1.1.35 3 回目の checkpoint\_restart 実行時間内訳

次に、表 1.1.33、表 1.1.34、表 1.1.35 で示した中でも、特に実行時間が長い処理である整合性判定、 バックアップディレクトリの再配置について 3 回実行したときの実行時間を表 1.1.36 に示す。続けて、 各タスクのアウトプットファイルの数とそのサイズを表 1.1.37 に示す。

表 1.1.36 整合性判定、バックアップディレクトリの再配置の実行時間

		checkpoint_restart の実行時間[秒]								
処理概要	実行回数	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B	
	1 🗆 🗏	0.40	0.34	1.25	0.77	1.10	0.34	1.48	1.01	
整合性判定	2 🗆 🗏	0.56	0.34	0.70	0.60	0.77	0.36	1.08	0.74	
	3 🗆 🗏	0.30	0.55	0.90	0.32	0.84	0.38	1.15	0.50	
	1 🗆 🗏	1.25	1.44	1.31	1.64	1.64	1.21	1.45	1.05	
バックアッフティレク	2 🗆 🗏	1.01	1.22	1.04	1.65	1.66	1.15	1.82	0.94	
トリ冉配直	3 🗆 🗏	1.00	1.32	1.28	2.24	1.77	1.04	1.70	1.21	

表 1.1.37 各タスクのバックアップファイルとそのサイズ

	Task3A	Task4A	Task5A	Task6A	Task7A	Task2B	Task3B	Task4B
ファイル数	14	22	198	311	604	267	620	435
ファイルサイズ[MB]	85.08	169.58	359.09	412.73	416.30	376.18	415.79	423.78

表 1.1.33、表 1.1.34、

表 1.1.35 から、ファイルアクセスが発生するタイミングと整合性判定、加えてバックアップディレクトリの再配置時のハードリンクの作成によって、実行時間にブレが発生していることがわかる。なお、2 つのタスクからリスタートする場合、整合性判定とバックアップディレクトリの再配置が 2 つのタスク分実施されるため、実行時間は倍になる。表 1.1.36、表 1.1.37 から、整合性判定、リスター

トファイルの再配置の実行時間は、ファイル数に比例して増加する傾向にあることがわかる。

以上のテスト結果および評価によって、開発したリスタート機能の動作確認および性能評価が完了 した。Genomon を用いたテストでは、分岐を含む複雑な実パイプラインでの動作検証および複数パイ プラインの稼働状況においても問題なくリスタート出来ることが確認された。リスタート機能利用時 の実行時間についてもテスト結果から問題ないと思われる。

(c) 大規模解析

大規模解析は昨年度実施予定項目であったが、昨年度報告書で述べたようにファイルシステムに起 因する問題の原因解明に多大な時間を要したこと、また、それに起因するジョブ実行による計算資源 消費により完了することが不可能であった。今年度、本未完了項目を完了したためその内容を報告す る。

本解析の目的は、複数検体を同時に大規模解析した際の効率と性能を確認することである。VGE 本 体の性能を測定するため、テストコードは次世代シーケンサーのデータ解析で最も一般的な Fastq デ ータの分割および BWA によるアラインメントのみで構成されている。これは、ターゲットアプリ Genomon のパイプライン前半部分を簡略化したものに相当する。入力データはリード長 152bp、ペ アードエンドの全ゲノムシーケンスデータ (WGS) 14 個、データサイズは合計 4.2TB である。計算 には「京」コンピュータ 25,055 ノード、2,000,440 コアを使用した。本解析は約 3 時間で完了し、全 計算は正常に終了した。



図 1.1.12 大規模解析時の VGE ワーカーへのジョブ割当の様子

図 1.1.12 に同解析を 5,000 ノードで実施した際の VGE の動作状況を示す。ノード数を減らした理 由は、先の計算では各ワーカーへのジョブ割当は平均 1 回であり、VGE のマスターノードの性能確認 には適さないからである。5,000 ノード使用の場合、ワーカー数に対して VGE へ投入される総ジョブ 数の方がはるかに大きな数となるため、ジョブ振り分け性能を確認することが出来る。図 1.1.12 (b)が 3 時間実行した際の割当状況の全体像を表している。各ワーカーに対し、色付けされている部分が何 らかのジョブ割当が行われ、ワーカーがジョブを実行していることを意味している。同色のワーカー は同一のアレイジョブ(複数プロセスを使用するジョブのこと)を処理していることを意味している。 色は再帰的利用である。

図 1.1.12 (b)において、計算開始から 20 分程度までの間は稼働しているワーカーがごく僅かである

(図中(1)部分)。本部分ではインプットデータの分割処理を実施しており、必要ワーカーは解析検体 数の2倍、すなわち28である。ここで重要なことは、次にBWAアラインメントのタスクがあるにも かかわらず、ワーカーにジョブが投入されていないことである。次のタスクは実行中の分割処理の結 果ファイルをインプットとする依存関係を持つタスクである。通常のグリッドエンジンにおいてはこ のような依存関係を含むパイプラインは頻繁に登場するため、依存関係の適切な処理は必要不可欠な 機能であり、VGEにおいても適切に処理されていることが確認できる。次に、実行されるのが BWA によるアラインメントである。Fastq分割ジョブのうち、データ量がほぼ同程度で最少だった検体の 分割処理がほぼ同時に完了し、ワーカーに3検体分のジョブが直ちに投入された様子が確認できる(図 1.1.12 (b)の(2)部分)。

これ以降のワーカーの動作状況は、各ワーカーのジョブ終了時刻がバラバラであるため、非常に複 雑な様相となる。図 1.1.12 (a)は全体図のうちワーカー番号 1200 から 1250 までの約 50 ワーカーに ついて、1.2 時間から 1.3 時間の区間における実行状況を抜粋表示したものである。各ワーカーはそれ ぞれバラバラのジョブが割り当てられている。しかし、どのワーカーも実行中のジョブが完了次第、 直ちに新しいジョブが割り当てられ、次のジョブ実行が始まっていることが確認できる(図 1.1.12 (b) の(3)部分)。

以上のことから、VGE はパイプラインソフトウェアの大規模解析を効率的に実行可能であると結論 できる。

本成果は、IEEE International Conference on Bioinformatics and Biomedicine 2018 における regular paper として採択(採択率 19.6%)され、口頭発表および出版済み[1]である。

## 参考文献

 [1] Ito S, Yadome M, Nishiki T, Ishiduki S, Inoue H, Yamaguchi R, Miyano S. "Virtual Grid Engine: Accelerating thousands of omics sample analyses using large-scale supercomputers," 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Madrid, Spain, 2018, pp. 387-392. doi: 10.1109/BIBM.2018.8621285

## 【(1)の今年度計画と成果】

平成 30 年度は、大規模がんオミクスデータ解析システムの継続発展(高速化・高感度化)を図 り、Genomonに基づくポスト「京」の詳細設計コデザインを実施しつつ、高感度解析において、700 検体/日のパフォーマンスを達成することを目標にして、各機能を確認してシステムを発展させる ことを実施内容とする。ここで高感度解析とは、通常の SNV 検出や融合遺伝子検出に加え複雑な変 異解析(中間的範囲の欠失挿入やウイルス挿入検出などの構造変異検出等)を含む。

パフォーマンス目標については、既に「京」フルノードで約 400 検体/日を達成したため、ポスト「京」性能なら 700 検体/日を大幅に超えることを確認済みであり、具体的数値は NDA 条項に含まれるため公表不可だが、当初目標は達成された。

各機能を確認しつつシステムを発展させることに関し、任意状況における計算中断に対して自動 的に復旧ポイントを設定し計算を再開可能にするリスタート機能を実装した。

- (2) がんの起源と集団内多様性を理解するためのシステムの開発
- A) Genomon による大量シーケンスデータの解析によるがんの個性と時間的多様性の解明

正常組織において加齢に伴うクローン拡大と発がんの関係を示す研究が多く発表されているが、が んの発症に先立つクローン拡大に関する、経時的な変化や外的リスクへの依存性についてはまだよく わかっていない。我々は、139人の担がん患者ないし健常人から得られた 682 個の食道上皮検体につ いてゲノム解析を行った。その結果、生理的には正常とおもわれる食道においても、加齢とともに、 NOTCH1を主要な標的とするドライバー変異を獲得したクローンが加齢とともに拡大すること、また、 この過程は、高度の飲酒や喫煙によって促進することが明らかとなった(図 1.2.1)。ドライバー変異 を獲得したクローンは、幼少期から、食道全体にわたって多数出現し、加齢に伴って、複数のドライ バー変異を獲得しつつ、数の増加と大きさの拡大を認め、高齢者なると正常食道の大半がドライバー 変異を有するクローンに置換されていた。



図 1.2.1 正常食道上皮と食道がんのドライバー変異の頻度の比較(左)、正常上皮におけるドラ イバー変異と年齢の関係および環境因子の影響(右)

食道がんと比較して、正常食道では、NOTCH1 変異や PPM1D 変異などのドライバー変異が高頻度 に認められたが、変異の獲得時期を変異獲得率と樹形図から推定すると、多くの例で最初のドライバ 一変異は思春期の後期までにすでに獲得されており(図 1.2.2)、ドライバー変異の個数は、高度の飲 酒や喫煙によって有意に増加していた。ドライバー変異を獲得したクローンによる食道上皮の再構築 は、正常の加齢に伴う避けがたい変化であって、これに高度の飲酒・喫煙が加わることによって発が んに関与していることが示唆された。[1]



図 1.2.2 70 歳男性の健常食道上皮で認められたクローンの進化。系統樹(右)と採取したサンプ ルの位置(左)を併せて示した。進化に過程における分岐の年代と遺伝子変異を系統樹中にしめし た。

さらに、進化のプロセスを正確にとらえ、個別化・予防医療へと発展させるためには、次項のB)と も関連するが、単細胞レベルで遺伝子の発現と変異を同時に測定する系の開発が不可欠である。この 技術を用いて変異を有した細胞がどのような形質を獲得するかを解析している。図 1.2.3 の例では、 骨髄異形成症候群から二次性白血病獲得の際に高頻度に変異を獲得する NRAS の変異を有する細胞 と有しない細胞とを、我々の開発中の手法で解析した。NRAS 変異を有する細胞は正常造血幹細胞よ りも分化した前駆細胞に類似した遺伝子発現様式をとっていた。この手法を用いることにより、難治 性クローンに特徴的なシグナルを現在解析中である。



図 1.2.3 NRAS 変異を有する白血病クローンは造血前駆細胞に類似した遺伝子発現形質を有する。

## 参考文献

- [1] Yokoyama A, Kakiuchi N, Yoshizato T, Nannya Y. er al, Age-related remodelling of oesophageal epithelia by mutated cancer drivers, Nature. 2019 Jan;565(7739):312-317. doi: 10.1038/s41586-018-0811-x.
- B) がんの進化モデルの研究:異なる進化ダイナミクスを統一的に記述する新しいがんの進化シミュ レーションモデル

がんの発生は単一の正常細胞が変異の獲得と自然選択を繰り返し、悪性化する進化の過程と捉える ことができる。しかしながら、これまでに、複数の異なる進化様式ががんの進化過程に観察されるこ



図 1.2.4 MASSIVE の概略

とが、分かっている。例えば、腫瘍内がんのタイプによって、自然選択によって一腫瘍内の異なる複 数のゲノムを持つ細胞集団、すなわち腫瘍内不均一性が生み出されるダーウィン分岐進化型、また自 然選択がなくとも腫瘍内不均一性が形成される中立進化型が存在することが分かっている。このよう な異なるがんの進化様式が再現される条件を探索するためには、がんの進化シミュレーションを利用 した理論研究が有用だと考えられる。すなわち、がんの進化を再現するシミュレーションモデルを構 築し、そのモデルのダイナミクスを、パラメータを変えながら精査する感受性解析を行うことによっ て、異なる進化様式が再現される条件が理解できると考えられる。我々はこれまでに、エージェント ベースドモデルを元にしたがんの進化シミュレーションを構築し、予備的なパラメータ感受性解析を 行っているが、幅広いパラメータ空間をより効率的に探索するための新規パラメータ感受性解析ツー ルが必要とされていた。そこで本研究ではエージェントベースドモデルを対象とした新規パラメータ 感受性解析ツール MASSIVE を開発した(図 1.2.4, [1])。MASSIVE はまず初めにエージェントベース ドモデルを用いたシミュレーションを様々なパラメータで超並列にスーパーコンピュータ上で試行す る。さらに Post-processing ステップとして各パラメータセットから産出されたシミュレーション結果 を複数種の要約統計量を用いて評価し、各シミュレーション結果を可視化し(例えばがんの進化シミ ュレーションの場合は変異プロファイル)、画像データとして保存しておく。MASSIVE ではこれらの ステップで産出された膨大なシミュレーション結果を新規に開発した対話的可視化ツール MASSIVE viewer により可視化する。MASSIVE viewer は最大 4 次元パラメータ空間をインタラクティブに表示 することで直感的にシミュレーション結果を解釈することを可能とし、viewer 一つの統計量に付いて パラメータ空間全体での傾向を精査するための focused モードと、3 種類の要約統計量を比較するため の comparative モードの二種類のモードを用意している。

MASSIVE viewer の応用例を示すために我々は、ダーウィン分岐進化の起こる条件を探索すべく、 一次元格子上でのがんの進化を再現するエージェントベースドモデルを構築しその進化ダイナミクス を解析した。そのモデルでは以下の仮定をおく。



図 1.2.5 一次元格子上でのがんの進化モデル

- 変異のない一つの正常細胞から単位時間あたり一定の確率gで細胞分裂を行う(細胞は不死 化していると仮定し死亡確率は0とする)。
- 一回の細胞分裂で生まれた二つの娘細胞は n ~ Pois(m/2) 個変異を獲得し(すなわち分裂 一回あたり平均 m 個の変異が生まれる)、各細胞は変異を最大 3 個の変異まで蓄積できる。
- 一変異あたり増殖速度がf倍になる。すなわち細胞の分裂確率はg・fN (N = Σn: 細胞の持つ変異数)
- 一次元の格子上で細胞を増殖させる(図 1.2.5)。変異を蓄積するごとに分裂確率は増加し、さ

らに端の細胞ほどリソースにアクセスしやすく増殖しやすいとする。リソースの分布は半減 距離 d の指数分布でモデル化し、d が小さいほどリソースの分布に偏りがあるとする。

● すなわち細胞の分裂確率はg・f<sup>N</sup>・(2<sup>-(i-1)/d</sup>+2<sup>-(p-i)/d</sup>)/2 (g=0.0001:正常細胞の増殖確率, N: 蓄 積した変異の数, f:一変異あたりの増殖確率の増加率, i:細胞の位置, p:細胞数, d:リソース 分布の半減距離)となる。

便宜的にパラメータを m' =  $-\log 10(m)$ , f' =  $\log 10(f)$ , p' =  $\log 10(P)$  and d' =  $\log 10(d)$ と変 換すると、今回の解析では m' ∈ {1, 2, 3}, f' ∈ {0.1, 0.2, 0.3, · · · , 1.0}, p' ∈ {3, 4, 5} お よび d' = {1, 2, 3, 4, 5}のすべての組合せに付いて、シミュレーションを行った。また各パラメータ 設定について 50 回のモンテカルロ試行を行い、計 3 × 10 × 3 × 5 × 50 = 22,500 回のシミュレーショ ンにおいて、全計算時間は 1,046 × 22,500 = 23,528,298 CPU core 秒 ≈ 273 CPU core 日であった。 しかしながら、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 SHIROKANE を用いて数千並列で計算を行うことにより、数時間で計算を終えることができた。

ー次元格子上でのがんの進化モデルを用いた超並列シミュレーションの結果を MASSIVE viewer で探索した(https://www.hgc.jp/~niiyan/massive)。複数の腫瘍内不均一性を評価するための要約統計 量を複数種類用意して評価したが、comparative モードで比較してみるとどれも同じような傾向を示 した(図 1.2.6; https://www.hgc.jp/~niiyan/massive/result1)。それらを代表するものとして、Shannon index 0.1 (10%以上の変異に絞った場合の変異で得られたサブクローンの割合より計算した Shannon index)の傾向を focused モードで見たところ、リソースの偏りが少ない (d'が大きい) 場合は、変異 の効果が大きい (f'が大きい) と腫瘍内不均一性は小さく、変異の効果が小さい (f'が小さい) と腫 瘍内不均一性が大きい、すなわちダーウィン分岐進化が促進されることが明らかになった (図 1.2.7; https://www.hgc.jp/~niiyan/massive/result2)。リソースの偏りが大きい (d'が小さい) 場合は、変異 の効果が小さい (f'が小さい) 場合でも、大きい (f'が大きい) 場合でも、腫瘍内不均一性が大きく、 リソースの偏りによっても分岐進化が促進されることが明らかになった。以上の結果により、腫瘍内 不均一性を生成する進化原理の一端が MASSIVE によって明らかになり、MASSIVE の有用性が示さ れた。

また前年度まで行っていた、早期大腸がんの多領域シークエンスの論文を公開した[2]。



 $\boxtimes$  1.2.6 comparative  $\exists - ec$ 



図 1.2.7 focused モード

#### 参考文献

- Niida A, Hasegawa T, Miyano S. Sensitivity analysis of agent-based simulation utilizing massively parallel computation and interactive data visualization. PloS One 2019 14:e0210678.
- [2] Saito T, Niida A, Uchi R, Hirata H, Komatsu H, Sakimura S, Hayashi S, Nambara S, Kuroda Y, Ito S, Eguchi H, Masuda T, Sugimachi K, Tobo T, Nishida H, Daa T, Chiba K, Shiraishi Y, Yoshizato T, Kodama M, Okimoto T, Mizukami K, Ogawa R, Okamoto K, Shuto M, Fukuda K, Matsui Y, Shimamura T, Hasegawa T, Doki Y, Nagayama S, Yamada K, Kato M, Shibata T, Mori M, Aburatani H, Murakami K, Suzuki Y, Ogawa S, Miyano S, \*Mimori K.. A temporal shift of the evolutionary principle shaping intratumor heterogeneity in colorectal cancer. Nature Communications 2018 9:2884.

#### 【(2)の今年度計画と成果】

平成 30 年度業務計画は、単一細胞や単一細胞由来のコロニーから遺伝子変異および網羅的な遺 伝子発現を解析する系を確立および最適化する。遺伝子変異がどのような細胞にどのような順序・ 組合せで発生し、なぜそのような組合せが特段に選択されていくのかを解析し、腫瘍の起源と多様 性構築機構のモデルを検証する。

(2)の成果は、Genomon を用いた大規模シークエンスデータ解析では、139人のがん患者ないし 健常人から得られた 682 個の食道上皮検体についてゲノム解析を行った。その結果、NOTCH1を 主要な標的とするドライバー変異を獲得したクローンが幼少期から生じ、加齢とともに拡大するこ と、また、この過程は、高度の飲酒や喫煙によって促進することが明らかとなった。70歳という高 齢者になると正常食道の大半がドライバー変異を有するクローンに置換されているという驚くべ き事実が判明した。これにより、がんの発生と進化について世界初の知見がえられ、その成果を Nature (2019)に発表した。

また、がんの進化を再現するシミュレーションモデルを構築し、そのモデルのダイナミクスを、 大規模並列にパラメータを変えながら精査する感受性解析を行った。結果としてこれまでに知られ ている異なるがんの進化様式を統一的に記述しうるシミュレーションモデルを構築し、包括的なパ ラメータ感受性解析を行うことで、それぞれの進化様式が生み出される条件を明らかにした。この うち、先ずシミュレーション結果について、昨年度までのものを含め Nature Communications (2018)及び PLoS One (2019)に発表した。

(3) 疾患に関連する胚細胞バリアントの効率的な解析システムの開発

平成 30 年度は、平成 29 年度までに開発してきた統計モデルの利用も行いつつ、細胞が変異を蓄 積する過程を表す進化系統樹の性質なども利用し、複数のシークエンスデータが得られる場合にバ リアント検出を高精度化する統計モデルと関連するシステムを開発することを目標にする。従来低 頻度変異検出の問題を困難にしている大きな要因は、DNAに含まれる変異頻度が塩基の観測エラー の頻度と近づくにつれ、真の変異箇所と観測エラーによる見かけ上の変異箇所を区別することが難 しくなることである。

平成 30 年度は、これまで検討を行ってきたベイズモデル統合法をさらに発展させ、進化系統樹の

性質なども利用し、一患者由来の複数のシークエンスデータが得られる場合に、バリアント検出を高 精度化する方法の開発を進めた。

これまで開発していた、バリアントの検出手法では、単一のシークエンスデータが得られる場合 に、有用な情報をベイズモデルの統合により利活用していた。単一のシークエンスデータから得られ る有用な情報は、近隣のヘテロ SNP の情報 (Usuyama *et al.*,2014) や重複したペアリードの情報 (Moriyama *et al.*,2017 [1]) などであり、基本的にはシークエンサーにより得られるリードのモデル 化を行うことにより利活用ができていた。

しかし、シークエンスデータの数が複数得られる場合においては、そもそもの有用情報の性質もこ れまでのものとは大きく異なる。複数のシークエンスデータが存在する場合では細胞が変異を蓄積 させた過程(進化系統樹)の性質を利用することが可能である。

細胞が変異を蓄積させた過程は以下の図 1.3.1 のように、枝を蓄積した変異の集合、葉を細胞集団 とするような、根付き木の構造により表すことができる。通常この進化の過程に関しては、ヒトゲノ ムのサイズが、3 Gbp もあることから、Infinite site assumption という仮定が広くなされている。 この仮定は、一つの遺伝子変異に関しては、それを含む進化系統樹上の枝は高々一つであるという仮 定である。

これらの仮定のもとでは、進化系統樹が存在するもとで得られる変異のプロファイルと、ランダム ノイズの間では性質が大きく異なる。図 1.3.1 は、進化系統樹に従った変異プロファイルと、ランダ ムノイズのシミュレーション結果を表しており、系統樹由来の変異プロファイル部分の列ベクトル はクラスターを形成しやすい性質がある。この列ベクトルのクラスタリングに関する性質は、数学的 にも示すことができ、シングルセルやバルクシークエンスの場合に置いても成り立つ。



図 1.3.1 開発中の手法の出力を表したもの。A)シミュレーションにより発生させた系統樹。図の ように根付き木の構造で表せる。B)変異候補の有無を推定した結果。C) バイクラスタリングの 推定結果。この推定結果と変異候補の有無の推定結果をもとに真に変異があるかを出力する。 進化系統樹などは複数のシークエンスデータにまたがった統計モデルを作成する必要があり、今ま でのようにリードのモデル化に関して着目する方法では不十分である。そのため、我々はリードの特 性に加え、進化系統樹樹の性質など、複数のシークエンスデータ特有の性質をモデル化する階層ベイ ズモデルを考案した。

図 1.3.2 に、作成した階層ベイズモデルの概要を示す。この階層ベイズモデルは、先ほどあげた進 化系統樹の性質以外にも複数のシークエンスデータ間でのハプロタイプの一致性や、シークエンス リードの特性まで加味する。



図 1.3.2 開発中のモデルのグラフィカル表現。A) 詳細を全て表したもの。B) 開発中の手 法では大きく分けて、5 種類の観測変数と確率変数を用いた生成モデルを作成した。D: 変 異候補の Bayes factor の結果を表す。観測データに相当する。Z: 各変異に対し、変異候 補の有無を表す。C: バイクラスタリングの状態を表す。Θ: バイクラスタリング内部の変 異候補割合などを表す。ν:変異のプロファイルから対応する系統樹が作成可能かを表す。

現在、この階層ベイズモデルに基づき、バリアントを検出する仕組みを実装し、実データをもとに するシミュレーションによって性能評価を行っている。この性能評価により、現在開発している手法 が、過去に我々が開発した手法[2]以上の性能で変異検出が可能であるかを検証する。

# 参考文献

- Moriyama T, Shiraishi Y, Chiba K, Yamaguchi R, Imoto S, Miyano S (2017), OVarCall: Bayesian mutation calling method utilizing overlapping paired-end reads. IEEE Transactions on NanoBioscience. 16(2),116-122.
- [2] Moriyama T, Imoto S, Hayashi S, Shiraishi Y, Miyano S, Yamaguchi R, (2019), A Bayesian model integration for mutation calling through data partitioning. Bioinformatics. doi: 10.1093/bioinformatics/btz233

【(3)の今年度計画と成果】

平成30年度業務計画は、胚細胞変異同定手法として、発生初期に胚細胞に生じた変異をDNAシー クエンスデータから高精度に検出・同定する数理的手法の開発を進める。細胞分化の結果、組織 ごとに変異アレル頻度は異なるが、頻度が低い場合でも同定できるよう、ハプロタイプ情報およ びシークエンスエラー発生過程を反映した情報などの複数情報を統合した統計的生成モデルの構 成法を検討し開発を進める。

(3)の成果は、高精度にバリアント検出するために、これまで検討を行ってきたベイズモデル統 合法を発展させ、進化系統樹の性質なども利用し、一患者由来の複数箇所のがん組織のシークエ ンスデータが得られる場合に、バリアント検出を高精度化する方法を開発した。つまり、胚細胞 変異同定手法として、発生初期に胚細胞に生じた変異をDNAシークエンスデータから高精度に検 出・同定する数理的手法の開発を進めた。細胞分化の結果、組織ごとに変異アレル頻度は異なる が、頻度が低い場合でも同定できるよう、ハプロタイプ情報およびシークエンスエラー発生過程 などの複数情報を統合した統計的生成モデルの構成法を検討した結果、リードの特性に加え、進 化系統樹の性質など、複数のシークエンスデータ特有の性質をモデル化する階層ベイズモデルを 新たに考案し、頻度の低い変異を効率よく確定できた。このうち、先ず体細胞変異検出に関する 複数リード特性についてBioinformatics 2019に発表した。

### 2. データ同化生体シミュレーションによる個別化医療支援(サブ課題 B)

本成果報告書4-1.実施計画の2.「(1)全脳循環代謝シミュレータの開発と高度個別化医療支援への応用」についての成果は本節の

- (1) 個別化医療用データに基づく全脳循環モデルの構築
- (2) 全脳循環シミュレータの大規模化・高速化
- (3) 脳微小循環代謝シミュレータの開発

に述べられている。ここでは、医用画像から取得できる主要脳動脈と脳静脈を接続する脳血管モデルを構築した。格子ボルツマン法と適合サブドメイン法を用いて大規模な血流計算を安定して効率よく行う手法を開発し、全脳レベルの血流シミュレーションを実行した。さらに脳微小循環シミュレータにより、脳毛細血管網内の赤血球の流動、分配、ガス交換の特性を調べ、全脳毛細血管に均一に血液が供給される受動的メカニズムの一つを明らかにした。

また、同じく実施計画の2. 「(2)脳機能障害に関わる個別化医療支援のための大規模生体シミュレータの開発」についての成果は

- (4) 脳動脈瘤に対する血管内治療デバイスの計算力学モデル開発
- (5) 左心房内血流解析にむけた 4D-CT 画像からの左心房変異場推定方法の検証
- (6) 微小循環遊走細胞解析プラットフォーム
- (7) ヒト運動機能の神経筋骨格系シミュレータの開発
- (8) 発話・舌運動シミュレータの開発

に述べられている。脳動脈瘤に挿入するコイルやステントの形状に与える摩擦係数の影響を確認した。また 4D-CT 画像から患者個別の 3 次元的左心房動態を取得することに成功した。さらに、脳機能障害が引き起こす循環障害、運動障害、発話障害の診断および治療を支援するための大規模生体シミュレータを開発し、ヒト運動機能解析、構音機能分析、微小循環遊走細胞の動態解析を行なった。

#### (1) 個別医用画像データに基づく全脳循環モデルの構築(伊井、石田、和田)

### A) はじめに

脳血管構造の特徴として、脳形態に応じた経路、脳表層から脳内への段階的な経路構造、脳実質内 の動脈・静脈の微小循環ユニット、毛細血管網における閉じた循環構造、異なる血管経路間の吻合、 動脈・静脈で異なる経路構造が挙げられる。この複雑な3次元血管構造と脳血流動態の関連は未だ明 らかではなく、その物理的本質の解明、医学的知見の獲得および医療への展開が望まれている。全脳 規模および局所規模での血流変化を同時に評価するためには、数値シミュレーションによるアプロー チが有用であり、そのための脳血管モデルが必要不可欠である。2016年度より、医用画像および数理 モデルを併用した全脳レベルでの脳血管構造の特徴を模倣・制御することが可能な脳血管モデルの構 築に取り組んでおり、脳表面で接合する閉じた脳血管系の構築に成功している。2018年度は、これま でに構築した脳血管モデルを発展し、段階的に血管経路を生成することが可能なアプローチを提案し た。ヒト大脳表層を走行する軟膜動静脈までの大域血管網構造に対し、脳血管が持つ階層的な循環経 路を再現することに成功した(図2.1.1)。



図 2.1.1 ヒト大脳を対象とした脳血管モデルと脳表層における観察結果との比較

B) 方法

脳血管モデルの基本コンセプトは実形状と数理モデルの複合による半仮想的な個人別の脳血管である(図 2.1.2)。これにより、解剖学的な特徴を満足した主要な脳血管と仮想的に構築した医用計測では取得できない小さな血管からなる動脈・静脈の末端が繋がる脳血管を生成する。



図 2.1.2 医用計測データと数理モデルの複合による脳血管構築のコンセプト

本コンセプトのもと、マルチステージ血管分岐アルゴリズム(図 2.1.3)を新たに提案し、階層的な 脳血管構造を構築した。



図 2.1.3 マルチステージ血管分岐アルゴリズム

以下詳細を述べる。

#### (a) 医用画像に基づく脳形状および数 mm 径までの動静脈血管の構築

脳形状は MRI 3D-T1 画像より画像処理用ソフトウェア(Amira 5.4.2, ZIB, Germany)を用いて 構築した。この際、MRI 画像では大脳皮質の領域を区別することは難しいため、その内側にある白 質を抽出し脳表面の位置まで膨張させた後スムージングをかけ脳表面形状とした。脳表面は 792,650 個の三角形メッシュで構成されており、表面積は 102,658 mm<sup>2</sup> であった。また、脳血管は 4D-CT の 画像より画像処理ソフト(Amira 5.4.2, ZIB, Germany)を用いて構築した。

#### (b) 脳表面へ向けた数百 µm 径程度までの動静脈血管網の構築

医用画像から抽出した血管の末端から脳表面に向けて血管を構築する。この手順ではまず、脳表面 を複数の領域に分割する。任意の個数の三角形要素を無作為に選び出し、これらを起点にした領域拡 張により脳表面を複数の領域に分割する。各領域に静脈末端点1個、動脈末端点1~6個となるよう に血管を構築する。この手順によって、太い血管が脳全体にくまなく構築される。血管構築は CCO(Constrained Constructive Optimization)モデル(Karch et al., 1999, Comput Biol Med 29, 19-38)を参考にする。本手法では、ランダムな位置に端点を設定してから、血管路の体積が最小となる ように血管を構築する。はじめに、乱数を使って脳表面のランダムな位置に血管を構築する。乱数生 成には Mersenne-Twister を使用した。そして、追加した端点から最近傍の血管と接続することで新 たな分岐経路を構築する。血管を構築する際、脳表面形状の距離関数場を利用することで、脳表面か ら一定の距離内に限定して血管経路を構築する。脳表面から離れた位置に血管が構築される場合は、 線分を細文化し線分を構成する点の座標を修正することで疑似的に脳表面を沿う曲がった血管を構 築する。

#### (c) 脳表面における数十 µm 径程度までの動静脈血管網の構築

前節の手順で脳表面へ向けた血管構築をした後に、構築した点の末端点を始点として、さらに細い 血管の構築を行う。血管の構築アルゴリズムは、前の手順と同様に CCO モデルを参考にしたものを 利用する。この手順では、始点と同じ領域内に限定して端点を選択し、血管を構築する。Risser ら (2009, Neurosci 27, 185-196)によって、脳皮質を貫通する血管の密度は 8.44~15 本/mm<sup>2</sup>であり、 その中で動脈と静脈の本数の比は約 2:1 であると報告されている。本研究では構築した血管の末端 部で脳皮質を貫通していると仮定する。そのため、動脈と静脈の末端点の数の比が約 2:1 となるよ うにしながら、末端点の密度が 8.44~15 個/mm<sup>2</sup> となるまで血管を構築する。本研究では、静脈を 構築した後に動脈を構築する。次の手順で動脈と静脈の接続は近傍にある動静脈同士で行うため、動 脈を構築する際、動脈の末端点は静脈の末端点が存在する三角形メッシュに限定する。

#### (d) 動静脈を接続する血管の構築

実際の脳では動脈と静脈は脳皮質内の微小循環網で接続される。本研究では脳実質を貫通する細 動静脈および毛細血管網まで構築しないが、同じ三角形メッシュ内にある動脈と静脈の末端点を結 ぶことで、微小循環系を形成するユニットとする。動脈の末端点と静脈の末端点の数の比は2:1で あるため、1つの静脈の末端点に2つの動脈の末端点が接続される。

### C) 結果および考察

手順 b までの構築結果を図 2.1.4 に示す。なお脳表面領域を 4,000 個、動脈の末端点の数を 8,000

個、静脈の末端点の数を4,000 個とし構築を行っている。手順bまでの動脈の最小径は184 μmであ り、静脈は315 μmとなった。また動脈の末端点数のヒストグラムも併せて示している。ヒストグラ ムより、動脈の末端は各領域に1~6本あり、1~2本となる領域が8割近くを占めていることがわかる。 また注目すべきは、末端数が0となる領域が一つもできなかった点であり、これにより脳実質に不足 なく血液が行き渡るための必要条件が満たされていることがわかる。血管生成において分岐血管を作 成する際、いくつかの制約条件が課されるが、手順bの段階で領域数をさらに増やすと制約条件によ り末端血管を含まない領域が出ることを確認している。今回提案したマルチステージ血管分岐アルゴ リズムにおいては段階的に血管成長の領域を考慮していくため、設定した領域全てに末端血管を生成 することが可能となっている。



図 2.1.4 手順 a (青) および手順 b (赤) における脳血管モデルの結果(左図:動脈、中図:静脈)および領域あたりの動脈末端数のヒストグラム(右図)

次に、手順cまでを用いたモデル結果を検討する。本構築では、Risser ら(2009, Neurosci 27, 185-196)が報告している血管密度 (8.44~15/mm<sup>2</sup>) に合わせ、8.65/mm<sup>2</sup> となるよう末端点数を調整した。 構築したモデルの貫通血管径は動脈と静脈でそれぞれ 56.31 ± 13.43 μm、71.07 ± 16.75 μm (平均値 ±分散値) となり、 Duvernoy と Delon (1981, Brain Reseach Bulletin 7, 519-579)により評価された 血管直径 (動脈: 15~240、静脈: 20~125) の範囲内となった。

手順 d までを用いて構築した結果を検討する。図 2.1.5 は構築された動脈・静脈の血管と脳形状を 示している。ヒト大脳の表層血管の実験観察(Duvernoy と Delon, 1981, Brain Reseach Bulletin 7, 519-579)と比較したところ、微小血管の生成の仕方や血管間隙の大きさなど、本モデルは実際の血管 を定性的に良く再現していることがわかる。また、間接的な評価として、各大脳動脈がカバーする脳 表層領域を比較したところ(図 2.1.5)、特段の条件を課すことなく、本モデルは実際の状況を自然に 再現することができた。



図 2.1.5 各大脳動脈がカバーする脳表層領域の比較。左図:本モデル(青:ACA、緑:MCA、橙:PCA)。 右図:ZwanとHillen (1991, Stroke 22, 1078)による実験観察(黒ACA、白MCA、灰:PCA)。

# D) 今後の展開

# (a) 血管モデルの発展

実際の脳血管では血管同士が末端部以外で繋がる吻合が見られるため、本モデルを用いて血管吻 合をパラメトリックに作り出し数値シミュレーション結果を議論することは有用である。これに向 け予備解析を行った。図 2.1.6 に静脈の吻合を再現した結果を示す。ここで、手順 c の血管モデルの 本数に対して 1%および 5%の割合で吻合を構築した結果をそれぞれ示しており、吻合率を制御する ことで異なる血管経路構造が作り出すことができている。また、構築した静脈の吻合血管は、実際に 観察されているものと同程度の太さを持つことを定性的に確認しているが、血管吻合については実 験観察においても不明な点が多く、数値実験と併用してさらなる検討が必要と考えられる。



図 2.1.6 静脈血管の吻合モデル(左:吻合率 1%、右:吻合率 5%)。吻合血管は緑色で表示。

## (b) 3次元解析および次数低減化に向けた予備解析

2017年度までの脳血管モデルに比べ、本年度構築した脳血管モデルはより微細な血管経路構造を とるため解析規模がより大きくなる。そこで、3次元数値シミュレーションの予備解析として、血管 内流れをハーゲン・ポアズイユ流れと仮定した 0次元数値モデルにより脳血流を再現する。また本 解析は、3次元血流モデルの次数削減化の予備解析としての意義も有する。

0次元モデルでは、血管を直円管と仮定し血流をハーゲン・ポアズイユ流れと仮定し分岐部における流量保存則を適用し得られる分岐圧力の連立方程式を解くことで、圧力および流量を求めること

ができる。入口流量として 11,667 mm3/s を 4 つの動脈入口に等分配に与え、出口圧力を 0 Pa とし 解析した結果を図 2.1.7 に示す。現段階では、連立方程式の収束が不安定であることからプログラム 上のミスが内在している可能性があるが、脳全体を行き渡るように流量が求められそれに相当する 圧力分布を得ることができた。この際、出口流量が左側に偏っていることから、全脳血流場と合わせ た脳血管モデル構築の検討が必要であることが示された。

今後は、本モデルにおける流れの自由度および血管壁弾性など物理的な自由度を上げることで、三 次元血流解析の次数低減化にも取り組んでいく。



図 2.1.70 次元モデルを用いた脳血流解析の様子(左:流量、右: 圧力)。

## (2) 全脳循環シミュレータの大規模化・高速化(石田、今井)

## A) はじめに

我々はこれまで、脳循環血流場の解析のため、ボクセル型流体解析シミュレータを開発し、仮想的 な脳血管網モデルに対する血流動態の数値計算に成功している。一方で、従来のシミュレータには、 並列計算性能の面でいくつかの問題点があり、全脳循環を対象とするさらなる大規模計算を実施する にあたって、シミュレータの改善が不可欠であった。まず計算手法について、有限差分法ベースの半 陰解法型ソルバーを用いており、圧力に関する大規模なポアソン方程式の求解が必要であった。ポア ソン方程式の求解は反復が必要な収束計算となるのに加え、全ノード間での通信を複数回必要とする 非常に計算負荷の高い処理である。また、領域分割に関して、従来は 32×32×32 のボクセル格子で構 成される box を血管形状に沿って配置し、1 MPI ノードが 1 box の計算を担当する形で領域を分割し ていた。この方法は、簡便ではあるものの、計算が不要な壁格子点を多く計算領域に含むのに加え、 MPI プロセス毎に流体領域となるボクセル数が異なるために計算負荷にインバランスが生じ、並列性 能の低下を招く。これらの問題点に対し、計算手法の見直しと、領域分割法の改善を行い、シミュレ ータの大幅な高速化を達成したので報告する。

## B)方法

計算手法には、従来の半陰解法型手法に変えて、ポアソン方程式の求解が不要な完全陽解法型手法 である格子ボルツマン法を採用した。格子ボルツマン法はアルゴリズムの局所性が高く、大規模並列 計算に適した手法であるとして、近年注目されている手法である。本研究では、最近開発された、数 値安定性に優れた D3Q27Cumulant 格子ボルツマン法 (Geier et al., Computers & Mathematics with Applications, 2015)を用いた。また壁面境界条件として、符号付距離関数に基づく Interpolated bounce-back 法(Bouzidi et al., Physics of fluids, 2001)を適用し、複雑形状に対する no-slip 条件を高 精度に満足している。一般に完全陽解法型手法は半陰解法型手法に比べて時間刻み幅が小さくなり、 同じ時間に達するまでのタイムステップ数が多くなるが、1 タイムステップの計算負荷が圧倒的に小 さくなるため、総合的には計算が高速化される。

領域分割法として、我々が過去に開発した適合サブドメイン法(Miki et al., Computer methods in biomechanics and biomedical engineering, 2012)を用いた。この手法は、サブドメインと呼ばれる 2×2×2 のボクセル格子で構成される小領域ブロックを形状に適合するように配置し、個々の MPI / ードが担当するサブドメインの数が等しくなるように領域を分割する。従来手法では計算の必要のない壁格子点を多く計算領域に含んでいたのに対し、新手法では、壁面近傍のごくわずかな壁格子点のみが計算領域に含まれるため、計算リソースの効率的な運用が可能である。同時に、個々の MPI / ードが担当する流体格子点の数がほぼ等しくなり、計算負荷バランスが大幅に改善される。

## C) 結果および考察

以上構築した新手法について、ベンチマークテストにより従来手法との比較を行う。図 2.2.1A に示 すような動脈瘤を含む血管内の流れを対象とし、流体格子点の総数は約 92 万 voxel である。従来手法 と新手法で、1 MPI プロセスあたりの格子点数を等しくなるように(約 32,700 点)分割すると、従来 手法では 134 MPI ノード、新手法では 34 MPI ノードで領域分割される。このときの従来手法による MPI 領域分割を図 2.2.1B に、本研究の MPI 領域分割を図 2.2.1C に示す。色は MPI のランク番号を 表している。従来手法では 32×32×32 の box を用いて領域分割を行っているために、壁内部の不必要 な計算領域が多く存在しているのに対し、新手法では、血管形状に沿うように計算格子が生成されて おり、不要な計算領域が少ないことがわかる。また、従来手法では、box のごく一部しか流体領域と なっていない MPI プロセスが多数存在しているのに対し、新手法ではすべての MPI プロセスが同程 度の流体領域を占めている。図 2.2.1D は従来手法と新手法の流体格子点の3 倍以上存在しているのに対 し、新手法では、計算の必要のない壁格子点が流体格子点の3 倍以上存在しているのに対 し、新手法では、壁格子点の総数は流体格子点の 10%程度となっている。流体計算を時刻 0.1 秒まで 行うのにかかった時間の比較を図 2.2.1E に示す。新手法では従来手法の 1/4 の MPI ノードしか使用 していないにもかかわらず、1.6 倍程度の高速化を達成した。全脳循環を対象とする解析では、多数の 細い血管が脳表面を覆うように存在しており、新手法による恩恵はさらに大きくなると考えられる。



図 2.2.1 ベンチマークによる従来手法と新手法の比較。(A) 解析対象となる脳動脈瘤を含む血管内流 れ。(B) 従来手法による MPI 領域分割。(C) 新手法による MPI 領域分割。(D) 従来手法と新手法の 流体格子点と壁格子点数の比較。(E) 従来手法と新手法の実行速度の比較。

構築した手法を用い、全脳循環を対象とした大規模血流計算を実施した。我々が開発した、この医用画像と数理モデリングを組合せた脳血管網モデル(北出ら、バイオフロンティア講演会、2018)を 用いて直径 400µm までの血管網を生成し(図 2.2.2)血流計算を行う。



図 2.2.2 脳血管網モデル。赤:動脈血管、青:静脈血管

最も細い血管を十分に解像できるよう、空間解像度 60µm とする。このとき総ボクセル数約4億の 計算となる。これを「京」コンピュータ 20000 ノードを用いて計算する。このときの MPI 領域分割 を図 2.2.3 に示す。このような非常に複雑な形状に対しても、開発した手法によって MPI 領域分割が 適切に行われていることが確認できた。

計算結果のスナップショットを図 2.2.4 に示す。左右内頚動脈および椎骨動脈から造影剤を模擬した、色を付けた流体を一定量注入することで可視化を行っており、青は左内頚動脈、黄色は右内頚動脈、赤は椎骨動脈より流入した流体を表している。流入した流体が脳全体へ行きわたり、頚静脈へと収束している様子がわかる。本研究で構築したシミュレータにより、従来手法では成しえなかった規模の脳循環血流計算が実施できることを示した。ポスト「京」では、最大で「京」の 100 倍の実効性能と言われており、単純計算で空間解像度 10µm に迫る計算が可能になると期待される。毛細血管を除く脳血管は最も細いもので数十µm 程度であり (Duvernoy et al., *Brain Research Bulletin*, 1981)、本研究で構築したシミュレータをポスト「京」で用いることで、脳血管全てを解像するフルスケールの全脳循環血流計算が可能になることを示した。



図 2.2.3 脳血管網モデルに対する 20000 MPI ノードの領域分割



図 2.2.4 脳循環血流計算のスナップショット。左:左側面図、右:背面図
## (3) 脳微小循環代謝シミュレータの開発(伊井、石田、今井、武石)

A) はじめに

脳活動の正常な機能の実現には、活動にともなう酸素需要に応じた適切な酸素供給が必要である。 特に脳微小血管網周囲では均一な酸素濃度場が実現されていると考えられるが、その仕組みを知るた めには、複雑な血管構造に基づく血球分配に加え、個々の赤血球による酸素輸送を理解することが重 要となる。2017年度は、微小血管網を模擬したネットワーク流路モデルを用いて分岐における血球 分配挙動を明らかにするとともに、予備研究として酸素輸送モデルの構築と非連成解析結果を示し た。これに引き続き、2018年度は、両モデルを組合せることで、赤血球流入量を変化させて血球分 配の均一化や組織内の酸素濃度との関連性を調べた。赤血球流動に応じた酸素の組織供給の時空間変 化を数値再現することに成功し(図 2.3.1)、新たな知見として、分岐における血球分配は流体力学的 機構に従って受動的に調節され、血球分配および酸素濃度の均一化が高へマトクリット値にて生じや すいことを示した。



図 2.3.1 赤血球分配(上図)と酸素輸送(下図)の時間変化の様子。

## B)方法

本研究では、赤血球内部・血漿および組織の3相を考慮する(図2.3.2)。ただし、血液流動に関しては組織領域を固定壁とするため解析対象には含めないが、酸素輸送は組織相を含めた領域で解析される。



図 2.3.2 赤血球内部・血漿および組織の3相の定義

#### (a) 流体構造連成モデル

流体と赤血球膜の力学的相互作用を考慮するために Immersed Boundary 法(Peskin, 2002, Acta Numerica 11, 479-517)を用いる。流体の支配方程式として、連続の式と非圧縮 Navier-Stokes 方程式を採用する。計算格子は直交格子を用いる。任意形状の壁面の境界条件には、Boundary data immersion 法(Weymouth, Yue, 2011, J Comput. Phys 230, 6233-6247)を適用し、no-slip 条件を与える。数値解法には DD のフラクショナルステップ法(Dukowicz, Dvinsky, 1992, J Comput. Phys 102, 336-347)を用いる。赤血球は両凹円盤形状のカプセルとしてモデル化し、膜は超弾性体と仮定し Skalak の構成則(Skalak et al, 1973, Biophys J 13, 245-264)を適用する。赤血球膜は三角形要素の集合として表現され、Galerkin 有限要素法を適用することで節点上の膜力として求める。

#### (b) 酸素輸送モデル

血漿などの液体成分中に溶解する酸素濃度 C と分圧 P の関係は Henry 則に従うとする。赤血球 内部のヘモグロビンに対するオキシヘモグロビンの割合(酸素飽和度)すなわちヘモグロビンの酸 素化・脱酸素化は反応速度論に基づく方法(Clark et al., 1985, Biophys J 47, 171-181)と同様の定式 化を行う。また、酸素の消費は組織でのみ起こるものとし、Michaelis-Menten kinetics モデル (Goldman, 2008, Microcirculation 15, 795-811)を用いて酸素消費量を定式化する。各相での酸素輸 送は Lücker ら(2015, Am J Physiol Heart Circ Physiol 308, 206–216)の一次元輸送モデルを三次 元に拡張する。

我々がこれまでに採用しているボクセル格子手法と親和させるため、導入した基礎モデルに対し 各媒質相を表現する相関数により混合化を行うことで、場全体で成立する平均化方程式を新たに導 出した。これを有限差分法により解くことで酸素輸送の時空間分布を得る。

#### (c) 解析条件

図 2.3.3 に示す微小血管網を簡易的に模擬した流路モデルを用いる。流入口には一様流速 U = 0.5 mm/s、出口には圧力 p = 0 Pa を与える。流路周囲の組織領域の大きさは 204.8×102.4×25.6 µm3 とし、外側境界にはノイマン境界条件を適用する。ここでは、初期状態で酸素濃度 C = 0 の領域 に、酸素飽和度 S = 0.7 を所持する赤血球を一定時間間隔で投入することで、虚血状態の領域に酸 素が供給される状況を模擬する。流体の直交格子幅は 0.2 µm、計算の時間刻み幅は 5 µs とし、準 定常状態となる 1.2 s まで計算を行う。



図 2.3.3 微小血管網を簡易的に模擬した流路モデル(直径 S<sub>1</sub>: 8  $\mu$ m, S<sub>2</sub>: 9  $\mu$ m, S<sub>3</sub>: 7  $\mu$ m, S<sub>4</sub>: 8  $\mu$ m, S<sub>5</sub>: 9  $\mu$ m)

## C) 結果および考察

## (a) 古典的分配則に基づく血球分布と酸素供給

複雑形状における血球分配およびそれに基づく酸素供給について調べる。図 2.3.3 の微小血管網 モデルにおいて、赤血球を 6 µs の時間間隔で投入し、血球分配および周囲領域の酸素濃度場の様子 を観察する。図 2.3.1 に血球分布および中心断面における酸素濃度の分布の時間変化を示す。入口 から投入された赤血球は、はじめ管径の大きい下側の血管 (S2) に優先的に流れる。しばらくして 血管 S2 を十分な数の赤血球が流れるようになると、もう一方の分岐 (S1) にも流れ始める。その 後 S2 を通過した赤血球は、同様に管径の大きい S5 に優先的に流れ、しばらくすると S4 にも流入 する。また S1 を通過した赤血球は、S4 へと流れる。この時点で、赤血球が多く通過した血管 S2 付近の組織で特に酸素濃度が上昇している。さらに、十分に時間が経過すると、血管 S4 の赤血球 量が増加し、最も管径の小さい S3 にも流入し始める。しかしながら、赤血球が出口に到達した後 も、S3 では、依然として血球分配が促進されなかった。また、周囲組織の広範囲で酸素が供給さ れているが、赤血球数が少ない S3 付近では、組織への酸素供給が他の領域よりも進まなかった。

血管径が赤血球と同程度であるとき、分岐部における赤血球の分配は、Zweifach-Fung効果 (Svanes, Zweifach, 1968; Fung, 1973)として知られているように、流量(血管径)に応じて一定量 の比率になるように分配される。しかしながら、本研究で対象としたようなネットワーク流路内で は、下流域での赤血球分布によって局所的に管内の流動抵抗(見かけの粘度)が変化する。このと き、それまで赤血球が流れていなかった微小血管の流動抵抗は、それより大きい血管に対して相対 的に低下するため、赤血球は小さな血管にも流れ始める。したがって、分岐における血球分配は、 ある一定比の不均一な分布状態を基準としながら時間変動を繰り返すことが予想される。このよう にして、ネットワーク流路においても、時間経過を見れば血球分配が均一化の方向に受動的に調節 され、一日のうちで必要な酸素供給を促進していると考えられる。流動抵抗の変化を利用して細い 血管にも酸素が供給されるメカニズムを計算機上で実現できたという成果を得た。

### (b) 赤血球流量の違いが血球分配や酸素供給に与える影響

(a)の解析において、下流域での赤血球分布によって流動抵抗が変化し、血球分配が受動的に調節 されることを示した。ここでは、血球分配が赤血球流量によってどのように変化するかを確認す る。ここでは、赤血球流量を表すパラメータとして、流入ヘマトクリット値 Hp=QRBC/Q、を用い る。ここで、QRBC は赤血球流量、Q は血流場全体の流量を示す。なお、本研究においては Q を一 定としているが、QRBC が赤血球流入時刻ごとに時間、空間的に周期的に変動するため、入口血管の 長軸方向に対して空間平均、流入初期を除いた時間で時間平均をとり、平均赤血球流量を導出す る。赤血球流量の設定は、赤血球投入の時間間隔を変えることで行う。

本解析により、赤血球流入間隔を短くするにつれ流入ヘマトクリット値が増加することを確認した。各 H<sub>D</sub>に対して時刻 t = 0.6 s での血球分布と領域断面内の酸素濃度分布の計算結果を図 2.3.4 に示す。図 2.3.4(i)より、流入ヘマトクリット値が比較的小さい(H<sub>D</sub> = 0.171)場合には、大部分の赤血球が管径の大きい血管(S2、S5)に分布しており、その他の管径の小さい分岐には赤血球はほとんど分布しない。しかし、H<sub>D</sub>を増加させると、管径の小さい S2、S4 にも多くの赤血球が供給されるようになる(図 2.3.4(ii))。さらに H<sub>D</sub>を増加させると、S2、S4 に流入する赤血球はさらに

増加し、最も最も管径の小さいS3にも流入していることがわかる。このように、分岐における分 配挙動は下流域での血球分布を含めて赤血球の流量に依存しているといえる。

この理由は、流入赤血球量によって流動抵抗の変化量が異なるためであると考えられる。流動抵 抗の変化は粘度の高い流体からなる赤血球が局所に集中することによる見かけの粘度の増加に起因 する。赤血球量が増加するほど、局所への集中による流動抵抗の変化が顕著になるため、血球分配 の受動的調節機構が働きやすくなると考えられる。



図 2.3.4 異なる流入ヘマトクリット条件における酸素分配の様子(時刻 t = 0.6 s)。

## D) 今後の展開

本解析はボクセル型流体解析プログラム(VBB·Flow: Voxel based Brain Blood-flow)の派生プロ グラム(VBB·IBM)により実行された。本プログラムでは領域分割による MPI 並列を採用してい るため、ラグランジュ的に扱われる赤血球に関しては、その挙動に応じて MPI 担当領域において個 数が変化する。本年度報告した解析では、計算規模はさほど大きくないため、プログラム実装の容易 性を優先し、MPI の全ランクが全ての赤血球情報を保持している。しかしながら、実際の微小循環 を対象とした場合、計算規模が増え並列数の増大にともなうメモリ不足および通信コスト増大が起き るため、高効率化に向けたプログラム変更が必要となる。そこで、各 MPI が担当する空間領域に含 まれる赤血球のみ変数として保持し必要に応じて情報量を隣接ランクと通信できるようプログラムの 変更を行った。その予備解析として、二光子顕微鏡で得られた脳微小血管画像より構築した三次元モ デルを用いて数値シミュレーションを実行した(図 2.3.5)。なお本計算は「京」コンピュータシステ ムを用いおよそ 4,000 ノード(MPI)×8 コア(OpenMP)のハイブリッド並列で行われている。細動脈か ら流入した赤血球が毛細血管に分配されている様子が確認できる。次年度は並列効率の評価を行うと ともに、およそ数秒までの解析を行い血球流動と酸素供給の関係を明らかにしていく。



図 2.3.5 脳微小循環の数値シミュレーション結果(左図:赤血球分配、右図:圧力場)

## (4) 脳動脈瘤に対する血管内治療デバイスの計算力学モデル開発

脳動脈瘤に対する標準治療として、コイルやステントなど、金属製ワイヤで構成される血管内治療 用デバイスを瘤もしくは瘤の母血管へ留置し、瘤へ流入する血流を遮断する方法がとられる。この方 法は低侵襲性に優れる反面、デバイスのサイズや種類の選択、留置手順が経験的に決定され、瘤内に おけるコイルの偏在や逸脱、またステントの展開不全といったリスクを常に抱えている。しかし、デ バイスの力学挙動は、デバイスを構成するワイヤの大変形や複数個所での摩擦接触などの複合要因に より決定され、デバイスが最終的に取りうる留置後形状について、力学的見地からの理解はほぼ試み られてこなかった。本年度において、治療過程におけるこれらのデバイスの力学挙動の解明にむけ、 ワイヤの弾性抵抗や摩擦を伴う接触を考慮したデバイスの計算力学モデルの開発を行った。

治療用デバイスの挙動の表現にあたり、デバイスを構成するワイヤを Kirchhoff's rod theory に基 づきモデル化し、ワイヤの弾性抵抗として長軸方向に対する引張圧縮、曲げ、ねじりの3つの弾性エ ネルギーを定義した。ワイヤを2節点で構成される梁要素で有限要素分割し、Battiniと Pacoste (*Compt. Methods Appl. Mech. Eng.*, 2002)の共回転定式化により梁要素の内力を求めた。本定式化 では、梁要素の各要素に独立な座標系(要素座標系)を定義し、まず要素座標系において微小変形・ 微小ひずみの仮定のもとで定義される内力を求め、次に要素座標系の回転を考慮して、要素座標系か ら全体座標系へ内力を座標変換する。これにより幾何学的非線形性を伴うワイヤの大変形を効率的に 扱うことが可能となる。ワイヤと血管壁、もしくはワイヤ間の接触力について、法線方向の接触力は ペナルティ法を用いて設定し、接線方向の接触力は slip-stick を考慮した一般化 Coulomb 則に従う摩 擦力を与えた。以上の内力と接触力をワイヤの運動方程式に与え、逐次的に解くことで、大変形およ び摩擦接触を考慮してワイヤの力学挙動を表現した。

計算例の一つとして脳動脈瘤に対するコイル塞栓を考え、脳動脈瘤を想定した球(径 6 mm)に対 するコイル(全長 250 mm)の挿入シミュレーションを実施した。実際のコイルの物性を先述のワイ ヤの計算力学モデルに設定することでコイルの力学挙動を表現した。コイルの無負荷時形状は直線状 と簡略化し、直線状のカテーテルの先端を瘤内部に設定し、コイル末端に強制変位を与え、カテーテ ルを通じてコイルを瘤内へと留置した。コイル挿入の初期段階において、コイルは瘤壁面に沿って規 則的かつ二次元的なループ構造をとるが、接触時における摩擦係数の違いにより、コイルの最終的な 分布形状は異なった(図 2.4.1)。摩擦係数が低い場合、瘤内のコイル分布は規則的なループ構造を維 持するが、摩擦係数が高い場合には挿入過程でコイル分布は不規則な構造へと遷移した。ここで摩擦 係数が高くなるにつれ、コイル分布は挿入過程のより早い段階で不規則な構造となった。

コイル分布の摩擦係数への依存性について考察するため、挿入過程においてコイル末端に生じる反 力を調べた(図 2.4.2)。コイル挿入の初期段階において、コイル末端に負荷される反力は単調に増加 し、急に減少したのち(図 2.4.2(a)-(d))、再び増加に転じた。このときの瘤内におけるコイル形状を確 認すると、カテーテルから解放された直後のコイルの軸方向に対する座屈が観察された。この原因と して、瘤内に配置されたコイルと瘤壁面との摩擦抵抗に対するコイルの降伏が考えられる。また、コ イルの座屈後構造は摩擦係数に応じて2つのパターンが見られた(図2.4.3)。摩擦係数が低い場合、 瘤内に留置されたコイル全体が瘤表面に沿って滑り、その規則的な構造を維持した(図 2.4.3(a), (b))。 一方、摩擦係数が高い場合には、瘤内に留置されたコイル全体の滑り運動は生じず、カテーテルから 解放された直後のコイルに大きなたわみが発生し、これが瘤内のコイルと干渉することで、規則的な コイル構造を崩壊させ、不規則な構造を形成し始めた(図 2.4.3(c), (d))。これらの結果は、摩擦抵抗 によるコイルの軸方向座屈および座屈後のコイルの挙動が、瘤内におけるコイルの分布形状の傾向を 大きく変化させる要因であることを示唆する。以上より、構築した計算力学モデルおよび得られた結 果は、コイルが瘤内で取りうる分布形状について、力学的見地からの理解への糸口となり得る。つま り、従来、デバイスのサイズや種類の選択、留置手順が経験的に決定され、瘤内におけるコイルの偏 在や逸脱、またステントの展開不全といったリスクを常に抱えている医療現場の現実を鑑み、現時点 から遡る「後ろ向き評価」に資するよう、今回力学的知見を引き出した。



図 2.4.1 コイル挿入過程のスナップショット。接触時の摩擦係数は上段から下段に向けて 0.02、0.1、 0.2 と設定した。



図 2.4.2 脳動脈瘤への挿入過程においてコイル末端に負荷される反力と変位の関係。



図 2.4.3 図 2.4.2 中(a)-(d)時点以降のコイル分布形状のスナップショット。

#### (5) 左心房内血流解析にむけた 4D-CT 画像からの左房変位場推定方法の検証

本研究では、左心房内血栓症の患者別リスク評価にむけ、患者個々の状態を反映した左心房内血流 の数値流体計算手法の構築を目指している。左心房は自ら能動的に伸縮することで左心室の補助ポン プとして機能しており、血流計算にあたって、左心房形状だけでなく動態を考慮した患者個別解析が 望まれる。これまでに我々は 4D-CT 画像から左心房の変位場を推定し、患者個別の左心房運動の取得 および数値流体計算への適用を試みてきた (Otani et al., *Ann. Biomed. Eng.*, 2016)。本年度はこの 推定手法を見直し、適切なパラメータの設定のもとで、本推定方法の妥当性と限界について検証した。

4D-CT 画像に基づく左心房変位場推定のフレームワークを図 2.5.1 に示す。4D-CT 画像計測では、 胸部 CT 撮影を複数の拍動周期にわたり連続して実施し、心電図同期により、一心拍の間を全 20 位相 (0%PP-95%、PP は心電図における P 波の間を指す)に分け、各位相の心臓形状を三次元 CT 画像 として出力する。各位相における CT 画像から、画像処理ソフト Mimics 21.0 (Materialize Inc.)を用 いて左心房輪郭形状を三角形要素の集合として表現する。左心房の変位場推定にあたり、左心房形状 を構成する点群について、0%PPの点群を制御点群、他の19位相の点群(5%PP-95%PP)を観測点 群に割り当て、制御点群と観測点群との位置合わせ問題を解く。点群間の位置合わせには Myronenko と Song (2010)が開発した Coherent point drift (CPD)法を用いており、本手法は点群の位置合わせ問 題を混合確率分布の推定問題として置き換え、EM アルゴリズムを用いた収束計算により、観測点群 と一致するように制御点群の変位場を推定する。この操作により、点群間のデータの欠損やノイズに 対するロバスト性を持たせている。

パイロットケースとして、自治医科大学呼吸器外科より提供された3症例の胸部4D-CT 画像から、 左心房の変位場推定を実施した(図2.5.2)。CPDにより推定した左心房形状はCT 画像から抽出した 観測形状と良好に一致し、全位相の推定結果において非現実的な壁面の交差は生じなかった。観測形 状と推定形状との最近接距離はおおむね0-2 mmの範囲であり、CT 画像のスライス厚が1 mm であ ることを勘案すれば、現在のCT の解像度の範囲では許容できる程度の差と考えられる。なお、解像 度の限界から、左心房の収縮期には左心耳が観測形状において欠損する症例が見られた(図2.5.2 下 中央)が、推定結果において、これらの領域の変位場は周囲点群の変位に基づき補間されており、欠 損データに対する本アルゴリズムのロバスト性を示すものである。これらの結果から、左心房の大域 的な変位場推定手法として、本フレームワークの有用性を示せた。今後は本フレームワークを拡張し、 時間方向に連続な変位場の推定を実施することで、数値流体計算における移動壁境界として直接的に 応用可能な左心房動態が取得できると考える。「後ろ向き評価」の一環として、現在の患者データと CPDによる推定結果が良好に一致することが示され、原因を探る方法が前進した。



図 2.5.1 4D-CT 画像からの左心房変位場推定のワークフロー



図 2.5.2 3 症例における CT 画像から抽出した左心房の観測形状(左)と推定形状(右)(20%PP、40%PP、60%PP、80%PP)。カラーバーは推定形状と観測形状との最近接距離を表す。

## (6) 微小循環遊走細胞解析プラットフォーム(松田)

微小循環系での細胞の動態を非侵襲で長時間観察できる蛍光顕微鏡で得られる膨大な時系列画像デ ータを高精度かつ高速に処理するための、大規模ライブセルイメージング解析技術を「京」を使って 開発している。平成 30 年度は、前年度に引き続き、生体内部を精密かつ経時的に観察できる二光子 励起顕微鏡を使って、微小環境でのライブイメージングデータを取得し、炎症の進行に伴う血管内部 の変化をイメージング解析により解明することで、大脳循環のシミュレーションのデータ同化に必要 な計測データを取得することを目指した。

具体的には、炎症を誘発する刺激剤の投与(以下、炎症刺激と略す)による血管内での炎症進行の 経過でモデル化している。炎症刺激後の血管内皮の状態変化を、緑色蛍光で標識した白血球(炎症時 に血管内皮の細胞に接着・浸潤することが知られている)の動きの変化として観察している。

これまでに取得したライブイメージングデータは、マウス生体内の微小環境中を二光子励起顕微鏡 で撮影した経時観察動画像であり、毎秒最大 30 フレーム、各フレーム 512×512~2048×2048 ピクセ ル、24 ビットカラーであり、1 回の観察ごとに 60~2000 フレームの画像を取得している。

平成30年度には新たに、次の2点について取り組んだ。

- ① 深層マッチングの3次元化
- ② 炎症状態の変化に伴うマクロファージの動態解析

①の深層マッチングは、コンピュータビジョンの分野で開発された動画像中の物体の追跡手法であ り、深層学習(畳み込みニューラルネットワーク)に着想を得ている。小領域間の畳み込み演算の結 果をプーリングによりサブサンプリングしてマッチング結果のマップを階層的に作成する。 前年度までに、二光子励起顕微鏡で観察された生体組織画像での細胞の追跡に深層マッチングを応 用したものを「京」上で実装していたが、顕微鏡の光軸に垂直な2次元平面に投影した細胞の動きし か追跡できていなかった。しかし、二光子励起顕微鏡は、焦点深度を変えて観察することで、光軸方 向の深さを変えながら撮像することが可能である。そこで、平成30年度は、深さの異なる複数のスラ イス画像を使って、深さ方向の成分を含む3次元の動きでも解析できるようにプログラムを拡張した。

図 2.6.1 は、深層マッチングによるマウス生体組織中の白血球の動きの検出を、2 次元平面上に限定 した場合(図 2.6.1 左)と、3 次元への拡張により深さ方向の動きも検出した場合(図 2.6.1 右)を表 している。検出された細胞の移動ベクトルで、2 次元平面上のベクトルを白色、深さ方向で手前方向 のベクトルを赤色、奥の方向のベクトルを水色で表示している。3 次元化により、従来は検出できな かった移動が検出できていることがわかる。



図 2.6.1 深層マッチングの 3 次元化による深さ方向の白血球の動きの検出例。左が従来の 2 次元平面上の検出例で右が 3 次元化による検出例(赤色が深さ方向で手前方向、水色が奥の方向の移動ベクトルを 表す。

②では、高脂肪食負荷をかけたマウスの脂肪組織内で生じる炎症状態の変化に伴うマクロファージ の動きを解析し、炎症状態の遷移を検出することを試みた。マクロファージは白血球の一種で、体内 に生じた変性物質や侵入した細菌などの異物を捕食して消化する役割を果たすが、炎症の際に活発に 遊走することが知られている。

実際に、マウスに高脂肪食を与えたときに脂肪組織内でのマクロファージの活動を二光子励起顕微 鏡で観察した例を図 2.6.2 に示す。図 2.6.2 で赤い領域が脂肪細胞で、緑色に光るのがマクロファージ を表している。高脂肪食ではなく普通食を与えたマウスの脂肪組織(図 2.6.2 左)では、マクロファー ジは点在しているが、高脂肪食を与え続け(以下、高脂肪食負荷と略す)、数日すると、炎症が起こり 始めてマクロファージが活発に動き出し、図 2.6.2 中央(高脂肪食負荷5日)のように一部にマクロ ファージの集積が見られる。さらに、高脂肪食負荷を数週間続けると、図 2.6.2 右(高脂肪食負荷8 週)のように、脂肪細胞の周囲をマクロファージが取り囲む王冠様構造(Crown-like Structure)と 呼ばれる特徴的な構造が現れる。王冠様構造内部では、細胞死に陥った脂肪細胞をマクロファージが 取り囲んで貪食・処理しており、この構造を中心として脂肪組織の慢性炎症が生じる。その結果、脂肪組織の線維化が起こり、糖尿病や虚血性心疾患のリスクが高まるとされている。





高脂肪食負荷5日

高脂肪食負荷8週

図 2.6.2 脂肪組織内でのマクロファージの観察画像

本研究では、炎症と非炎症状態の違い、および慢性炎症と炎症開始期の違いを、ライブイメージン グ画像解析により得られるマクロファージの動態により判別することを試みた。具体的には、図 2.6.2 の3種類の動画像に対して深層マッチングを適用し、それぞれでのマクロファージの移動ベクトルの 大きさの分布を比較した(図 2.6.3)。図 2.6.3 では3状態の相互の分布の比較について Mann-Whitney 検定を行っており、非炎症状態(普通食)、炎症開始期(高脂肪食負荷5日)、慢性炎症(高脂肪食負 荷8週)は相互に統計的に有意に異なることが示された。この結果、二光子励起顕微鏡でマクロファ ージを観察して、深層マッチングにより移動ベクトルを求めることで、脂肪組織における炎症の進行 を判別できる可能性が示唆された。



図 2.6.3 図 2 の 3 種類の状態でのマウス脂肪組織中でのマクロファージの移動ベクトルの分布の比較 (7) ヒト運動機能の神経筋骨格系シミュレータの開発

A) はじめに

本研究テーマでは、ヒト運動制御メカニズムの解明および神経疾患に起因する運動障害発現メカニ ズムの解明に資することを目的とした大規模計算ニューロメカニクス研究の基盤プラットフォームと しての神経-筋-骨格系の統合シミュレータの開発を行っている。昨年度までに、医用画像から再構築さ れた3次元筋骨格モデルにおいて、ミクロな収縮タンパクの確率的な挙動から、骨格筋の収縮挙動お よび関節運動に至る筋-骨格系のマルチスケールシミュレーションが実施できるようになった (Yamamura et al., IEEE 16th Intern. Conf. on Bioinf. & Bioeng., pp. 338-341, 2016.)。

本年度は、ヒト静止立位時の姿勢制御を研究対象として、本シミュレータによるヒト静止立位時の 姿勢ゆらぎを再現する神経筋骨格系モデルの構築を行った。特に、解剖学的・力学的に精緻な静止立 位保持のシミュレーションモデルの開発を行い、静止立位維持のシミュレーションを介して、静止立 位姿勢制御に関する従来仮説(教科書レベルの定説)と最新仮説の妥当性に関する比較検討を行った。

本年度の開発により、ヒト静止立位の姿勢制御シミュレーションにおいて、筋骨格系のミクロな収 縮タンパクのレベルからマクロな関節運動に至るマルチスケールシミュレーションが可能になり、運 動制御メカニズム解明のための筋骨格系シミュレーションプラットフォーム開発の大きなマイルスト ーンを達成した。

#### B) ヒト静止立位時の姿勢ゆらぎのシミュレーション

ヒトの静止立位の姿勢ゆらぎの観測は、運動障害疾患の早期発見や早期治療につながる臨床的に重 要な知見を与えるとともに、ヒトの運動制御メカニズムの理解という神経科学的にも価値のある情報 を得ることが可能である。本年度は、ヒト静止立位シミュレーションモデルの構築を行い、ヒト静止 立位時姿勢ゆらぎの再現を通じて、静止立位姿勢制御に関する従来仮説(定説)とそれを刷新する可 能性のある最新仮説に関する検討を行った。



図 2.7.1 3D musculo-skeletal model of lower limb for stationary standing simulation (a) ヒト静止立位シミュレーションモデル

ヒト静止立位シミュレーションモデルは、図 2.7.1 に示すように、アキレス腱を含む下腿三頭筋モ

デルと下肢骨格モデルから構成される。下腿三頭筋は医用画像から作成した有限要素モデルで、筋 内にはその起始と停止をつなぐ複数の筋線維を設置した。各筋線維内には直列に複数のサルコメア モデルが設置され、各サルコメアモデルでは、収縮タンパクの確率的な挙動から収縮力が計算され る。骨格は剛体リンクモデルとし、足関節は回転関節として設定した。下腿三頭筋(腓腹筋・ヒラ メ筋)およびアキレス腱の剛性は文献(Blemker et al., J. Biomech. 38, pp.657-665, 2005., Rosager et al., Scand. J. Med. & Sci. in Sports, 12, pp.90-98, 2002)より、骨と筋および腱の付着位置は解 剖学的な情報(Schunke et al., Prometheus LernAtlas der Anatomie, 2005)。から設定した。

## (b) ヒト静止立位保持のシミュレーション

ヒト静止立位姿勢制御の従来仮説では、前方への微小転倒からの回復は、下腿三頭筋の伸長によ って復元力が発生し、姿勢を後方へ引き戻すことで生じるとされ、これの繰り返しが静止立位時の 姿勢ゆらぎとなる。また、アキレス腱は筋に比べて剛性が高いため伸長せず、姿勢制御に寄与しな いと考えられている。

そこでまず初めに、従来仮説を検討するため、足底部を固定し、重心に前方初期変位を与えて、足 関節まわりの運動シミュレーションを行い、アキレス腱の伸長および重力と筋の復元力から釣り合 いが安定に保持できるかを検討した。

その結果、アキレス腱の伸長は筋の伸長の 1/3 程度生じることが明らかになり、アキレス腱の伸長 が姿勢制御における弾性回復に寄与することが示唆された。これは、アキレス腱の剛性は筋の剛性と 比べて 1,000 倍以上大きいがアキレス腱の断面積が小さいために伸長したと考えられる。

図 2.7.2 に前方初期変位を変化させたときの重心の移動量を示す。この際、ヒト直立時の姿勢で は、下腿三頭筋は伸張された状態であるとした(Muraoka et al., Cells Tissues Organs, 171, pp.260-268, 2002)。前傾変位が小さい場合には後ろ側へ転倒、大きい場合には前側に転倒する。こ の解析により、姿勢制御の従来仮説に従うモデルでは、持続的な筋収縮筋を伴う筋の弾性による復 元力(弾性係数)のみでは立位姿勢を安定に維持することはできないことが明らかとなった。すな わち、下腿三頭筋の復元力と前方への重力転倒トルクの釣り合い姿勢を安定化するような下腿三頭 筋の高い剛性は実現できないことが示された。



 $\boxtimes$  2.7.2 Forward displacement of CoM during stationary standing simulation by passive muscle deformation

近年、静止立位時の姿勢ゆらぎにおいて、筋の間欠的な収縮が実験的に観察され(Loram et al., J. Physiol., 556.3, pp.683–689, 2004.)、さらに、Asai ら(PLoS ONE, 8(5), e62956, 2013)によって、 適切なタイミングで素早く筋を弛緩させることで不安定な立位姿勢が安定化されるとする新しい制 御仮説が提唱されている。そこで、図 2.7.3(a)に示すような、間欠的な神経入力を与えることによ って、立位姿勢の保持が可能であるか否かを検討した。

その結果、入力電流の有無により、筋線維内には図 2.7.3(b)に示すような筋小胞体からのカルシウ イオンの放出および吸収によるカルシウムイオン濃度の変化が生じ、その結果として収縮タンパク に筋収縮力が発生し、最終的に図 2.7.3(c)に示すように、前方転倒および収縮力による後方引き戻し の繰り返しによって、ゆらぎを伴う立位姿勢の安定化が可能であることが示された。このような立位 姿勢の安定化には、運動神経指令の間欠的な停止(図 2.7.3(a)に示した個々のパルス的入力の立下り タイミング)に対して、収縮タンパク質の十分素早い応答(多数のアクチン・ミオシン間クロスブリ ッジの協調的乖離)が重要な役割を果たしている可能性がある。



(a) Input current on a muscle fiber (b) Ca<sup>2+</sup> concentration in myofibrillar space (c) Displacement of CoM

🗵 2.7.3 Control of stationary standing by intermittent muscle contraction

#### C) おわりに

これまで開発した収縮タンパクのレベルから関節運動に至る力学的および生理学的に精緻なマルチ スケール筋骨格系運動シミュレータを用いて、ヒトの運動制御メカニズムおよび神経疾患に起因する 運動障害の発現メカニズムの解明を目指して、解剖学的に精緻な静止立位時の姿勢制御シミュレーシ ョンモデルの構築を行った。静止立位保持のシミュレーションの結果、従来の姿勢制御の仮説では姿 勢を保持することはできず、間欠制御仮説に基づく間欠的な筋収縮が姿勢の保持に有効であることを 示した。さらに、間欠制御による姿勢の安定化には、骨格筋の収縮タンパク質群が有する協同現象的 ダイナミクスが重要な役割を果たしている可能性が示唆された。

本開発により、ヒト静止立位姿勢制御のための筋骨格系の運動シミュレーションが可能となり、運動制御メカニズム解明に向けた筋・骨格シミュレーションのプラットフォームが完成した。今後は、さらに運動神経および感覚神経などの神経系とのカップリングにより、神経・筋・骨格系の統合シミュレータの開発を行い、ヒト静止立位の姿勢ゆらぎのシミュレーションから、身体運動の神経制御メカニズムの解明を目指していく予定である。

### (8) 発話・舌運動シミュレータの開発

A) はじめに

言葉を組み立てる語音の発音は高次脳機能の一つであり、我々人間が他の動物と比較して最も異な る機能の一つである。音が発生する部位の違いにより、語音は母音と子音とに分けられる。人間が複 雑な情報を効率よく伝達するためには、多種類の音素を発音する必要があるが、その多くは舌を複雑 に動かすことで肺からの呼気の流れを阻害し、音の元となる渦とその時間変化を生み出し、空気の疎 密波の発生につなげることで達成される。舌を複雑に動かす結果、口腔形状が様々に形成され共鳴周 波数を変化させることができる。このようにして、一定の特徴をもった音が生成され、聴覚を通して 認識されコミュニケーションの成立がなされると考えられる。しかし、一定の特徴がどのように形成 されるのかに関しての理解は十分ではない。

構音障害は様々に分類分けがなされるが、我々の研究において想定しているのは、器質性の構音障 害、すなわち解剖学的、機能的に障害が生じたために発生する構音の障害である。こうした構音障害 の解剖学的、機能的に障害と語音明瞭度との関係に関しては、構音障害者に関する臨床研究に進む前 に、健常人におけるベースラインの構築が必要となる。そこでは、聞き分け可能な語音を構音できる か否かの鑑別が問題となるが、個人差による口腔の解剖学的な違いや構音の違いの影響も問題となる。

今年度我々は、最も複雑とされる摩擦音の構音に関して健常人における解剖学的形状と構音機能の 個人差を同定し、それに起因する摩擦音の特徴への影響を調査した。

#### B) 構音機能の違い

構音機能を果たすために必要な舌運動の中で、摩擦音の構音時に特徴的な機敏な舌先端部の挙上に ついて、個人差を無視し機能面での特徴に着目した。その際、摩擦音構音時の口腔形状は5つの断面 積の異なる矩形管から構成し単純モデルとした。このモデルは、一人の被験者が摩擦音を構音した際 の口腔内形状を CT により計測し、冠状面での断面積を5箇所について算出した後に、その値を矩形 管断面積に適用し構築したもので、実際の音声と15 kHz までスペクトル包絡が一致することが確認 されている。この単純モデルの舌前方部に当たる部位を上下方向に可動するように改造を加えた。

構築した摩擦音単純モデルを用いて舌の挙上速度と声道流量の制御が歯茎摩擦音/s/の発音に与える 影響について調べた。実験は図 2.8.1(a)に示すように単純モデルの舌先端付近に可動性を加えた。可 動部にはラック・アンド・ピニオンが接続され、タイミングベルトでステッピングモータと結ばれて いる。ステッピングモータはデジタル制御装置(Aruduino UNO)により制御した。図 2.8.1(b)は実際 の実験環境を示しており、エアコンプレッサーが流量計を介して単純モデルに接続された。単純モデ ルは音響計測のため簡易半無響室内に設置された。

実験結果から、舌先端付近に可動性を与えた単純モデルにより、被験者の摩擦音を含む音節「うすい」と同様のスペクトル特性を示す音の再生が可能であることがわかった。ただし、そのためにはマスフローメータ(Series 4043, TSI)で流量を制御で被験者の呼気流量の変化を再現する必要があった(図 2.8.1(c))。

空力音の発生に関する Lighthill のアナロジーでは、レイノルズ応力テンソルの空間微分値の時間 変動が音源となる。前歯隙間後流の音源成分を実験で完全に計測することは不可能であるため、数値 流体シミュレーションを用いた。図 2.8.2(a)は舌先端付近の挙上により生まれた矩形管狭窄によるジ ェットと渦管である。同(b)は単純モデル矢状断面でのライトヒル音源を可視化したものである。これ により音源は上顎前歯舌側領域から生じ上下前歯間部を通過し下顎前歯唇側に分布することがわかっ た。



 $\boxtimes$  2.8.1 (a) Simplified vocal tract model of /s/. (b) Schematic of the experimental setups. (c) Flow rate and tongue height for the experiment.



🗵 2.8.2 Preliminary flow simulation<sup>5):</sup> (a) velocity magnitude and vortex tubes in the simplified model;

## (b) magnitude of Lighthill sound source.

舌前方部付近の上下運動および流量を変化させることにより単語発話時の摩擦音/s/の発生を再現し、 音と前歯部の流速変動を同時計測した結果を図 2.8.3 に示した。舌挙上時には、*Re* が約 1,200 で舌が /s/の位置に到達する前に音が発生し始め、/s/の位置に舌が到達した後も OASPL は上昇し続けた。一 方、舌下降時には *Re* が下っているにもかかわらず、高い OASPL を保ち、*Re* が約 1,000 になると急 激に OASPL が減少した。これらの結果は3つの舌速度で同様に観測された。OASPL と流速変動の 大きさには相関が見られ、舌の動きに対する乱流の音源発生タイミングのずれが、発話での舌の動き および音の制御に大きな影響を及ぼしていることが示唆された。



☑ 2.8.3 OASPLs of generated sound at each corresponding Reynolds number.

## C) 解剖学的な個人差

健常人における解剖学的形状の個人差を考慮し、それに起因する摩擦音の特徴への影響を調べた。 具体的には日本人話者3名の医療画像より抽出した/s/と/sh/の口腔形状に対して圧縮性流体のLESを 行うことにより、発音時の口腔内の流れ場および音場に関して被験者間の違いを調べた。

摩擦音/s/および/sh/発音時の口腔形状を3名の構音障害のない日本語男性話者の医療画像より抽出した。被験者A(42歳)の口腔形状はCT画像より抽出した。摩擦音/s/はコーンビームCT (CB MercuRay, 日立メディコテクノロジー)、/sh/は320-row Area Detector CT (Aquilion one, 東芝メディカル)により撮像し、口腔領域を含む被験者頭部の画像を取得した。この時、/s/は9.6秒間持続して発音している間に撮像し、/sh/は「うすいみそしる」と発音する際の1コマ(0.05秒間)として撮像した。

その他の二名の被験者 B および C(26 歳および 24 歳)は MR 装置(MAGNETOM Prisma fit 3T, Siemens)を用いて計測した。この時、被験者は装置内で約 30 秒間/s/および/sh/の発音状態を保ち、口腔形状の計測を行った。また、MR 装置では前歯形状を計測できないため、Takemoto らの手法を参考に、前歯に舌および口唇を押し当てることにより前歯形状を抽出し、声道形状から差し引くことで口腔形状を形成した。図 2.8.4 に MR 画像および抽出した前歯、声道形状を示す。



⊠ 2.8.4 Vocal tract, upper and lower teeth geometries extracted from magnetic resonance images. The MR images and vocal tract geometries are shown for subject B.



 $\boxtimes$  2.8.5 Computational grids of vocal tract walls of /s/ (a) and /ʃ/ (b) for subject A.

得られた口腔形状に対して、計算格子を構築することにより、空力音響シミュレーションを行った。 摩擦音/s/および/sh/発音時の被験者 A の口腔流路壁面の計算格子を図 2.8.5 に示す。計算コスト削減 のため、咽頭部(口唇から後方に約 64 mm)より後方の流路は計算領域から除いた。また、ジェット 流による乱流の渦と音源の発生を計算格子で解像するため、狭窄から口唇にかけて格子幅を細かくし た。

LES では、圧縮性流体の連続の式、運動方程式、エネルギー式、状態方程式

$$\frac{\partial\bar{\rho}}{\partial t} = -\frac{\partial\bar{\varphi}_j}{\partial x_j} \tag{2.8.1}$$

$$\frac{\partial \bar{\varphi}_i}{\partial t} + \frac{\partial \bar{\varphi}_i \tilde{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial \sigma_{ij}}{\partial x_j}$$
(2.8.2)

$$\frac{\partial \bar{\rho} \bar{E}}{\partial t} + \frac{\partial \bar{\varphi}_j \bar{E}}{\partial x_i} = -\frac{\partial \bar{p} \tilde{u}_j}{\partial x_i} + \frac{\partial \sigma_{ij} \tilde{u}_j}{\partial x_i} - \frac{\partial q_j}{\partial x_i}$$
(2.8.3)

$$\bar{p} = (\gamma - 1)\bar{\rho}\bar{e}, \qquad \bar{e} = C_{\rm v}\bar{T}$$
(2.8.4)

を支配方程式とし、離散的に解いた。ここで、tは時刻、 $\bar{\rho}$ は密度、  $\tilde{u}_i$ は $x_i$ 軸(i = 1, 2, 3)方向の速度、  $\bar{\varphi}_i = \bar{\rho}\overline{u}_i$ は流束、 $\bar{p}$ は圧力、 $\bar{E} = 1/2 |\tilde{u}_i|^2 + \bar{e}$ は全エネルギー、 $\bar{e}$ は内部エネルギー、 $\bar{T}$ は温度である。  $\sigma_{ij}$ および $q_j$ は粘性応力テンソルと熱流束であり、と計算する。

$$\sigma_{ij} = 2\bar{\rho}(\nu + \nu_{\text{SGS}}) \left( s_{ij} - \frac{1}{3} \delta_{ij} s_{ll} \right)$$

$$q_j = -\bar{\rho} \gamma C_{\nu} (\alpha + \alpha_{\text{SGS}}) \frac{\partial \bar{T}}{\partial x_j}$$
(2.8.6)

この時、動粘度 $\nu_{SGS}$ と熱拡散率 $\alpha_{SGS}$ はLESのサブグリッドスケール (Sub Grid Scale、 SGS) での渦 粘性と熱拡散を表しており、本研究では、Furebyの一方程式モデルを用いて決定した。 $s_{ij}$ は変形速度 テンソルであり、 $s_{ij} = 1/2 (\partial \tilde{u}_i / \partial x_j + \partial \tilde{u}_j / \partial x_i)$ と計算する。また、気体に関する定数である動粘度 $\nu$ 、 比熱 $C_{\nu}$ 、比熱比 $\gamma$ 、熱拡散率 $\alpha$ は、摂度20度の空気の値を用いた。さらに、空力音源の大きさについて、 Lighthillの音源

$$\varphi = \frac{\partial^2 \rho u_i u_j}{\partial x_i \partial x_j} \tag{2.8.5}$$

の時間変動のRMS値を計算した。

以上の式を有限体積法数値計算ソフトウェア OpenFOAM 2.3.1 を用いて口腔流路内の格子点上で 計算を行った。計算の境界条件としては、入口部に一様流速、出口部に無反射境界、口腔壁面に断熱 ノンスリップ条件を課した。入口流量は被験者 A に対して 400 cm<sup>3</sup>/s 、被験者 B、 C に対して 250 cm<sup>3</sup>/s と設定した。



☑ 2.8.6 Midsagital plane of the instantaneous velocity field in the vocal tract of /s/ for subject A
(a), B (b) and C (c).



 $\boxtimes$  2.8.7 Midsagital plane of the instantaneous velocity field in the vocal tract of /sh/ for subject A (a), B (b) and C (c).

シミュレーションにより得られた、被験者 A、B、C の歯茎摩擦音/s/発音時の口腔矢状断面における 流速比の瞬時値を図 2.8.6 に示す。全ての被験者において、舌先端と歯茎部によって形成された狭窄 流路からジェット流が発生し、上下前歯付近で大きく乱れて口唇の外へ出て行く様子が観察された。 この時、狭窄流路における最大平均流速は被験者 A、 B、 C でそれぞれ 50.1 m/s、 47.9 m/s、 45.9 m/s となった。また、上下前歯の隙間から出たジェット流は被験者 A、B において上唇面に沿って外 へ出て行ったのに対し、被験者 C では下唇面に沿って出て行った。

後部歯茎摩擦音/sh/発音時の口腔矢状断面における流速の瞬時値を図 2.8.7 に示す。全ての被験者の 口腔形状において、舌先端が歯茎摩擦音/s/にくらべて後方へ移動しており、狭窄流路の位置が後部へ 移っていることが確認できた。狭窄流路における最大平均流速は被験者 A、 B、 C でそれぞれ 33.4 m/s、 40.1 m/s、36.5 m/s となり、/s/の最大平均流速と比べて小さな値となった。これは、/sh/の狭窄 流路の横幅が/s/に比べて広くなったことが原因と考えられる。また、狭窄流路から出たジェット流は、 まず下前歯と舌先端の空間で大きく乱れた後、上下前歯の隙間を通って口唇へ流れて行った。上下前 歯の隙間から出たジェット流は、全ての被験者の口腔形状において、下唇面に沿って外へ出て行った。



 $\boxtimes$  2.8.8 Midsagital plane of RMS values of velocity fluctuation in the vocal tract of /s/ for subject A (a), B (b) and C (c).



 $\boxtimes$  2.8.9 Midsagital plane of RMS values of velocity fluctuation in the vocal tract of /sh/ for subject A (a), B (b) and C (c).

歯茎摩擦音/s/発音時の口腔矢状断面の流速時間変動に関する RMS 値を図 2.8.8 に示す。狭窄流路 から出たジェット流は、全ての口腔形状で、上下前歯の前後の空間において大きな流速変動が発生し た。後部歯茎摩擦音/sh/の口腔矢状断面の流速の RMS 値を図 2.8.9 に示す。後部歯茎摩擦音/sh/では 歯茎摩擦音/s/と比べ上下前歯の前後だけでなく、狭窄流路の出口付近でも大きな流速変動の値が見ら れた。これは、下前歯と舌先端による空間において旋回流が発生したことにより、狭窄流路から出た ジェット流がさらに乱れされたからだと考えられる。また、被験者Aでは狭窄流路の出口と上下前歯の流速変動の値はほとんど同じ値だったのに対し、被験者B、Cでは上下前歯の隙間に比べ狭窄流路 出口付近の流速変動の値の方が大きかった。

歯茎摩擦音/s/の各被験者の口腔内の流速分布より式(2.8.5)で計算した音源の大きさを図 2.8.10 に示 す。それぞれの口腔形状において、上下前歯の前後の空間において大きな音源値の分布が見られた。 この時、被験者 A に比べて、最大流速が小さいにもかかわらず、被験者 B、C の口腔内における音源 の大きさの方が大きかった。この原因としては、被験者 A の/s/の口腔内は下前歯や歯茎部に凹凸が少 なく、滑らかな流路となっているのに対し、被験者 B、C の口腔内には下前歯の先端や歯茎部の凹凸 が見られ、より大きな音源の発生につながったと考えられる。





 $\boxtimes$  2.8.10 Midsagital plane of Lighthill's sound source values of velocity fluctuation in the vocal tract of /s/ for subject A (a), B (b) and C (c).

 $\boxtimes$  2.8.11 Midsagital plane of Lighthill's sound source values of velocity fluctuation in the vocal tract of /s/ for subject A (a), B (b) and C (c).

後部歯茎摩擦音/sh/の各被験者の口腔内の音源の大きさを図 2.8.11 に示す。歯茎摩擦音/s/と同様に 上下前歯の前後の空間で音源が発生するとともに、狭窄流路の出口付近において大きな音源の値が見 られた。これらの結果より、大きな流速変動の値が観測された位置において、大きな音源の値が観測 されており、流速変動と空力音源の大きさに相関があることが示唆される。これまで行われていた口 腔内での熱線流速計による計測で観測された流速変動は、空力音源の発生を示していると考えられる。

各被験者の口腔形状に対して行ったシミュレーションにおける遠方場(口唇先端から 10 cm)の圧力 のスペクトルを図 2.8.12 に示す。被験者 A の/s/と/sh/の音の違いとしては、/s/において約 5.2 kHz 以 上のブロードバンドノイズとなったのに対し、/sh/は約 4.3 kHz にピークを持ち、高周波数域の音圧 レベルは/s/より/sh/が小さくなった。その一方、被験者 B の/s/は 5.5 kHz、/sh/は 4.3 kHz 以上のブロ ードバンドノイズとなり、被験者 C のとなり/s/は 5.5 kHz、/sh/は 5.1 kHz 以上のブロードバンドノ イズとなった。全ての被験者において、ピーク周波数は/sh/よりも/s/の方が高く、これまで報告されて きた一般的な/s/と/sh/の音響特性の違いと同様の傾向となった。このことから、それぞれの音素に関す る個人間の違いよりも/s/と/sh/間の違いがより優位であることが示唆された。



 $\boxtimes$  2.8.12 Spectra of sound generated by the vocal tract of /s/ and /sh/ for subject A (a), B (b) and C (c).

3名の被験者の歯茎摩擦音/s/および後部歯茎摩擦音/sh/の口腔形状に対して圧縮性流体のLESを行 うことにより、発音時の各口腔内における流れ場と音場を明らかにするとともに、被験者の違いにつ いて議論を行った。全ての被験者において、これまで報告されてきたように/s/と/sh/で狭窄流路の位置 が前後に異なり、狭窄流路からジェット流が発生することにより上下前歯の前後の空間および/sh/に 置いては舌と下前歯の間の空間において大きな流速変動と空力音源が見られた。細かな口腔形状の違 いは見られたものの、音源の分布位置や強さはほとんど変わらず、これまで報告されてきた被験者の 歯茎摩擦音の発生メカニズムが、日本人3人の被験者でも見られることが明らかとなった。また、そ れぞれの口腔形状から発生した音は、一般的な/s/と/sh/の音響特性の違いを示し、ピーク周波数の違い はほとんど見られなかった。

## D) まとめ

本年度の研究結果を踏まえると、構音障害の解剖学的、機能的に障害と語音明瞭度との関係に関す る研究を開始するために必要な、健常人におけるベースラインの構築が出来つつある。これにより、 聞き分け可能な語音を構音できるか否かの鑑別について、個人差と語音の音響学的特徴の違いを物理 的に踏まえた治療計画の立案が実現できると考えられる。すなわち、一般的な個人差の範疇に収まら ない解剖学的形状の違いや、構音機能の違いがどのようなものであるのか、を予測、推定するための 道筋が見えてきた。

【今年度計画と成果】

平成30年度業務計画は、

- 全脳循環代謝シミュレータの開発と高度個別化医療支援への応用 医用画像から取得できる主要脳動脈と脳静脈を接続する脳血管モデルを構築し、大規模ボクセル 型血流シミュレータを用いて、全脳レベルの血流動態を再現する。また、脳微小循環シミュレー タにより、脳毛細血管網内の赤血球の流動、分配、ガス交換の特性を調べ、全脳毛細血管に均一 に血液が供給されるメカニズムを明らかにしていく。さらに、得られたシミュレーション結果を 臨床で計測される MRI や CT 画像で表現し、脳血流データと同化させる手法の開発を進める。 これにより、脳循環系個別化医療支援に向けた基盤技術の確立を目指す。
- 2. 脳機能障害に関わる個別化医療支援のための大規模生体シミュレータ開発 脳機能障害が引き起こす循環障害、運動障害、発話障害の診断および治療を支援するための大規 模生体シミュレータの開発を進める。具体的には、ヒト運動機能解析のための神経筋骨格系統合 シミュレータ、構音機能分析のための流体音響連成シミュレータ、微小循環遊走細胞の動態解析 プラットフォームの開発を進める。また、心原性脳塞栓の発生リスク評価のための左心房内血流 シミュレータおよび脳動脈瘤コイル塞栓シミュレータを臨床データの後ろ向き評価に適用し、 個別化医療支援に対する有効性を検討する。

1. に関しては医用画像から取得できる主要脳動脈と脳静脈を接続する脳血管モデルを構築した。 格子ボルツマン法と適合サブドメイン法を用いて大規模な血流計算を安定して効率よく行う手法 を開発し、全脳レベルの血流シミュレーションを実行した。その結果、計画に対応する成果内容は、 ヒト大脳表層血管の実験観察との比較の結果、本モデルでは微小血管の生成の仕方や血管間隙の大 きさなど、実際の血管を定性的に良く再現できた。また、脳微小循環シミュレータにより、細い血 管では血流分配が全く促進されないのではなく、赤血球量が増加するほど、局所への集中による流 動抵抗の低下が顕著になり、必要な酸素が分配されることを示した。さらに得られたシミュレーシ ョン結果の表現方法については、脳形状データはMRI 3D-T1 画像として、脳血管データは 4D-CT の 画像として表現し、臨床現場での有効な医療支援ツールとなることを示した。

2. に関しては、脳機能障害が引き起こす循環障害、運動障害、発話障害の診断および治療を支援するための大規模生体シミュレータを開発し、ヒト運動機能解析については、前方転倒および収縮力による後方引き戻しの繰り返しによって、ゆらぎを伴う立位姿勢が安定することを示した。個別化医療支援に向けて、運動障害疾患の早期発見や早期治療につながる知見を得たため、姿勢制御のための筋骨格系の運動シミュレーションが可能となり、運動制御メカニズム解明に向けた筋-骨格シミュレーションのプラットフォームが完成した。

構音機能分析については、解剖学的・機能的に起因する構音障害と語音明瞭度との関係解明に必要な健常人における個人差を同定した。これにより聞き分け可能な語音を構音できるか否かの鑑別

について、個人差と語音の音響学的特徴の違いを物理的に踏まえた個別化治療計画の立案が可能と なった。

微小循環遊走細胞の動態解析については、炎症と非炎症状態の違い、および慢性炎症と炎症開始 期の違いを、深層マッチングを利用した画像解析から得られるマクロファージの動態により判別し た。脂肪組織における炎症の進行を自動的に判別支援できる個別化医療の可能性が示唆された。

心原性脳塞栓の発生リスク評価のための左心房内血流シミュレータの改善については、現在の患 者データと CPD による推定結果が良好に一致することが示され、原因を探る方法が前進した。

脳動脈瘤コイル塞栓シミュレータを改善した。即ち従来、医療現場ではデバイスのサイズや種類 の選択、留置手順が経験的に決定され、瘤内におけるコイルの偏在や逸脱、またステントの展開不 全といったリスクを常に抱えていたが、今回瘤内壁面とコイルの摩擦係数に関する有益な力学的知 見を引き出した。

これらの成果により個別化医療支援に向けて臨床データに基づいた解析(後ろ向き評価)ができ るように進捗させた。

# 3. 心臓シミュレーションと分子シミュレーションの融合による基礎医学と臨床医学の架橋 (サブ課題C)

【総括】

タンパク機能(遺伝子)の変化が最終的に臓器レベルにどのような変化を及ぼすかを物理的に解明・予 測することができれば、その応用は無限に広がる。ポスト「京」のパワーを活かし、本サブ課題では、世 界で初めて心臓シミュレーションと分子シミュレーションを融合させた真のマルチスケールシミュレー ションを実現する。これにより計算科学の歴史に新たなマイルストーンを築くと共に、現実の医療・創薬 にブレークスルーをもたらす。

- (1) 目標1:心臓病、特に「心不全」の根本的解明と最適治療を可能とするマルチスケール心臓シミュレータの開発を目的とする。モデル化は心不全研究において得られた遺伝子レベルの変化、発現調節、細胞内情報ネットワークに関する膨大な情報、臨床情報を効果的に取り入れて推進する。 平成30年度は、ミオシン首振りサイクルにおける全ヌクレオチド状態に対し統一した生物種の構造 (Myosin-VI)を鋳型にしたサルコメアモデルの高度化とリモデリングの検証を行った。また分子モデル・有限要素モデル間の連成手法を最終化した。さらに心不全治療と外科手術の慢性期予測法を完成した。
- (2) 目標2:目標1は分子レベルの力学と心臓のポンプ機能の関係に焦点を当てたものであるが、同様の 技術を致死性不整脈の病態解明と予防・治療に対して適用する。即ち、重点課題1において進められ るイオンチャンネルの分子シミュレーションと UT-Heart を融合することで、候補物質の分子構造 と遺伝子情報のみから不整脈のリスク評価を可能とするシステムの完成を目標とする。 平成30年度は、カリウムチャネルの一つであるhERGイオンチャネルと薬剤のドッキングシミュレ ーションを実施し、これと細胞パッチクランプ実験により求めたチャネル阻害率に基づく心臓興奮 伝播シミュレーションから評価した不整脈発生リスクを比較することにより、検証を行った。

【詳細】

以下では目標1、目標2に大別して実施内容を説明する。

(1) 目標1(心不全の解明と治療を目指すマルチスケール心臓シミュレーション)

心臓シミュレータ UT-Heart と分子シミュレータ CafeMol (京都大学・高田教授、理研・金田 TL)を 融合させることにより、ミクロ・マクロ間の相互作用により病態が進行する心不全の解明と最適治療 を可能とするマルチスケール心臓シミュレーションを実現することを目指す。これまでの UT-Heart では、サルコメアを構成するミオシン分子のダイナミクスは、アーム部がばねで表された 1 自由度 のミオシンヘッドが、ヌクレオチド状態に依存して確率的に首を振るモンテカルロシミュレーショ ンにより表現されている。このサルコメアモデルにより生理学的に妥当な心拍動、血行動態が再現さ れただけでなく、心拍動におけるミオシン分子の協調性の役割などにつき重要な知見が得られた[1]。 しかしミオシン分子を構成するアミノ酸残基の変異に起因すると言われる肥大型心筋症のような心 疾患のメカニズム解明に切り込むためには、より実際の分子構造に立ち入ったシミュレーションモ デルの導入が望まれる。



図 3.1.1 「京」で用いて来たサルコメアモデルとポスト「京」で開発を目指す新たな分子モデ

そこで図 3.1.1 に示されるように、ミオシンやアクチン分子を構成するアミノ酸一つ一つの自由度と タンパクの立体構造を考慮に入れた Ca・粗視化分子モデルに基づく CafeMol[2]を用いた分子動力学 シミュレーションと UT-Heart による心臓シミュレーションを連成させることにより、従来と次元 を異にする心疾患の解明に挑戦する。



図 3.1.2 ヒト心筋アクトミオシン系のクロスブリッジサイクル

アクチン・ミオシンの ATP 加水分解反応に伴う首振り運動のサイクル、即ちクロスブリッジサイ クルは次の様に起こると考えられている。まず始めにミオシンヘッドがアクチンに結合し、Pi を放 出する直前の「Pi-release 状態」を出発点とすると、Pi を放出することで第一段階の首振り運動、即 ちパワーストロークを生じ「ADP 状態」となる。続いて ADP ヌクレオチドを放出することで第二 段階目のパワーストロークが生じる。そしてアクチンとミオシンが最も強く結合する「Rigor 状態」 となる。更に ATP が結合することで、ミオシンがアクチンから解離をして、レバーアームのリカバ リーストロークが生じる。更にヘッド内で ATP 加水分解反応が生じて「ADP\*Pi 状態」となり、最 後に元の状態 Pi-release 状態に戻る。

以上のクロスブリッジサイクルを粗視化シミュレーションで実現する為には、各ヌクレオチド状態におけるアクトミオシンの構造情報が必須となる。しかしヒト心筋のアクトミオシンの構造情報 は不足しているため以下のようにホモロジーモデリングによりヒト心筋アクトミオシンモデルを構築した。



図 3.1.3 ヒト心筋アクトミオシン系のホモロジーモデリングのための新旧鋳型構造の比較

ホモロジーモデリングにおける具体的な鋳型構造として、昨年度までの旧モデルにおいては、クロ スブリッジサイクルにおける各化学状態において図 3.1.3 のような鋳型構造を PDB データベースか ら抽出・選択した。それにより in-vitro 実験[3]とも定性的に一致するパワーストローク時の発生力 を得る事に成功したが、このモデルではヌクレオチド毎に異なる生物種の鋳型構造を使用している ため状態間で力学的、構造的な整合性が保たれていない心配があった。そこで今年度は、クロスブリ ッジの全状態において、同一の生物種、同じミオシンファミリーの鋳型構造を適用することで力学的 整合性を担保する。具体的には Myosin-VI の生物種: Sus scrofa (猪豚)の構造を鋳型にした。各々 の化学状態で適用した鋳型構造の PDB-ID は、Pre-Powerstroke 状態 (4ANJ)、Pi-release 状態 (4PFO)、ADP 状態(6BNW)、Rigor 状態(6BNV)である。

- 1. 人間のアミノ酸配列と鋳型構造の配列をシーケンスアライメント[MOE]
- 2. シーケンスアライメントを基に鋳型構造から人間モデルを構築 [Modeller(Saliet al., 1993)]

人間ミオシン(重鎖S1)と鋳型(Sus scrofa, moyosin-VI)のシーケンスアライメント結果

 Human
 MGDSEMAVFGAAAPYLRKSEKERLEAQTRPI
 DLKKDVFVPDDKQEFVKAKIVSREGGKVTAETEYGKTVTVKEDQVMQDNPVFDKIEDMAMLTFLHEPA

 Human
 VLYNLKDRYGSWMIYTSGLFCVTVNPVKWLPVYTPVVAAYRGKKRSEAPPHIFSISDNAYQYMLTDRENOSILITGESGAGKTVNTKRVIQYFAKI

 Human
 VLYNLKDRYGSWMIYTYSGLFCVTVNPVKWLPVYTPVVAAYRGKKRSEAPPHIFSISDNAYQYMLTDRENOSILITGESGAGKTVNTKRVIQYFALA

 Human
 VLYNLKDRYGSWMIYTYSGLFCVTVNPYKWLPVYTPVVAAYRGKKRSEAPPHIFSISDNAYQYMLTDRENOSILITGESGAGKTVNTKRVIQYFALA

 Human
 IGKSKKDQSPGK)

 IGKSKKDQSPGK)
 TLEDDIIQANPALEAFGNAKTVRNDNSSRFGKFVEIHFNEKSSVVGGSVSHYLLEKSRVIFOLKAERDYHIFYQLSAGASED IRER

 Human
 LLITNNPYDYAFISQGETTVASIDDAEELMATDNAFDVLGFTSEEKNSWYKLTGAIMHFGNMKFKLKOREEQAEPDGTEEADKSAYLMGLNSADLLKGLC

 Human
 LLITNNPYDYAFISQGETTVASIDDAEELMATDNAFDVLGFTSEEKNSWYKLTGAIMHFGNMKFKLKOREEQAEPDGTEEADKSAYLMGLNSADLLKGLC

 Human
 LKITNNPYQYAFISQGETTVASIDDAEELMATDNAFDVLGFTSEEKNSWYKLTGAIMHFGNMKFKLKOREEQAEPDGTEEADKSAYLMGLNSADLLKGLC

 HUman
 HKKCGLEVTFTIDFGNDLQACIDLEKPMENWYTRINATLETKOPROYFIGVLDIAGFEIFENSFEGCINFCNEKLQGFFNRHMFYLGEGE

 -TRVMLTTAGGAKGTKVEQANNARDALAKTYYSMLFDHVWNOCFPF-ETSSYFIGVLDIAGFEYFEHNSFEGCINFCNEKLQGFFNREHNMVTRINATLETKOPROYFIGVLDIAGFEYFEHNSFEGCINFCNELQGFFNRENLKLOGFNRENKLAGAVCYETTOFVEKNDDA

 Human
 HKKEGLEVTFTIDFGNDLQACIDLIEKPMGINSILEEECMFFKATDMTFKKAKLFDNHLGKSNFGMPRNIKGKRCHALGNIDVYNIGVDGFFNERTLKEGGG

 Human
 HKKEGLEVTYVCQUNAVAGADALKYVYNMVNUTTNATLETKOPGYFIGVLDIAGFEYFEH



図 3.1.4 ヒト心筋アクトミオシン系のホモロジーモデリングのための新旧鋳型構造の比較

具体的に鋳型構造からヒト心筋のアクトミオシンモデルを構築するには次の二つの手順で行う。 (1) まず、ヒト心筋アミノ酸配列と鋳型である Sus scrofa (猪豚)のアミノ酸配列をソフトウェア MOE でシーケンスアライメントする。(2) 続いて、シーケンスアライメント(マッピング情報)を基 に、Modeller(Sali et al., 1993)を用いて鋳型構造からヒト心筋の構造を構築する。まず、(1)のシー ケンスアライメントの結果を図 3.1.4 に示す。これは、ミオシンの重鎖 S1 と鋳型である Myosin-VI、 Sus scrofa のアミノ酸配列のマッピングの結果である。各行の上がヒトで、下が Sus scrofa のアミ ノ酸配列を示す。なお、ヒト心筋のミオシンファミリー:Myosin-II と異なるミオシンファミリ ー:Myosin-VI を鋳型に使用しているので、アライメントの前にヒト心筋のミオシン-II には存在しな いインサーション領域を取り除いてアライメントした。



図 3.1.5 Modeller のスコア分布とスコアがベストの Pi-release 構造

このマッピング情報を基に model 構造を構築した。その際、各ヌクレオチド状態に対して Modeller で各々300 サンプル構築し、その中で最もスコアの良い構造を抽出した。図 3.1.5 左に Modeller の スコアの分布を示す。一例として、Pi-release 状態におけるベストの構造を右に示す。但し、鋳型構 造においては、ヌクレオチド状態によらずレバーアーム部分と N 末部分が構造化されていないとい う問題がある。



図 3.1.6 Myosin-Ⅱ構造の Leverarm と軽鎖を援用して合成した Pi-release 構造モデル

そこでレバーアーム部分については Myosin-II のチキン由来のレバーアーム構造を活用する方針 を取った。まず、Myosin-II(chicken)の Rigor 状態における X 線結晶構造(PDB-ID: 2MYS)を鋳型に して、MOE と Modeller によりヒト心筋ミオシンの複合体構造(レバーアームと2本の軽鎖)を構築 した(図 3.1.6 の赤と青の部分)。そうして作成した Myosin-II チキン由来のヒト心筋ミオシンの Converter 領域(709-732 & 736-768)を、Rigor 状態と ADP 状態の Myosin-VI 由来のヒト心筋モデ ル構造の対応領域に rigid-body フィットして chiken-myosin-II 由来のレバーアーム(resid: 769-)と 二つの軽鎖に挿げ替えた。Pi-release 状態と Pre-powerstroke 状態のレバーアーム部構造のモデリ ングの際は、上記で作成した新しい Rigor 状態のヒト心筋モデル構造を活用した。即ち、Pi-release 状態の場合は残基番号: 704-710 の領域に、Pre-powerstroke 状態の場合は残基番号: 707-713 の領域 に新しい Rigor 構造の対応する領域を rigid-body fitting して残基番号:709-以降の新しい Rigor 状態 のモデル構造ですげ替えることでレバーアーム部のモデリングを行った。一方、N 末の非構造化部 については、ADP、Rigor、Pre-powerstroke、Pi-release 各状態での Central-body の N 末側の領域 98-110 に 2MYS Chiken-Rigor 由来のヒト心筋モデルの対応する領域を rigid-body fitting して、 Chiken 由来のヒトモデルの N 末(1-97)領域で挿げ替えることでモデリングした。但し、その際、 ADP 状態においては、構造化した重鎖のN末と軽鎖の一部が絡まってしまう問題を生じた。そこで 軽鎖の残基番号:1147-1154 に関わる斥力相互作用は全て不活性にし、1147-1154 に関わる Gocontact 相互作用を削除した上で、軽鎖: resid 1147-1154 のみを動かしたショート MD を遂行し構 造的な絡みを取り除いた。



図 3.1.7 Myosin-II構造の Leverarm と軽鎖を援用して合成した Pi-release 構造モデル

ー方、アクトミオシン系におけるアクチン分子については後述のように計算負荷軽減のためにカ ーテンレールモデルに置き換えるが、ミオシン分子との相対的位置関係は重要である。そこで本研究 ではアクチン・ミオシン複合体構造も以下のプロトコルでモデリングした。先ず、Rigor 状態の複合 体モデルについては、PDB: 6bnv 構造におけるミオシン chain-K の insertion1 領域(278-303)を除 く領域に、上記で構築した新しい Rigor 状態におけるミオシン単体モデルの 40-708 領域を rigidbody fitting することで複合体構造を形成した。続いて ADP 状態の複合体モデルについては、 PDB:6bnw 構造におけるミオシン-chain-K の insertion 1 領域(278-303)を除く領域に、上記で構築 した新しい ADP 状態におけるミオシン-chain-K の insertion 1 領域(278-303)を除く領域に、上記で構築 した新しい ADP 状態におけるミオシン-bite である。また Pi-release 状態におけるアクチン・ミオシンの複合体モデル構造は存在 しないので、次の手順でアクトミオシンの複合体を構築した。即ち、上記で構築した Rigor 状態にお けるアクトミオシン複合体モデルにおけるミオシンの Lower-50K 領域 (463-596, 648-663, 686-708)に Pi-release 状態における新しいミオシン単体モデルの対応する領域を rigid-body fitting して アクトミオシン複合体構造を形成した(Nature, 2016, J Ecken et al. 論文の記述を参考に Lower 50K に fitting する方針を取った)。なお、Pre-power stroke 状態はアクチンと解離している状態な ので、複合体構造を形成することは必要ない。



全原子模型の力場と全原子MDの揺らぎに基づきアミノ酸種/相互作用種毎に決定

CafeMolにおいてはエネルギーランドスケープ理論に基づいた構造依拠型のポテンシャルが適用 される。その基盤となる AICG2 モデル[4][5]を図 3.1.8 に示す。このモデルでは、アミノ酸配列に 沿ったローカル相互作用と、配列的に離れた残基間に働く non-local 相互作用からなる。ローカル 相互作用は、ボンド長、シュードボンドアングル項、二面角相互作用からなる。式中の添え字 0 は、参照構造(モデル構造)におけるボンド長や、二面角であることを示す。各々の関数形は二次 関数の形状をしており、ボンド長や、二面角がモデル構造に近付くほどエネルギー的に安定になる ことが分かる。一方、non-local 相互作用は Go-conatet 相互作用と排除体積項からなる。タンパク 分子は天然構造において図中の概念図の様に折れ畳まっているが、特にこの Go-contact 相互作用 は、空間的に互いに近くにある残基間にのみ働き、モデル構造を安定に保持する作用を持ってい る。このモデルにおいて、相互作用強度を表すパラメータ K, ε は、全原子力場や全原子シミュレー ションに基づき、アミノ酸種、相互作用種毎に決定される。



図 3.1.9 新たに構築したヒトミオシンモデルによるクロスブリッジサイクル

CafeMolによりクロスブリッジ運動のフィージビリティスタディを行った結果を図 3.1.9 に示 す。先ず、Pi-release から出発して、各々の状態で一定シミュレーションステップ後に CafeMol の 粗視化ポテンシャルをスイッチすることで、模擬的に ATP 加水分解サイクルを再現した。なお、 計算負荷軽減のためアクチンは後の図 3.1.12 右に黄色線で示すカーテンレールで置き換えた。また 同図に示されるように、レバーアーム C 末側に Light meromyosin 相当のバネ(モーメントも伝達 できるよう実質 7本に分散)を取り付けた。詳細は昨年度報告書に記載した通りである。Pi-

図 3.1.8 CafeMol が基盤とする Atomic Interaction-based Course Grain (AICG2) model

release 状態から出発して Pi 放出、ADP 放出に伴い首振り運動を実現し、次いで ATP が結合する ことでレールから解離して再び Pi-release に戻り、アクチンと結合する様子が再現出来ていること が分かる。更に定量的には次の図 3.1.10 のようになる。パワーストロークに伴うレバーアームの重 心変位:80~100Åは実験で知られるストローク値と一致している。



図 3.1.10 クロスブリッジサイクルにおける Leverarm とヘッド重心の運動

次に以上の分子モデルとマクロ有限要素との連成について説明する。



図 3.1.11 基本的状態遷移モデル

先ず図 3.1.11 に基本のミオシン分子状態遷移モデルを示す。Pi-release には Weak bind と Strong bind の 2 状態を設けた。収縮力の発生は ATP 状態から Pi-reealse(weak bind)への遷移速度k<sub>+1</sub>を時 間依存として制御する。ここで収縮力は下段にある 3 状態(strong bind)でのバネの伸びから決まる ものとする。Pi-release 状態(strong bind) と ADP 状態間の遷移速度はレバーアームのスウィング 距離 LAswingに依存して決まるものとする。ここでLAswingはレバーアームのヘッド結合部に対す る Z 方向(フィラメント方向)の相対座標から定める。ADP 状態から Rigor 状態への遷移は LAswing が 50Åを上回ったもののみ許されるもととする。また Rigor 状態から ATP 状態への遷移速度を Front door の開閉度に応じて定めたモデルも導入する。既述のように CafeMol は構造依拠型のポテ ンシャルに基づくため、化学状態の変化はポテンシャルの切り替えによって表さざるを得ないが、こ のように切り替えの遷移速度を、刻々の分子構造状態を特徴付ける変数の関数として与えるモンテ カルロ法によって、分子構造を反映したクロスブリッジサイクルのシミュレーションが可能となる。



図 3.1.12 マクロ(有限要素リング)とミクロ(ミオシン分子モデル)のカップリングの概念

図 3.1.12 にマクロモデル、即ち心筋を模擬した有限要素リングとミクロモデル、即ちミオシン分 子モデルのカップリングの概念を示す。リングモデルの有限要素解析から連続体としての変位、歪、 応力が算出されるが、線維方向の歪からストレッチ(伸び率) λを求めることが出来る。ある一つの ミオシン分子に着目して、ミオシンがアクチン(黄色の線で表されるカーテンレール)にバインドし た後のストレッチの増分をΔλとすると、その値からレバーアーム C 末と緑色太線で表されるミオシ ンフィラメント間に取り付けられたバネの伸び-Δzを知ることが出来る。ここで*SL*<sub>0</sub>は無負荷状態に おけるサルコメア長である。一方、ミオシンのパワーストロークによりレバーアームは回転するが、 そのバネの伸びへの寄与分を*d<sub>PS</sub>*と表すと、-Δzとの和がバネの合計の伸びになる。このばねの伸び により生じる力が線維方向の収縮力としてマクロの有限要素にフィードバックされる。しかし、一般 に有限要素解析の時間刻みと分子シミュレーションの時間刻みが異なるうえ、上記のようにストレ ッチと収縮力を単純に交換するだけでは連成解析は成功しない。以下に、開発した連成計算法の詳細 を記す。



図 3.1.13 ミクロとマクロの多重時間ステップカップリング法

先ず、新たに開発した分子モデルと有限要素モデルの多重時間ステップカップリング法につき説 明する。図 3.1.13 に示すように、マクロ有限要素モデルの時間刻みを分子モデルの時間刻みよりも 大きくとることによって計算を切り替える際のオーバーヘッドおよびマクロ計算時の計算機の遊び 時間の比率を短縮し、並列効率を上げる対策を講じた。マクロ有限要素モデルの時間刻みを $\Delta T$ とし て、分子モデルの時間刻みはそれをn分割し $\Delta t = \Delta T/n$ により与えるものとする.時刻Tでのマクロ計 算が終了した時点でマクロ有限要素から分子モデルへのフィードバックは、各時刻 $t_k = T + k\Delta T/n$ , (k = 1, ..., n)におけるサルコメア長変化を以下のストレッチを元に定めることにより与える。

$$\tilde{\lambda}_k = \frac{k}{n} \left( {^T}\lambda - {^{T-\Delta T}}\lambda \right) + {^{T-\Delta T}}\lambda$$

ここで<sup>T</sup> $\lambda$ は時刻Tでの有限要素の線維方向のストレッチで有限要素の変位勾配テンソルFと線維方向 単位ベクトルfから <sup>T</sup> $\lambda$  = ||Ff||により定められる。上式により分子モデルへのフィードバックが時刻 TからT +  $\Delta$ Tにかけて段階的に少しずつ前回の(時刻T –  $\Delta$ TからTにかけての)分子モデル計算時のフ ィードバック最終値<sup>T- $\Delta$ T</sup> $\lambda$ から最新のストレッチ<sup>T</sup> $\lambda$ に修正され、 このストレッチをもとに分子モデ ルのバネの付け根の結合後の収縮方向のすべり距離を各時刻t = T + k $\Delta$ T/nにおいて以下で与える。

$$d_k = -\frac{SL_0}{2} \left( \tilde{\lambda}_k - \tilde{\lambda}_{\text{attach}} \right)$$

ここで $\tilde{\lambda}_{attach}$ はミオシン分子が結合した時点でのストレッチ、 $SL_0$ は既述のように無負荷状態におけるサルコメア長である。

マクロ有限要素を駆動する線維方向の収縮力を時刻 $T + \Delta T$ で与える際は,結合している分子の時間幅[ $T, T + \Delta T$ ]における力積の総和が両スケール間で整合するように与える。その際、結合している各分子の時刻 $t_k = T + k\Delta T/n$ での寄与は以下の式から定める。

$$F_k = \tilde{F}_k + S_k \frac{k\Delta T}{n} T^{+\Delta T} \dot{\lambda}$$

ここで、 $\tilde{F}_k$ は分子シミュレーション中の時刻 $t_k$ におけるバネの収縮力、 $S_k$ はその剛性である。右辺 第2項は時刻 $T + \Delta T$ でのストレッチレートに依存する修正量である。この第2項によりマクロ有限 要素変数(変位、速度、加速度)の時刻Tから $T + \Delta T$ への更新においてクロスブリッジによる剛性が適 切に取り込まれ安定した計算が可能になる。ただしマクロの更新は以下の運動方程式を Newmark- $\beta$ 法により時間積分することにより求める。

$$\begin{cases} \int_{\Omega} \boldsymbol{\delta} \boldsymbol{\dot{u}} \cdot \rho \boldsymbol{\ddot{u}} \ d\Omega + \int_{\Omega} \boldsymbol{\delta} \boldsymbol{\dot{z}} : (\boldsymbol{\Pi} + 2pJ\boldsymbol{F}^{-1})^T \ d\Omega = P \quad \int_{\Gamma} \ \boldsymbol{\delta} \boldsymbol{\dot{u}} \cdot \boldsymbol{n} \ d\Gamma \\ \int_{\Omega} \delta p \left( 2(J-1) - \frac{p}{\kappa} \right) \ d\Omega = 0 \end{cases}$$

ここで、 $u = {}^{T}u(X) = {}^{T}x(X) - X$ は物質点 $X \in \Omega$ の変位ベクトル、 $\rho$ は心筋の密度、 $F = \partial x / \partial X$ は変 形勾配テンソル、 $Z = \partial u / \partial X$ は変位勾配テンソル、 $J = \det F$ はヤコビアン、pは心筋内圧、 $\kappa$ は体 積剛性係数、Pは内壁 $\Gamma$ にかかる血圧である。 $\Pi$ は第 1Piola-Kirchhoff応力テンソルであり、以下 のようにアクティブ、パッシブ、粘性応力により構成される。

$$\boldsymbol{\Pi} = \boldsymbol{\Pi}_{\rm act} + \boldsymbol{\Pi}_{\rm pas} + \boldsymbol{\Pi}_{\rm vis}$$

ここで時刻T+ΔTにおけるアクティブ応力IIactは以下の式から与えられる。

$$\boldsymbol{\Pi}_{\text{act}} = \frac{T, \Delta T}{T + \Delta T} \boldsymbol{\lambda} \boldsymbol{f} \otimes \boldsymbol{f} \cdot \left(\frac{\partial^{T + \Delta T} \boldsymbol{x}}{\partial \boldsymbol{X}}\right)^{T}$$

ここで要素に与える線維方向fの張力 $^{T,\Delta T}F_f$ は、その力積 $^{T,\Delta T}F_f \cdot \Delta T$ が先の分子の収縮力 $F_k$ の $[T,T + \Delta T]$ 間の力積の要素内分子の総和と整合するよう以下により定める。

$$T_{,\Delta T}T_f = \frac{1}{SA_0} \frac{n_{M0}}{n_M} \sum_{i=1}^{n_M} \frac{1}{n} \sum_{k=1}^n F_{k,i}$$

ここで $SA_0 = 1,000$ nm<sup>2</sup>は筋原線維内の1アクチンフィラメントあたりの断面積、iは要素内に含まれるミオシン分子の番号、 $n_{M0} = 80$ は1アクチンフィラメントに結合できるミオシン分子数、 $n_M$ は実

際に計算で用いたミオシン分子数である。先に定義した $F_{k,i}$ 内の $^{T+\Delta T}$ えを含む第2項により、時刻 $T+\Delta T$ における収縮力が時刻 $T+\Delta T$ におけるストレッチと結合分子の剛性から適切に決まるようになっているため安定な解法が実現される。



図 3.1.14 連成計算におけるミクロ剛性の導入の効果

図 3.1.14 は単一の有限要素を収縮させるシミュレーションにおいて、要素の収縮率の時間変化を表 したものである。Explicit は上式において剛性 $S_k$ をゼロと仮定して、マクロへは力 $\tilde{F}_k$ のみを渡して計 算を進めた場合である。Explicit の場合、クロスブリッジの数が増え剛性が $\Delta T$ から決まる許容値を 超えた時点で不安定となっているが(青線)、Implicit ではこのような不安定性は剛性の考慮により 回避され、しかも分子モデルの細かい時間刻みに合わせて通信した場合(黒線)と同様の結果が得ら れていることがわかる。

パッシブ応力**Π**<sub>pas</sub>は以下のように連続体の歪エネルギー関数から応力が定まる異方性超弾性体を 受動的構成側として用いた。

$$W_{\text{pas}} = c_1(\tilde{l}_1 - 3) + c_u \frac{\exp(Q_u) - 1}{2}$$

ここで、  $\tilde{I}_1$  は左Cauchy-Green変形テンソル $C = F^T F$ から定められる第一不変量

$$\tilde{I}_1 = \det(\boldsymbol{C})^{-\frac{1}{3}} \mathrm{Tr}(\boldsymbol{C})$$

 $Q_u$  はGreen-Lagrange歪テンソルE = (C - I)/2から定められる以下の2次形式である。

$$Q_u = b_{ff}E_{ff}^2 + b_{ss}E_{ss}^2 + b_{nn}E_{nn}^2 + 2b_{fs}E_{fs}^2 + 2b_{fn}E_{fn}^2 + 2b_{sn}E_{sn}^2$$

ここでそれぞれの成分は以下のように線維f,シートs,シートノーマルn方向での成分表示に基づくも
のである。

$$\begin{cases} E_{ff} = \mathbf{E} : \mathbf{f} \otimes \mathbf{f}, E_{ss} = \mathbf{E} : \mathbf{s} \otimes \mathbf{s}, E_{nn} = \mathbf{E} : \mathbf{n} \otimes \mathbf{n} \\ E_{fs} = \mathbf{E} : \mathbf{f} \otimes \mathbf{s}, E_{fn} = \mathbf{E} : \mathbf{f} \otimes \mathbf{n}, E_{sn} = \mathbf{E} : \mathbf{s} \otimes \mathbf{n} \end{cases}$$

通常の心臓シミュレーションと同様のパラメータ( $c_1 = 71.8$ Pa,  $c_u = 600$ Pa,  $b_{ff} = 5$ ,  $b_{ss} = 6$ ,  $b_{nn} = 3$ ,  $b_{fs} = 10$ ,  $b_{fn} = 2$ ,  $b_{sn} = 2$ )を採用した。以下に導入するリングモデルではfに周方向、sに軸方向、nに半径方向を割り当てた。粘性応力 $\Pi_{vis}$ はニュートン粘性を仮定し、以下のように与えた( $\mu_s = 100$ Pa·s)。



図 3.1.15 心臓を簡易的に模擬した 8 有限要素リングモデル

連成解析のためのマクロモデルとしては図 3.1.15 に示す 8 要素からなる有限要素リングモデルを 用いた。心室の境界条件を簡易的に模擬するため、各節点の高さ方向にあたる Z 座標は固定し、リ ング内腔の容積変化と圧力を変数として、後負荷を表す windkessel モデルと連成させた。リングの 形状は人心臓と同等の容積となるように、無負荷状態でのリング内壁の半径を 3cm、壁厚を 1.5cm、 高さを 5cm に設定した。後負荷に関しては、キャパシタンスを3ml・mmHg<sup>-1</sup>、血管抵抗を心臓に近 い方から0.075mmHg・s・ml<sup>-1</sup>、1.2mmHg・s・ml<sup>-1</sup>と設定した。一有限要素には半サルコメアに対 して 16 個のミオシン分子モデルが、リング全体では 128 個のミオシン分子モデルが埋め込まれてお り、その運動の様子を拡大してリングモデルの収縮に重ねて可視化できるようにした。ピンクとグレ ーの線がアクチンフィラメントとミオシンフィラメントを表す。



図 3.1.16 基本的状態遷移モデルでの圧容積時間変化と心筋張力および形状変化の様子

図 3.1.16 に分子モデルとして基本的状態遷移モデルを採用した場合の連成シミュレーションの結果 を示す。収縮制御は図 3.1.11 に示すように ATP 状態から Pi-release 状態(弱結合)への遷移速度定数 を時間の関数として与えることにより行っている。分子モデル計算の1時間ステップが実時間幅  $\Delta t = 2.5ns$ に相当すると仮定した。図 3.1.2 の検討をもとにマクロ有限要素の時間刻み幅は $\Delta T =$ 1,000 $\Delta t = 2.5\mu s$ とした。したがって、0.5s までの計算に要した分子モデルの総ステップ数は 0.5/2.5×10<sup>-9</sup> = 2×10<sup>8</sup>回である。1分子に1コアを割り当てて 0.5 秒の計算に約 4 日の計算時間 を要した。



図 3.1.17 ミオシン分子フロントドア



図 3.1.18 フロントドア距離と収縮力の関係(rigor 状態)
 左:時間緩和なし、中央: τ = 0.1ms、右: τ = 2.5ms

次に Rigor からの解離速度  $k_{+5}$ が時々刻々のフロントドアの距離<sup>t</sup>dに応じて決まる分子遷移モデ ルを検討した。ここでフロントドアは図 3.1.17 に示されるように Upper 50K と Central Body に属 する 2 残基 (ResID 132、322) によって定義され、この残基間距離<sup>t</sup>dが広がることで ATP がミオ シン分子内に侵入し結合できると考える。図 3.1.18 左は $k_{+5} = 200[1/s]$ とした基本モデルにおいて リングモデルに埋め込まれた 128 個のミオシンの Rigor 状態におけるフロントドア距離とレバーア ームに接続したバネの収縮力との相関をプロットしたものである。 ここで距離<sup>t</sup>dとの相関に加え、 以下のように緩和時定数 $\tau = 0.1ms$ または $\tau = 2.5ms$ で緩和した距離<sup>t</sup> $\tilde{d}$ との相関もプロットした(中央、 右)。

$${}^{t}\tilde{d} = \frac{1}{\tau} \int_{-\infty}^{t} \exp\left(-\frac{t-s}{\tau}\right) {}^{s}d ds$$

緩和時定数 $\tau$ が大きくなるほど、収縮力と距離 $t\tilde{a}$ の相関が強くなり、距離が大きくなれば解離速度も 大きくなると仮定すると、マイナスの収縮力を発生しているミオシンがより早期に解離することで 効率アップにつながることが期待される。



図 3.1.19 基本モデルでのシミュレーションにおける 10番目のミオシンの状態遷移とフロントドア距離

図 3.1.19 上段は基本モデルでのシミュレーションにおける 10 番目のミオシンの状態遷移および 時間緩和を変えた場合のフロントドア距離の推移を表したものである。下段の赤線は、特にτ = 2.5ms としたとき収縮力と距離<sup>t</sup>dの rigor 状態での軌跡を表している。2 回目のサイクルで見られるように 収縮中の要素ではパワーストロークにより生成されたバネのプラスの歪がフィラメント間すべり運 動に伴って減少し、やがてはマイナスになって要素の収縮運動の妨げとなる。逆に 4、5、6 回目の サイクルでは、パワーストロークで得たバネの歪エネルギーを使い切る前に解離しており、これも効 率的に好ましくない。図 3.1.20 は顕著なマイナスの歪の例であり、下段の図に示すようにもし緩和 されたフロントドア距離が 29Åあたりを境に大きくなるのであれば、収縮運動の効率がアップする ことが期待できる。



図 3.1.20 基本モデルでのシミュレーションにおける8番目のミオシンの状態遷移とフロントドア距離



図 3.1.21 緩和時定数 τ = 2.5ms、緩和距離閾値 29 Åの場合の性能比較

図 3.1.21 は、 $\tau = 2.5ms$ として解離速度を緩和距離が 29Å以下であれば基本モデルの 50%に、 29Å 以上であれば 200%または 400%に設定した場合の容積変化と ATP 消費を比較したものである。特 に 200%に設定した場合は基本モデルに比べて ATP 消費が増えることなく拍出量が増加しているこ とがわかる。



図 3.1.22 緩和時定数 T = 0.1ms、緩和距離閾値 29 Åの場合の性能比較

一方、図 3.1.22 は、*τ* = 0.1*ms*として同様の計算をした例であるが、どの設定においても ATP 消費 は削減されているものの拍出量はほとんど変化していない。 図 3.1.18 を見ても、この短い緩和時定 数では緩和距離を用いた効率アップが難しそうなことがわかる。



図 3.1.23 Rigor および ATP 毛構造における結合距離、結合角、二面角の比較

これまでの結果は状態遷移をシミュレートする際に各状態の形状を元に定めた状態固 有のポテンシャルに全面的に切り替えることにより得られた結果である。しかし、実際の 物理現象においてヌクレオチド状態の変化がこのような全領域に渡るポテンシャル変化 を引き起こしているとは考えづらく、実際には限定された箇所での局部的な変化が大き な形状変形を引き起こしているのではないかと考えられる。実際に、CafemolのAICG2型 ポテンシャルを採用し、そのときの結合距離(bond length:隣り合う残基間の距離)、結合 角(aicg13:二つ先の残基間距離)、二面角(aicg14:三つ先の残基間距離)をRigor構造(黒)と ATP構造(赤)で比較してみると図3.1.23に見るようにほとんどの領域で一致している。 aicg13, aicg14ポテンシャルが適用されていない領域(青線が底辺)を除いて違いが比較的 大きな部分を抽出すると点線で囲ったところのみになる。これらの領域の位置を示した のが図3.1.24である(図右がRigor形状)。ここで緑はaicg13およびaicg14が異なる領域で、 オレンジはbond lengthが異なる領域である。



図 3.1.24 Rigor 形状(右)と点線内の赤線内ポテンシャル切り替え後の snapshot

そこでRigor形状から出発し、これらの領域のみRigor状態のものからATP状態におけ るポテンシャルに切り替えれば、レバーアームの巻き戻しの傾向が形状変化に現れるこ とが期待できる(図3.1.24左)。図3.1.25はRigor状態から出発し、局部的にポテンシャルを ATP状態のものに切り替えたときのLAswing(レバーアームスウィング距離)の頻度分布 である。ただし形状変化を引き起こすためにRigor状態のGoポテンシャルは25%に弱めて いる。頻度分布は200万ステップの計算を実行し、後半の100万ステップ分の形状から得 たものである。ポテンシャル切り替え無し(黒)に比べ、切り替えた場合(赤)は大きく巻き 戻し側に分布が広がっていることがわかる。このような分布の違いを生み出している源 をさらに特定するため、レバーアームに最も近い残基グループ405・495とその近くにある 残基グループ703・709のみのaicg13およびaicg14ポテンシャルを切り替えて同様のシミ ュレーションを実行した結果がピンクであり、それ以外の残基グループ(567における aicg13およびaicg14ポテンシャルを切り替えてシミュレーションを実行した結果が音である。 これらの結果から、前者はLAswingの分布を両方向に広げる効果があり、後者がそれを巻 き戻し方向に限定させる効果があるらしいことがわかる。レバーアームから遠くはなれ た後者のグループに対するポテンシャル切り替えがこのような効果をもたらすことは興 味深い。今後はこのような試みをGoポテンシャルについても考案し、さらに解析対象を パワーストーク運動にも広げる予定である。この研究を通して状態遷移をより自然に再 現することを目指す。



図 3.1.25 LAswing(レバーアームスウィング距離)の頻度分布

以上のようにミオシン首振りサイクルにおける全ヌクレオチド状態に対し統一した生物種の構造(Myosin-VI)を鋳型にしたサルコメアモデルの高度化を行った。また分子モデルのGo-contact等に関する改良、構造情報を取り入れた状態遷移モデルの改良等に関する研究は引き続き行っていく予定であるが、分子モデル・有限要素モデル間の連成手法については、ミクロ剛性を考慮した多重時間ステップカップリング法を開発し、最終化を行った。

## 参考文献

- [1] Washio T, Okada J, Takahashi A, Yoneda K, Kadooka Y, Sugiura S, Hisada T, Multiscale heart simulation with cooperative stochastic cross-bridge dynamics and cellular structures, SIAM Multiscale Modeling and Simulation, 11(4),965-999(2013)
- [2] http://www.cafemol.org/
- [3] Sugiura S, Kobayakawa N, Fujita H, Yamashita H, Momomura S, Chaen S, Omata M, Sugi H, Comparison of unitary displacements and forces between two cardiac myosin isoforms by the optical trap technique: Molecular basis for cardiac adaptation, Circ Res,82:1029-1034(1998)

- [4] W. Li, P. Wolynes, S. Takada, Frustration, specific sequence dependence, and nonlinearity in large-amplitude fluctuations of allosteric proteins, PNAS (2011) 108 (9) 3504-3509
- [5] W. Li, S. Takada, Energy landscape views for interplays among folding, binding, and allostery of calmodulin domains, PNAS (2014) 111 (29) 10550-10555

次にリモデリングの検証並びに心不全治療と外科手術の慢性期予測法について説明す る。心臓は圧負荷と呼ばれる「収縮期の負荷」および容量負荷と呼ばれる「拡張期の負荷」 に対して、それぞれ壁を厚くする「求心性肥大」、容積を増大する「遠心性肥大」様の変 化をして適応するが(図3.1.26)、適応にも限界があり過大な負荷が持続すると心不全状 態に陥る。また遠心性肥大においては心室の形状が正常に見られる回転楕円体から球形 に近づくことも知られている。



図 3.1.26 心室の適応([1]より改変)

こうした現象の背後には細胞レベルの変化があり求心性肥大では個々の細胞がサルコメアの横方向 への数の増加をともなって軸に垂直な方向に肥大するのに対し、遠心性肥大ではサルコメアが縦方 向に数を増すことによって細胞は軸方向に伸長することが知られている(図 3.1.27)。



図 3.1.27 肥大に伴う細胞レベルの変化([2]より改変)

容量負荷をもたらす代表的な疾患には逆流性の弁膜症がある。手術による弁の修復または人工弁 への置換によって治療することができるが、手術のリスクを考慮すると治療のメリットのない軽症 には行うべきでないというのがコンセンサスである。ところが一方で重症例においては手術の時期 を逸すると、治療の効果が得られずデメリットしか残らないという問題があり、手術時期の見極めは 極めて重要である。手術に対する反応を決定する要因は心筋の変化が適応の範囲内であるか否かで あるが、その見極めは現在の臨床検査では極めて困難であり、現在のガイドラインでは心臓の大きさ のみが判断基準になっていることが多い[3][4]。

細胞レベルの変化をもたらすものはメカノセンサーと呼ばれる複数の分子であり、その多くは細胞への力学刺激の方向(細胞の軸方向か否か)およびタイミング(収縮期か拡張期か)を感知できるようにサルコメア内に合理的に分布している。代表的なメカノセンサーであるタイチンはサルコメア内でZ帯からミオシンフィラメントをつなぎ、その中に複数のセンサーエレメントを持っている[5][6](図 3.1.28)。



図 3.1.28 タイチンの構造と信号伝達機構([6]より改変)

容量負荷においては弛緩しクロスブリッジが形成されていない状態のサルコメアが伸展されるの で図 3.1.29 に示すようにタイチンの I-band region が伸展されることになる。



図 3.1.29 サルコメア長とタイチンの構造変化の関係([5]より改変)

一方、圧負荷ではクロスブリッジが形成された収縮期のサルコメアに伸展が加えられるため、I-band region には伸展が加わらずむしろミオシンフィラメントに付着した領域(M-line region)のタイチ

ンと Z-disk に伸展刺激が加わると考えられる(図 3.1.30)。



図 3.1.30 収縮期におけるサルコメアおよびタイチンの構造

いずれの場合にもセンサー分子は細胞(サルコメア)の長軸方向への機械的刺激に反応する構造になっており、リモデリングの検討には心臓内における線維(細胞)の方向の機械刺激を考慮することが 重要である。しかし現状の臨床検査によって線維構造を知ることはできない。

一方、前述の通り容量負荷は回転楕円体から球形への心室の形状変化を伴っているが、この球形への変化が心不全の重症度および予後と関係していることが再確認されている[7][8]。またこれに伴う線維方向の変化も示唆されており臨床における線維方向の測定を目指した研究が進められている[9]。

容量負荷の解除のモデルとして外科手術の慢性期のリモデリング予測法を完成した。以下に形状の決定に焦点を合わせて方法を示す。

Step 1: 術前モデルの作製

測定された拡張末期容積、拡張末期圧から Klotzの方法によって拡張期圧・容積関係を決定する[10]。 これによって心室壁の各セグメントの自然長も決定される。さらにこの圧・容積関係を実現するよう 以下に示す心筋組織の物性を表す構成式のパラメータを決定する。

 $W_{pass} = \mathbf{a} \cdot W_{Hum} + b \cdot W_{Lin}$ 

 $W_{Hum} = c_1(\alpha - 1)^2 + c_2(\alpha - 1)^3 + c_3(l_1 - 1) + c_4(l_1 - 1)(\alpha - 1) + c_5(\alpha - 1)^2$ 

 $W_{Lin} = c_6(e^Q - 1)$ 

$$Q = c_7(l_1 - 3)^2 + c_8(l_1 - 3)(l_4 - 1) + c_9(l_4 - 1)^2$$

ここで  $I_1 = trC$ ,  $I_4 = fCf$ ,  $\alpha = \sqrt{fCf}$  であり、f は無負荷形状における心筋細胞の向きを示す単位 ベクトルである。

Step 2: 外科手術シミュレーションによる急性効果の評価

心臓のパラメータを変更することなく、手術による介入のシミュレーションを行う。下に弁膜症手術 によって容量負荷が軽減された様子を図 3.1.31 に模式的に示すが、丸で表した拡張末期の状態は Step 1 で決定した圧・容積関係の上を移動する。すなわちこの時点では心室組織の物性の変化は起こ っていない。



図 3.1.31 手術による容量負荷軽減の急性効果

心筋組織の各要素について見ると、手術の結果伸展が軽減されている。これがリモデリングの刺激と なるとのモデルに従って次のリモデリングシミュレーションを行う。

Step 3: リモデリングシミュレーション

図 3.1.29 から容量負荷に対するメカノセンサーは閾値以上の伸展に対して信号を発生すると考えられる。手術による過伸展の軽減がリモデリングを促進するとのモデルを構築した。

心室内各セグメントにおける線維方向の strain (S) の手術前の値、手術直後の値、リモデリング後の値をそれぞれ $S_0$ 、 $S_1$ 、 $S_2$  とすると

 $S_2 - S_1 = -0.86 \cdot (S_0 - S_1)$ または

 $S_2 = -0.86 \cdot (S_0 - S_1) + S_1$ 

この関係を全ての要素に適用した。

閾値はサルコメアについての実験研究から1.1と設定した5。

外科手術の慢性期予測として、マルファン症候群に伴う大動脈弁閉鎖不全症に対し弁置換術を受けた症例について検証結果を以下に示す。

術前の臨床データに基づいて個別化心臓モデルを作製し、電気生理現象から血行動態までを再現 したマルチスケール・マルチフィジックスシミュレーションを行った。心電図シミュレーションの結 果を図 3.1.32 に示す。実際の心電図とシミュレーションの良い一致は患者の電気生理現象が正確に 再現されたことを表している。



図 3.1.32 電気生理シミュレーションによる心電図の再現。 上段:シミュレーション結果、下段:実際の患者の心電図

Klotz らの方法に基づいて拡張末期の左室容積および左室圧から自然形状を含む拡張期圧-容積関係 を決定した(Step 1)。この大動脈弁閉鎖不全を伴う心臓モデルに対して、手術シミュレーション(弁 閉鎖不全の解除)を施行し、急性効果を観察した(Step 2)。 手術前後の興奮伝播および収縮の様子を図 3.1.33 に比較する。



術前(大動脈弁閉鎖不全)。

左心室の縮小などの効果は図 3.1.34 示す左心室の圧容積関係より明らかである。



図 3.1.34 左心室圧-容積関係 黒破線:手術前(大動脈弁閉鎖不全あり) 黒実線:手術後(閉鎖不全な し) 黒細線:拡張期圧-容積関係

拡張末期における心筋の線維方向における自然形状からの伸び率を比較したものが図 3.1.35 である。



図 3.1.35 拡張末期における線維方向の伸び率

心筋のサルコメアは無負荷状態のおよそ  $1.9 \mu$  m から  $2.2 \sim 2.4 \mu$  m の間で作動しているので[11]伸 び率は 1.1 から 1.26 程度にあると考えられる。この数字と比較すると術前においては左心室の赤道 面を中心に過度の伸展が見られ、特に中隔側で範囲が広い。術後は全体的に伸び率の低下が見られた が歪の閾値を越える部位が広く認められる。図 3.1.36 に手術前後での伸び率(stretch ratio: SR)の 関係を閾値を越えた要素について示す。赤線で示した line of identity (傾き 1、切片 0 すなわち SR AR(-)=SR AR(+))上の対応する点との差が解除された分の過伸展となる。



図 3.1.36 各要素における伸び率(SR)の手術前(SR AR(+))と手術後(SR (AR(-))の比較。 赤線: line of identity

また左心室腔の短径と長径の比(R/L)は sphericity index (SI) と呼ばれるが、術前は 0.77 であるの に対し術後では 0.68 とより細長い形状に変化しているものの SI の正常値 (0.45~0.62) [12]に比べ れば大きく完全な回復には至っていない。本シミュレーションでは大動脈弁閉鎖不全の治療が赤道 面を中心にした容量負荷による過伸展を是正したが歪は残留し SI は正常化しないという結果が得ら れた。

引き続き Step 3 に従ってリモデリングのシミュレーションを行った。各要素の伸び率を手術前の値 と比較したものを図 3.1.37 に示す(青色)。手術直後(AR(-):オレンジ色)に比べ明らかに伸展の 軽減が認められる。



図 3.1.37 伸展の軽減度 各要素について手術前の伸展度(SRAR(+))と比較した。 青:手術直後、 オレンジ:リモデリング後

心室全体については図 3.1.38 に示す通り拡張末期容積とともに拡張末期圧が減少しており多少では あるがリモデリング効果が認められる。



図 3.1.38 術後の圧容積関係の比較 黒破線:術前 黒実線:術後(急性効果) 赤実線:リモデリ ング後。青点線、赤点線はそれぞれリモデリング前後の拡張期圧-容積関係を示す。

線維方向の伸び率はリモデリングシミュレーションの結果閾値にほぼ収束し、また SI も 0.67 と多

少改善した。この症例ではリモデリングの刺激となる伸び率はリモデリング前よりかなり低下して おりリモデリング効果は比較的小さいものであった(図 3.1.39)。



図 3.1.39 リモデリング後の線維方向の伸び率

本症例においては術後1か月の時点でCTによる経過観察が行われた。そのデータから再構成した 左心室の形状をシミュレーションと比較したところ、術後の拡張期容積は300mlでリモデリング後 のシミュレーション結果とほぼ一致していた。さらにSIについても0.68と予測値に良く一致して おり本法により手術後のリモデリングが精度よく予測できたことを示している(図3.1.40)。



図 3.1.40 左心室形状の比較

想定されるリモデリングの過程を図 3.1.41 に示す。心室の拡張によっても細胞への伸展刺激が解除されないような場合には過度の刺激の持続が非可逆的な変化を細胞にもたらす。本症例では手術の後に残留したリモデリング刺激が小さかったため慢性期の効果が少なかったものと考えられる。



図 3.1.41 容量負荷およびその解除に伴う左心室リモデリング

次に心不全治療の慢性期予測として心臓再同期療法におけるリモデリングについての検証結果を示す。

心臓再同期療法(Cardiac Resynchronization Therapy: CRT)の効果判定には予後および症状の改善とならんで左室のリモデリングの程度が用いられるが、その中でも収縮末期容積(End-systolic volume: ESV)の減少が多く採用されている。CRTの対象となる左心室の収縮非同期を呈する症例では遅れて興奮収縮する領域は早く収縮する領域によって伸展されるが、逆に収縮が始まると先行して収縮した領域を伸展する(図 3.1.42)。



図 3.1.42 収縮の非同期を呈する症例の左心室の収縮様式。 断面上の各色 のの点における壁厚の変化を示す。[13]より引用

すなわち、圧負荷に似た収縮している組織の伸展が起きており容量負荷の評価に用いられた sphericity index は有効ではなく<sup>8</sup>、当然拡張期の伸展もリモデリングの刺激とはならない。CRT が もたらすリモデリング効果を予測するには収縮の効果と伸展の効果を併せて評価する必要がある。 実際に臨床研究においても CRT 植込み前の正(伸展) 歪と負(短縮) 歪の比が ESV の減少と相関 するとの報告がある。今回この事実を応用し、個別化 CRT シミュレーションによって治療前後の正 (伸展) 歪と負(短縮) 歪の比を計算することで、ESV の減少を精度予測する方法を完成した。

方法:

心不全の病態におけるリモデリング刺激を定量化するために左心室を分割した各領域における仕事 (strain: ɛ×stress:S)を正仕事(Positive work: P)および負仕事(Negative work: N)それぞれに ついて収縮期の間について計算し比較している。

$$e(t) = \frac{d\varepsilon(t)}{dt} \qquad e_P = max(0, e(t)), \ e_N = min(0, e(t))$$
$$P = \int_0^T e_P(t) \cdot S(t) dt$$
$$N = \int_0^T e_N(t) \cdot S(t) dt$$

T=systolic time interval

治療効果予測については左心室全体についての正仕事と負仕事の総和の比を指標とした。 方法の検証のために CRT 植込み前の臨床データに基づいて個別心臓モデルを作製し、CRT シミュ レーションを実施した。CRT 前後における正仕事と負仕事の分布の例を図 3.1.43 に示す。



図 3.1.43 CRT 前後における負仕事および正仕事の分布の比較 左:分割の方法、星印はペーシン グ電極の位置を示す。 右:負仕事(上段)と正仕事(下段)の変化 体積平均 [µJ]を色で示 す。

明らかに CRT によって負仕事が減少し正仕事が増加したことが分かる。8 例において P/N比と 3 か 月後の時点での ESV の減少率((CRT 前の ESV-CRT 後の ESV) /CRT 前の ESV)を比較したと ころ図 3.1.44 のように良好な相関を得、CRT 症例におけるリモデリング予測法を確立することがで きた。



図 3.1.44 CRT 前後における Negative work と Positive work の比の変化と ESV の変化の関係

以上、容量負荷のモデルとしての心臓弁膜症の手術後の予後予測、圧負荷と容量負荷の両方の要素を もつモデルとしての心不全に対する心臓再同期療法の効果予測の方法を確立した。

# 参考文献

- Katz AM. Heart failure. Physiology of the heart. Philadelphia: Lippincott Williams & Wilkins; 2001:630-657.
- [2] Opie LH. The Heart physiology, from cell to circulation. 3rd ed. Philadelphia: Lippicott-Raven; 1998.
- [3] Borer JS, Supino PG, Herrold EM, Innasimuthu A, Hochreiter C, Lrieger K, Girardi LN, Isom OW. Survival after Aortic Valve Replacement for Aortic Regurgitation: Prediction from Preoperative Contractility Measurement. Cardiology 2018;140:204-212.
- [4] Cox DA, Walton K, Bartz PJ, Tweddell JS, Frommelt PC, Earing MG. Predicting left ventricular recovery after replacement of a regurgitant aortic valve in pediatric and young adult patients: is It ever too late? Pediatr Cardiol 2013;34:694-699.
- [5] Granzier HL, Labeit S. The giant protein titin A major player in myocardial mechanics, signaling, and disease. Circ Res 2004;94:284-295.
- [6] Linke WA, Lruger M. The giant protein titin as an integrator of myocyte signaling pathways. Physiology 2010;25:186-198.
- [7] Roscani MG, Polegato BF, Minamoto SET, Lousada APM, Minicucci M, Azevedo P, Matsubara LS, Matsubara BB. Left ventricular sphericity index predicts systolic dysfunction in rats with experimental aortic regurgitation. J Appl Physiol 2014;116:1259-1262.
- [8] Zeng D, Chen H, Jiang CL, Wu J. Usefulness of three-dimensional spherical index to assess different types of left ventricular remodeling A metanalysis. Medicine 2017;96.

- [9] Watson S, Dormer JD, Baowei F. Imaging technologies for cardiac fiber and heart failure: a review. Heart Fail Rev 2018;23:273-289.
- [10] Klotz S, Dickstein ML, Burkhoff D. A computational method of prediction of the end-diastolic pressure-volume relationship by single beat. Nat Protoc 2007;2:2152-2158.
- [11] Ter Keurs HEDJ, Rijnsburger WH, van Heuningen R, Nagelsmit MJ. Tension development and sarcomere length in rat cardiac trabeculae Evidence of length-dependent activation. Circ Res 1980;46:703-714.
- [12] 有田武史. 心エコーによる心不全モニタリング. J Cardiol Jpn Ed 2012;7:159-164.
- [13] Okada J-I, Wahio T, Nakagawa M, Watanabe M, Kadooka Y, Kariya T, Yamashita H, Yamada Y, Momomura S, Nagai R, Hisada T, Sugiura S. Multi-scale, tailor-made heart simulation can predict the effect of cardiac resynchronization therapy. J Mol Cell Cardiol 2017;108:17-23.

(2)目標2(創薬を加速する心毒性スクリーニングシステムの開発)

本テーマでは、重点課題1(奥野恭史・重点課題責任者)の寺田 透博士との連携により、心臓シ ミュレータ UT-Heart と心筋細胞イオンチャネルの分子シミュレーションを融合し、創薬における 候補物質の分子構造と遺伝子情報のみから不整脈のリスク評価を可能とするシステムを開発するこ とを目的とする。



図 3.2.1 UT Heart と細胞パッチクランプ実験を組み合わせたハイブリッド心毒性スクリーニング システム

具体的には、既に開発済の図 3.2.1 に示される UT-Heart と細胞パッチクランプ実験と組み合わせた 「ハイブリッド心毒性スクリーニングシステム」の細胞薬理実験(赤破線内)の部分を分子シミュレ ーションに置き換える。



図 3.2.2 ハイブリッド心毒性スクリーニングシステムによる 12 薬剤の解析結果 (Okada et al, Science Advances, 2015 [1])

本ハイブリッド心毒性スクリーニングシステムは、図 3.2.2 に示されるように既に 12 種類の薬剤に 対し検証を終え、従来のガイドラインにおける指標(例えば QT 間隔延長)に較べ偽陽性も出ない など優れた判定が出来ることが示されている[1]。



図 3.2.35 次元心電図データベース[2]

更に、ハイスループット心毒性評価システム実現へ向けての戦略として、各チャネルの阻害率が得られた後に毎回大規模心臓シミュレーションを実施するコストを省くことを目的とし、図 3.2.3 に示すような割合で INa、IKs、IKr、ICa、INaL の各電流が抑制された場合の 5 次元心電図データベースを作成済である[2]。



図 3.2.45 次元心電図データーベース[2]

一例として、図 3.2.4 に同データベースを活用した Bepridil の評価を示す。左下の図のような阻害 曲線のパラメータ(後述の IC50 と Hill 係数)を入力すれば、大規模計算を行うことなしに薬剤の 不整脈発生リスクを判定できるだけでなく、各イオン電流の抑制の効果が多次元的に理解できる。

分子シミュレーション開発は二つのパートに大別される。第1は電位センサードメインの立体構 造変化に関するシミュレーション手法の開発、第2は不整脈を誘発する可能性のある薬剤と心筋イ オンチャネルとの間の相互作用を予測するためのポアドメインでのドッキングシミュレーションに 基づく手法の開発である。本年度は、カリウムチャネルの一つである hERG イオンチャネル並びに ナトリウムチャネル Nav1.5 と薬剤のドッキングシミュレーションを実施し、これと細胞パッチク ランプ実験により求めたチャネル阻害率に基づく不整脈発生リスクを比較することにより検証を行 った。なお、hERG チャネルは 2017 年 5 月 Wang ら[3]により公表された電子顕微鏡構造(PDB ID: 5VA2)を用い、ナトリウムチャネル Nav1.5 は Yan らの電気ウナギの NaV1.4 の電子顕微鏡構 造 (PDB ID: 5XSY)を鋳型としてモデリングを行った。



図 3.2.5 hERG チャネルの立体構造(重点課題1寺田博士作成)

図 3.2.5 は 2018 年 5 月に公表されたカリウムイオンチャネルの立体構造を上(細胞外) および横 (細胞膜切断面) から見たものである。周囲の 4 領域(ホモ 4 量体構造) は電位センサードメイン (VSD) の立体構造であり、各電位センサードメインを構成する 4 本のヘリックスのうちの一つ S4 の正に荷電した残基が電位センサーの役割を果たし、脱分極状態では S4 が細胞外側に移動する(Up 状態となる) ことによりチャンネルが開く。また分極状態では S4 が細胞質側に移動する(Down 状 態となる) ことによりチャンネルは閉まる。なお赤丸によって囲まれた細孔を構成する領域はポアドメ インの立体構造であり、ここに細胞内部より薬剤が結合することでチャネル機能を阻害する。



図 3.2.6 イオンチャネル分子シミュレーションと UT-Heart の新しい連成方法

図 3.2.1 に示したハイブリッド心毒性スクリーニングシステムの破線部を MD シミュレーション に基づく方法に置き換えた新たな連成手法の概念を図 3.2.6 に示す。このシステムによれば、新たな 薬剤に対してチャネル阻害率を評価するドッキングシミュレーションと結合自由エネルギー計算の ための MD シミュレーションは毎回行う必要があるものの、膜電位に依存するチャネルの構造変化 は確率モデルに置き換えているため、構造変化の分子シミュレーションについては再度行う必要が ない。多数の候補薬に対する心毒性評価を効率的に行うことを目指した現実的連成法であると言え る。

図 3.2.6 におけるチャネル阻害率をαと表記すると以下の式が成立する。

$$E + I \rightleftharpoons EI \qquad K_d = \frac{[E][I]}{[EI]}$$

$$[I]_0 = [I] + [EI]$$

$$[E]_0 = [E] + [EI] = [EI] \left(1 + \frac{K_d}{[I]}\right) = ([I]_0 - [I]) \left(1 + \frac{K_d}{[I]}\right)$$

$$\alpha = \frac{[EI]}{[E]_0} = \frac{[I]}{[I] + K_d}$$

ここで[I]<sub>0</sub>は薬剤濃度、[E]<sub>0</sub>はチャネル濃度、K<sub>d</sub>は解離定数である。第3行目の式は[I]に関する2 次方程式になるので、これを解いて最後の式に代入すれば阻害率 $\alpha$ が得られる。従って、基本的には 解離定数 K<sub>d</sub>を分子シミュレーションから求めれば良いことになるが、具体的手順としては $\alpha$ =0.5 と なる薬剤濃度 IC<sub>50</sub>を以下のように評価する。 ・hERG チャネルの場合:

1. ドッキングシミュレーションと MD シミュレーションにより、ある薬剤 A について結合自由 エネルギー値 dG<sub>calc</sub> を求める

2. 同じ薬剤 A について阻害定数の実験値 Kdexp から結合自由エネルギーの実験値を理論式 dGexp=RTlogKdexp にて求める

3. 以上により薬剤 A について  $dG_{exp}$  と  $dG_{cale}$  の値の組が求められた。これを 12 化合物につい てプロットして回帰直線を決定

4. ある薬剤 B について dG<sub>calc</sub> をドッキングシミュレーションと MD シミュレーションで求め、 上記の回帰直線に当てはめて結合自由エネルギーの(実験値に基づき補正された)予測値 dG<sub>pred</sub> を求める

5. dGpred から Kd の予測値を Kdpred=exp(dGpred/RT) にて求める

6. IC<sub>50</sub>=Kd+0.5\*E0を用いて IC<sub>50</sub>を求める (チャネル濃度[E]<sub>0</sub>は仮に 10 nM としているが、Kd の値が[E]<sub>0</sub>に較べ大きいので[E]<sub>0</sub>に対する感度は低い)。

・Nav1.5 の場合:

1. ドッキングシミュレーションと MD シミュレーションにより、ある薬剤 A について結合自由 エネルギー値 dG<sub>calc</sub> を求める

2. 同じ薬剤 A について阻害定数の実験値 Kdexp から結合自由エネルギーの実験値を理論式 dGexp=RTlogKdexp にて求める

3. 以上により薬剤 A について dG<sub>exp</sub> と dG<sub>calc</sub> の値の組が求められた。これを 12 化合物につい てプロットして回帰直線を決定

4. ある薬剤 B について dG<sub>calc</sub> をドッキングシミュレーションと MD シミュレーションで求め、 上記の回帰直線に当てはめて結合自由エネルギーの(実験値に基づき補正された)予測値 dG<sub>pred</sub> を求める

5. dGpred から Kd の予測値を Kdpred=exp(dGpred/RT) にて求める

6. IC<sub>50</sub>≒Kd<sub>pred</sub>にて IC50 を近似する(Kd の値がチャネル濃度[E]<sub>0</sub>に較べ大きいので、
 IC<sub>50</sub>=Kd+0.5\*[E]<sub>0</sub>における[E]<sub>0</sub>の項は無視できる)

なお、上記1における結合自由エネルギー計算には藤谷らによる MP-CAFEE 法[4]を使用した。

以上の IC<sub>50</sub> に加え、不整脈リスク予測には IC<sub>50</sub> おける阻害率の薬剤濃度に対する勾配である Hill 係 数が必要となる。これは Hill の式へのフィッティングから求められるが、以下に示すように事実上 きわめて-1 に近い値となる。

10-11-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1-1							
	  合回の値(MD)			従亚の値(実験)			
		ラ回の値(MD)					
	iC50(pred)µivi	log(ICSU)	HIII	1050	log(ICSU)	HIII	
Amiodarone	0.126	-0.900	-1.000	0.756	-0.122	-0.820	
Astemizole	0.001	-2.835	-1.019	0.028	-1.551	-1.750	
Chlorpromazine	13.730	1.138	-1.000	1.118	0.048	-0.900	
Cisapride	0.022	-1.661	-1.001	0.015	-1.832	-1.330	
Dofetilide	0.032	-1.492	-1.001	0.010	-2.009	-0.890	
Sotalol	3.356	0.526	-1.000	356.400	2.552	-1.020	
Verapamil	0.514	-0.289	-1.000	0.201	-0.696	-1.040	
INaチャネル							
	今回の値(MD)			従来の値(実験)			
	IC50μM	log(IC50)	Hill	IC50	log(IC50)	Hill	
Amiodarone	1.890	0.277	-1.018	0.975	-0.011	-0.750	
Astemizol	7.090	0.851	-1.004	1.862	0.270	-1.530	
Dofetildie	73.020	1.863	-1.000	124.500	2.095	-0.330	
E-4031	1521.640	3.182	-1.000	737.300	2.868	-0.330	
Verapamil	0.710	-0.149	-1.050	4.272	0.631	-0.850	

表 3.2.1 分子シミュレーションおよびパッチクランプ実験から得られた IC50 と Hill 係数の比較

以上の方法に基づき、hERG チャネルに対しては、Amiodarone、Astemizole、Chlorpromazine、 Cisapride、Dofetilide、Sotalol、Verapamil の 7 薬剤について、Nav1.5 チャネルに対しては、 Amiodarone、Astemizole、Dofetilide、E-4031、Verapamil の 5 薬剤について IC<sub>50</sub> と Hill 係数を 求め、従来の実験値と比較した結果を表 3.2.1 に示す。



図 3.2.7 7 種の薬剤に対する hERG チャネル電流(青色線)と Nav1.5 チャネル電流(黒色線)の阻害率 曲線の比較。参考のため実験値としての ICa、IKs、INaL 電流の阻害率曲線(赤色、緑色、空色線)も 併記した。緑色丸印は各薬剤の常用量に相当する濃度。 また、図 3.2.7 は 7 種の薬剤に対する hERG チャネル電流(青線)と Nav1.5 チャネル電流(黒 線)の阻害率曲線を比較したものである。参考のため実験値としての ICa、IKs、INaL 電流の阻害 率曲線も併記した。緑色丸印は各薬剤の常用量に相当する濃度である。



図 3.2.8 Amiodaron の濃度を変化させた場合の心電図の比較

ー例として、Amiodaronの濃度を変化させた場合の心電図の比較を図 3.2.8 に示す。左列(MD)で は hERG と Nav1.5 チャネルの阻害率に分子シミュレーションの結果を用いた。他のチャネルの阻 害率については実験値を採用している。分子シミュレーションを用いた結果では常用量の 800 倍で TdP 様の不整脈が発生していることが分かる。一方、従来の実験データに基づく方法では不整脈は 発生しない(右列)。その原因は特に hERG チャネルの阻害の程度を、実験よりシミュレーションの 方が高めに評価しているからであると想像される。



図 3.2.9 データベースを用いた Amiodaron の分析

そこで更に5次元心電図データベースを用いて分析を行ったものが図 3.2.9 である。IKr、INa、ICa の 3 電流の抑制率を 3 軸として Amiodaron の濃度を赤丸で示される常用量から増加させていくと 図中のようなトラジェクトリーが得られた。奥にある暗色領域は不整脈発生領域であり、ここに初め て到達する濃度が不整脈発生の閾値となる。従来の実験に基づく方法では Amiodaron の濃度増加と 共に IKr のみならず ICa と INa が抑制され、これによって濃度が増加しても不整脈発生領域に突入 することが回避されていること、一方で MD シミュレーションに基づく今回の計算結果では、IKr の 抑制率が高いことに加え INa の抑制率が実験より低めであることから不整脈発生領域に突入するこ とが分かる。



図 3.2.10 Chlorpromazine の濃度を変化させた場合の心電図の比較

次に Chlorpromazine の濃度を変化させた場合の心電図の比較を図 3.2.10 に示す。Amiodaron と同様、左列(MD)では hERG と Nav1.5 チャネルの阻害率に分子シミュレーションの結果を用い、他のチャネルの阻害率については実験値を採用している。従来の実験に基づく方法、今回の分子シミュレーションに基づく方法の何れでも不整脈は発生しない。



図 3.2.11 データベースを用いた Chlorpromazine の分析

図 3.2.11 は Amiodaron の場合と同様、5 次元心電図データベースを用いて IKr、INa、ICa の 3 電

流の抑制率を3軸として Chlorpromazine の濃度を赤丸で示される常用量から増加させていくこと により図中のようなトラジェクトリーを得た。従来の実験に基づく方法では ICa 電流と INa 電流の 抑制により、かろうじて不整脈発生を回避している様子が観察できるが、一方で MD シミュレーシ ョンに基づく方法では、かなり安全裕度が高めに評価されていることが分かる。その原因は図 3.2.10 に立ち戻ると、IKr の阻害度が MD シミュレーションでは低めに評価されていること、そして INa の阻害度は実験値と大同小異であることから理解できる。

表 3.2.2 分子シミュレーションおよびパッチクランプ実験から得られた7薬剤のリスク評価結果の比較

Drug name	CiPA risk (Gintant, Sager & Stockbridge, 2016)	Redfern risk (Redfern et al., 2003)	CredibleMeds (Woosley, Heise & Romero)	データベースによる予測 (実験値データ)		データベースによる予測 (分子シミュレーション)	
				不整脈発生の 有無	Threshold concentrations relative to ETPCunbound	不整脈発生の 有無	Threshold concentrations relative to ETPCunbound
Dofetilide	High	1	Known Risk of TdP	Arrhythmia	21	Arrhythmia	60.3
dl-Sotalol	High	1	Known Risk of TdP	Arrhythmia	71	Arrhythmia	0.7
Cisapride	Intermediate	2	Known Risk of TdP	Arrhythmia	13	Arrhythmia	25.2
Chlorpromazine	Intermediate	-	Known Risk of TdP	Safe	-	Safe	-
Astemizol	Intermediate	2	Known TdP risk	Arrhythmia	182	Arrhythmia	119
Verapamil	Low	5	-	Safe	-	Safe	-
Amiodarone	-	1	Known Risk of TdP	Safe	-	Arrhythmia	491

表 3.2.2 にデータベースを用いた 7 種の薬剤に対する不整脈発生リスクの予測結果を示す。青色の 2 列が従来の実験に基づく方法による不整脈発生の有無ならびに不整脈発生閾値(常用量 (ETPCunbound)に対する倍率)、クリーム色の 2 列が今回の分子シミュレーションによる同種の結 果である。参考の為、第 2 列目に CiPA (FDA 等によって提案された <u>Comprehensive in vitro</u> <u>Proarrhythmia Assay</u>)によるリスク分類、第 3 列目に Redfern[5]によるリスク分類、第 4 列目に Woosley 等[6]による TdP リスクの有無を併記した。

本年度は、重点課題1(奥野恭史・重点課題責任者)の寺田 透博士との連携により、カリウムチ ャネルの一つである hERG チャネルに対して7種の薬剤のドッキングシミュレーションを実施し (そのうち4薬剤についてはナトリウムチャネルNav1.5とのドッキングシミュレーションも実施)、 得られたチャネルの阻害率から心電図データベースを活用して不整脈発生リスクを評価した。そし て従来の細胞パッチクランプ実験により求めたチャネル阻害率に基づくリスク評価結果と比較した。 当然のことながら MD およびパッチクランプ実験に基づく両結果には、薬剤ごとに様々な程度の差 異が生じている。その理由に関して、前記の如く Amiodaron と Chlorpromazine を例にとり 5 次元 データベースに基づく検証を実施し知見を得た。他の薬剤についてもほぼ類似の分析となるため省 略するが、今後は実用化を視野に入れ、MD によるドッキングシミュレーションからイオンチャネル の阻害率評価に至る各過程での不確定性の分析と精度改善、パッチクランプ実験において避けられ ないばらつきの評価等を総合しシステムを完成させて行く必要があると考えられる。

## 参考文献

- Okada J, Yoshinaga T, Kurokawa J, Washio T, Furukawa T, Sawada K, Sugiura S, Hisada T, Screening system for drug-induced arrhythmogenic risk combining a patch clamp and heart simulator, Science Advances, 1 (4), e1400142(2015)
- [2] Okada J, Yoshinaga T, Kurokawa J, Washio T, Furukawa T, Sawada K, Sugiura S, Hisada T, Arrhythmic hazard map for a 3D whole-ventricles model under multiple ion channel block British Journal of Pharmacology (2018) DOI: 10.1111/bph.14357
- [3] Wang W, MacKinnon R, Cryo-EM Structure of the Open Human Ether-à-go-go-Related K+ Channel hERG, Cell,169,3, 422–430.e10(2017)
- [4] Fujitani H, Tanida Y, Ito M, Jayachandran G, Snow CD, Shirts MR, Sorin EJ, Pande VS. Direct calculation of the binding free energies of FKBP ligands. J. Chem. Phys. (2005) 123 084108
- [5] W. S. Redfern, L. Carlsson, A. S. Davis, W. G. Lynch I.MacKenzie, S. Palethorpe, P. K. Siegl, I. Strang, A. T. Sullivan, R. Wallis, A. J. Camm, T. G. Hammond, Relationships between preclinical cardiac electrophysiology, clinical QT interval prolongation and torsade de pointes for a broad range of drugs: Evidence for a provisional safety margin in drug development. Cardiovasc. Res. 58, 32-45 (2003)
- [6] Woosley RL, Heise CW, Romero KA, QTdrugs List. Available at www.CredibleMeds.org, AZCERT, Inc. 1822 Innovation Park Dr., Oro Valley, AZ 85755

#### 【今年度計画と成果】

平成30年度業務計画は、

- 分子モデルとリモデリング・慢性期予測、連成手法最終化 心臓病、特に「心不全」の根本的解明と最適治療を可能とするマルチスケール心臓シミュレータ の開発を目的とする。モデル化は心不全研究において得られた遺伝子レベルの変化、発現調節、 細胞内情報ネットワークに関する膨大な情報、臨床情報を効果的に取り入れて推進する。 平成30年度は、サルコメアモデルとリモデリングの高度化及び検証を行う。また連成 手法を最終化する。さらに心不全治療と外科手術の慢性期予測法を完成する。
- チャネル分子と薬剤のドッキングシミュレーション・検証
   目標1は分子レベルの力学と心臓のポンプ機能の関係に焦点を当てたものであるが、同様の技術を致死性不整脈の病態解明と予防・治療に対して適用する。即ち、重点課題1において進めら

れるイオンチャンネルの分子シミュレーションとUT-Heart を融合することで、候補物質の分子 構造と遺伝子情報のみから不整脈のリスク評価を可能とするシステムの完成を目標とする。 平成30年度は、イオンチャネルと薬剤のドッキングシミュレーションを実施し、細胞実験によ り検証する。

成果は、

1. サルコメアモデルの高度化に関しては、ミオシン首振りサイクルにおける全ヌクレオチド 状態に対し統一した生物種の構造(myosin-vi)を鋳型にした分子モデルを新たに作成し、この 新モデルにおいてレバーアームのパワーストロークや発生張力等が実験値と比較して妥当な 値を示すことを検証した。リモデリングの高度化と検証ならびに心不全治療と外科手術の慢 性期予測法に関しては、圧負荷・容量負荷の概念ならびにサルコメア内構造に関する生物学 的知見から線維方向のストレッチに関するリモデリング予測式を提案し、これにより外科手 術についてはマルファン症候群に伴う大動脈弁閉鎖不全症に対し弁置換術を施行した症例を 用いて検証を行い、また心不全治療については心臓再同期療法(CRT)の症例において検証 を行い、治療前後の正(伸展) 歪と負(短縮) 歪の比を計算することで、何れも良好な予測が 行えることを示した。また、分子モデル・有限要素モデル間の連成手法については、モンテカ ルロ法を活用し、またミクロ剛性を考慮した多重時間ステップカップリング法を取り入れる ことで、最終化した。

2. カリウムチャネルの一つである hERG チャネルならびにナトリウムチャネルの分子モデルを作成し、 前者については 7 種類、後者については 5 種類の薬剤とのドッキングシミュレーション並びに結合自由 エネルギーシミュレーションを行い、各イオンチャネルの阻害率を評価すると共に、これに基づく心臓興 奮伝播シミュレーションを実施し不整脈発生リスクを評価した。一方、細胞パッチクランプ実験により計 測したチャネル阻害率に基づく心臓興奮伝播シミュレーションも同様に行い不整脈発生リスクを評価し、 両者を比較した。両結果には薬剤ごとに様々な程度の差異が見られたが、その原因につき Amiodaron と Chlorpromazine を例にとり 5 次元データベースに基づく検証を実施した。その結果、Amiodaron につ いての MD シミュレーションでは、カリウム電流の抑制率が高いことに加えナトリウム電流の抑制率が 実験より低めであることからパッチクランプ実験ベースでは起こらなかった不整脈発生が生じること、 いずれの方法でも不整脈が発生しなかった Chlorpromazine について、MD シミュレーションではカリ ウム電流の抑制率が低めに評価される一方、ナトリウム電流の抑制率はパッチクランプ実験と大同小異 であることなどの知見が得られた。

## 4. プロジェクトの総合的推進

プロジェクト全体で密な連携を取りながら運営が円滑に進むよう、業務計画を策定・審議し、運営委員 会や研究会を主催し、また成果を積極的に公開した。

(1) 運営委員会

プロジェクト運営を円滑かつ効率的に行うために、重点課題 2 責任者、サブ課題責任者などで構成した運営委員会を実施している。研究開発およびプロジェクトの総合的推進の活動進捗の確認や、計算資源の配分、課題についての調整等を、サブ課題参画各機関と協調して行った。

平成30年度は以下の日程で開催した。

第1回	4月20日(金)	東京大学医科学研究所にて開催
第2回	6月15日(金)	東京大学医科学研究所にて開催
第3回	7月20日(金)	東京大学医科学研究所にて開催
第4回	9月21日(金)	東京大学医科学研究所にて開催
第5回	10月19日(金)	東京大学医科学研究所にて開催
第6回	12月6日(木)	東京大学医科学研究所にて開催
第7回	2月21日(木)	東京大学医科学研究所にて開催
第8回	3月22日(金)	東京大学医科学研究所にて開催

(2) 諮問委員会

プロジェクトの推進に資するため外部有識者からなる諮問委員会を開催した。研究開発の成果や プロジェクト体制、今後の方向性などについて、諮問委員から幅広く意見を伺い平成 30 年度の研究 開発活動に反映させた。

A) 諮問委員(◎委員長)

礒田 治夫	(名古屋大学脳とこころの研究センター 教授)
梶谷 文彦	(川崎医療福祉大学 名誉教授)
高井 義美	(神戸大学大学院医学研究科 教授)
◎樋口 知之	(情報・システム研究機構 統計数理研究所 所長)

B) 開催状況

2018 年 12 月 26 日(水) 中間評価後の活動進捗および指摘事項の対応について報告 今後の方向性および成果ヒアリングを見据えたサブ課題の進捗と成 果に関する審議

(3) シンポジウムおよびワークショップの開催
 他機関と連携したシンポジウムやワークショップ等を開催した。
 平成 30 年度に行った活動概要は以下の通り(共催、協力、協賛、後援を含む)。

A) シンポジウム

次元俯瞰と攻略

(a) 名称:第3回その予防・医療、時代遅れです 未来はここでとっくにはじまっている
開催日:2019年1月18日(金)
場所:ホテル雅叙園東京参加者数:52名
主催:ポスト「京」重点課題2
協賛:「システムがん新次元」文部科学省科学研究費補助金新学術領域研究(複合領域 4701) がんシステムの新



図 4.1 シンポジウムの様子

内容:医療応用をめざし最先端の生命科学研究を行っている研究者とともに、計算科学と生命科 学の視点から、ポスト「京」コンピュータを用いた医療技術・予測医療の可能性を考えることを 目的としてシンポジウムを開催した。基調講演者に大阪大学国際医工情報センター三宅淳教授 を招聘し、「人工知能と医療の可能性」について、実際にデータを活用した研究内容のほか、人 工知能を用いた世界の最新の動向や成果に関する講演を行った。各サブ課題の研究開発状況に ついては、サブ課題A「スーパーコンピュータが照らし出すがんの多様性」、サブ課題B「空力 音響学からみた摩擦子音発音の仕組みと構音障害の予測・治療への取組み」、サブ課題C「心臓 病の解明・治療に貢献するUT-Heartー身近な病気から難病まで-」の講演を行った。アンケー ト回答では、「どれも大変有意義に拝聴させていただきました」「とても勉強させていただきまし た」等、満足度の高い結果を得た。

講演者:

三宅 淳(大阪大学国際医工情報センター 教授/サブ課題 B 課題協力者) 野崎 一徳(大阪大学歯学部附属病院 助教/サブ課題 B 課題参加者) 久田 俊明(株式会社 UT-Heart 研究所 代表取締役会長/サブ課題 C 課題責任者) 新井田 厚司(東京大学医科学研究所 助教/サブ課題 A 課題参加者)

- (b) 名称: 3nd International Symposium on Research and Education of Computational Science 開催日: 2018年9月20日(木)~21日(金)
  場所: Koshiba Hall、 School of Science、 The University of Tokyo
  主催: The Computational Science Alliance of the University of Tokyo
  協 替: Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures、 HPCI Consortium、 Post-K Computer Priority Issue 2-9、 RIKEN Center for Computational Science
- (c) 名称:第5回「京」を中核とする HPCI システム利用研究課題 成果報告会 開催日:2018年11月2日(金)
   場所:THE GRAND HALL(東京)
   主催:高度情報科学技術研究機構
共催:理化学研究所計算科学研究センター 協賛:HPCIコンソーシアム、スーパーコンピューティング技術産業応用協議会 協力:国立情報学研究所、他 12 機関 後援:計算科学振興財団、ポスト「京」重点課題実施機関

- (d) 名称: The 1st R-CCS International Symposium
  K and Post-K: Simulation、 BigData and AI supporting Society 5.0
  開催日: 2019年2月18日(月)~19日(火)
  場所: Room 301、 Kobe International Conference Center、 Kobe、 Japan
  主催: RIKEN Center for Computational Science
  後援: Ministry of Education、 Culture、 Sports、 Science and Technology、 Priority issues
  (9 issues) on Post-K computer、 HPCI Consortium、 Research Organization for Information
  Science and Technology、 FOCUS、 Artificial Intelligence Research Center、 AIST、 RIKEN
- (e) 名称:見える化シンポジウム 2019「バーチャルでリアルを超えろ~難解サイエンスを映像で感覚的に伝える~」
  開催日:2019年3月2日(土)
  場所:秋葉原 UDX シアター
  主催:ポスト「京」重点課題7次世代の産業を支える新機能デバイス・高性能材料の創成
  共催:ポスト「京」重点課題1、9、理化学研究所計算科学研究センター
  協力:豊橋技術科学大学、ポスト「京」重点課題実施機関、萌芽的課題「基礎科学の挑戦」、TIA
  かけはし「京算と計測のデータ同化による革新的物質材料解析手法の調査
- (f) 名称:第5回 大型実験施設とスーパーコンピュータとの連携利用シンポジウム
  開催日:2019年3月15日(金)
  場所:東京・秋葉原 UDX 4階 NEXT・2 および NEXT・3
  主催:高輝度光科学研究センター、総合科学研究機構、高度情報科学技術研究機構
  協賛:理化学研究所、ポスト「京」重点課題1~6、8、9、他9機関
  協力:ポスト「京」重点課題7

後援: 文部科学省、兵庫県、茨城県

- B) ワークショップ
- (a) 名称:ポスト「京」重点課題2ワークショップ
   開催日:2018年12月26日(水)
   場所:東京大学医科学研究所
   参加者数:34名
   主催:ポスト「京」重点課題2
   研究開発の進捗(変)および報告の想 またプロジェー



図 4.2 ワークショップの様子

研究開発の進捗確認および報告の場、またプロジェクト参加機関内での実質的な相互連携の推 進を目的として、プロジェクト参加研究者を対象にワークショップを開催した。参加研究者をは じめ諮問委員からの質疑応答では活発な議論が交わされた。

- (b) 名称: RIKEN International HPC Summer School FY2018
  開催日:2018年7月2日(月)~4日(水)
  場所:理化学研究所計算科学研究センター 6 階講堂
  後援:神戸大学計算科学教育センター、兵庫県立大学大学院シミュレーション学研究科協力:ポスト「京」重点課題9課題実施機関
- (c) 名称:2018年度理化学研究所神戸キャンパス一般公開開催日:2018年11月23日(土) 場所:理化学研究所計算科学研究センター 6階講堂一般来場者数:4、404名 内容:理研計算科学研究センター6F講堂に設けられた 重点課題の研究紹介や工作体験などを行う「神戸スパコ ンシミュレーション王国」にて、重点課題2の研究内容 を紹介したポスター展示と人体の臓器絵合わせカード や計算生命科学クロスワードパズル(日本語版・英語版)



図 4.3 ブースの様子

などの体験ゲームを実施した。生命科学への関心を促すため臓器絵合わせカードや解説資料を 提供した。重点課題2ブースへの来場者数は、昨年に引き続き約300名程度で大変好評であった。

(d) 名称: RIKEN R-CCS Youth Workshop
開催日: 2019年2月15日(金)~2月17日(日)
場所:理化学研究所計算科学研究センター 6 階講堂
主催:理化学研究所計算科学研究センター
後援:ポスト「京」重点課題9課題実施機関、計算科学振興財団、HPCI コンソーシアム、高度
情報科学技術研究機構

(e) 名称: KOBE HPC スプリングスクール(中級) 一並列計算をさらに勉強したい人に向けて一 開催日:2019年3月13日(水)~3月15日(金)
場所:兵庫県立大学神戸情報科学キャンパス、神戸大学計算科学教育センター
共催:神戸大学計算科学教育センター、兵庫県立大学大学院シミュレーション学研究科、理化学 研究所計算科学研究センター
後援:ポスト「京」重点課題9課題実施機関、高度情報科学技術研究機構、計算科学振興財団

(4) ホームページ (日本語・英語)

昨年度に引き続きサイトコンテンツの充実および拡充を行った。



図 4.4 日本語版



図 4.5 英語版

(5) ニュースレター、パンフレット発行(電子版)

ー般および企業等を対象として、研究開発の進捗や成果などを分かりやすく説明することで研究 活動の理解を深めるとともに、社会における重要性や価値の理解促進を図るため、昨年度に引き続き ニュースレターを発行した。



🗵 4.6 Vol.9, 10

図 4.7 Vol.11,12

🗵 4.8 Vol.13,14

## 表 4.1:研究報告「Research Report」について

Vol	タイトル	執筆者
0	心不全の解明と治療を目指すマルチスケール心臓シミュレーシ	久田 俊明、杉浦 清了、
9	ョン/サブ課題 C	鷲尾 巧、金田 亮
10	スーパーコンピュータが照らし出すがんの多様性/サブ課題 A	新井田 厚司
11	スパコンが拓く脳動脈瘤コイル塞栓術治療の個別化医療支援/	大谷 智仁、伊井 仁志、
11	サブ課題 B	和田 成生
19	Ut-Heart 活動便り 創薬を加速する心毒性スクリーニングシス	寺田 透、根上 樹、奥野
12	テムの開発/サブ課題C	恭史、久田 俊明
19	全脳循環シミュレータの開発〜脳血管の構造とモデル化〜/サ	和田 成生、伊井 仁志
19	ブ課題 B	
14	次世代のがん治療に向けた変異検出の高精度化/サブ課題 A	森山 卓也

- (6) セミナー、講習会等の開催
  - A) 講習会

「SiGN-BN による遺伝子ネットワーク解析実習」では、理論的な仕組みや遺伝子ネットワーク

のサイズに応じた三つのネットワーク構造推定アルゴリズムについて講義および実習を行った。 「Genomon2 Tutorial」では、がんのゲノム・トランスクリプトーム解析に興味のある医学生物 学系の研究者を対象に、がんの後天的変異・構造変異・融合遺伝子を検出する一般的原理、ゲノ ム解析に必要となる実践的なテクニックを紹介する講習会を行った。ハンズオンセッションで は、Genomon2 の体験実習を行い、より実践に即した解析方法を説明し、ゲノム解析の深い理 解に繋げることを目指した。

「Rの基礎、Rによる統計解析」では、統計解析言語 Rの入門に加え、遺伝子・タンパク質発 現データ等の実データを用いた統計解析、バイオインフォマティクスに関する説明と実習を行 った。「Rの基礎、Rによる統計解析」の応用編にあたる講習会として「Rによるバイオインフ ォマティクス(遺伝子セット解析、シングルセル解析等)」を開催した。

開催場所:東京大学医科学研究所

7月27日	遺伝子ネットワーク解析実	遺伝子ネットワーク推定ソフトウェア SiGN-BN の
2:00-5:00pm	羽 百	講義と実習
9月7日	Genomon2 Tutorial	がんゲノムシーケンス解析の原理と Genomon2 の紹
2:00-5:00pm		介
		Genomon2 ハンズオンセッション
11月29日-30	R の基礎、R による統計解	HGC のスパコンで利用可能な統計解析言語 R に
日 10:00am-	析(遺伝子発現解析等)	ついての講義と実習
5:00pm		
1月24-25日	R によるバイオインフォマ	HGC のスパコンで利用可能な統計解析言語 R に
10:00am- 5:00pm	ティクス(遺伝子セット解	ついての講義と実習
0 0 0 P	析、シングルセル解析等)	

表 4.2 平成 30 年度講習会の詳細

4-3. 活動(研究会・受賞・書籍等)

1. 研究会・一般向け講演会・セミナー等

発表した成果(発表題目、口頭・ポ	果(発表題目、口頭・ポ 発表者氏名		発表し	国内·
スター発表の別)	(所属機関)	(学会等名)	た時期	外の別
生体内現象の連続体シミュレーシ ョンと計測データを活用した逆解 析アプローチの紹介、招待講演	伊井 仁志(首都 大学東京)	第11回数理デザイン道場、 東京	2019 年1月	国内
個人別に異なる生体現象をコンピ ュータでどうやって再現する か?、招待講演	伊井 仁志(首都 大学東京)	大阪大学数理・データ科学 教育研究センター公開講座 「データサイエンスが切り 拓く生命科学・生体工学の 未来」、大阪	2018 年9月	国内

生体内の巨視的流動現象に対する 数値計算アプローチの今と未来、 招待講演	伊井 仁志(首都 大学東京)	日本機械学会バイオエンジ ニアリング部門若手講演交 流会、静岡	2018 年7月	国内
生体力学現象に対するデータ同化 アプローチの紹介、招待講演	伊井 仁志(首都 大学東京)	日本機械学会RC277第3回 分科会、北海道	2018 年7月	国内
生体物理現象の解明に向けた計算 科学アプローチの確立、招待講演	伊井 仁志(首都 大学東京)	TAMA協会「地域イノベ・医 療イノベーションフォーラ ム」、東京	2018 年7月	国内

2. 受賞等

友 升·	受賞者氏名(所属機	極當機問(登入友效)	受賞した時	国内・国	
名 你 	関)	仅貝機)(子云石寺)	期	際の別	
	武石直樹、Marco E.				
) 注日研究 :n 在今9019	Rosti、今井陽介、和	日本法体力学会年合	2018年9月	国内	
在日研先 III 平云2018	田 成 生 、 Luca	日本孤体刀子云平云	3日		
	Brandt				
注日研究 in 在今9018	吉永司、野崎一徳、	日本法体力受全在今	2018年9月	国内	
在百刎九 III 平云2018	和田成生	日本佩仲乃于云千云	3日		
紫綬褒章	小川誠司		2018年5月	国内	
奨励賞	大谷智仁	日本機械学会	2018年4月	国内	
平成30年度科学技術分野の文	杉浦洼了 (丹丰)	<b>立</b> 如利学学	2018年4日	国内	
部科学大臣表彰		又叫竹子百	2010年4月	1日1.1	

3. メディアへの情報発信、業界紙、企業誌、ウェブサイト等での情報公開

メディア名称	日付	見出し	備考	著者名と所属
信濃毎日新聞(夕刊)科学 する人18	2018年12         月 18日         (火)	がん患者ごとに最低な薬		宮野 悟、東大医科 研
胡口东明 97五利学の豆	2018年9月	「想定外」を考える AI		宮野 悟、東大医科
₩口利闻 21 国科子·07庫	17日 (月)	医療で手術ミス		研
三菱UFJモルガン・スタン レー証券	2018年8月	病気のない世界へ 躍進 する先進医療 Special Interview 次世 代医療への挑戦		宮野 悟、東大医科 研
<ul> <li>読売新聞 33面くらしサ</li> <li>イエンス サイエンス</li> <li>View</li> </ul>	2018年7月 8日(日)			宮野 悟、東大医科 研
週刊新潮	2018年5月	「AI&ゲノム」が一変さ	p42-45	宮野 悟、東大医科

	31日	せる「がん治療」の最前		研
		線特集		
		特集 AI の実用化と進展		
機関誌「Re」No.198 建築	2018.4	がもたらすもの	p24-27	宮野 悟、東大医科
保全センター		スパコン・AIで加速する		研
		がんの研究・治療		

## 4. 書籍、専門誌

タイトル	執筆者	書籍名	発行者	発刊時期、巻数
がんゲノム医療における人口 知能 がんの理解は人知を超え てしまった	宮野 悟、東 大医科研	BRAIN and NERVE.	医学書院	71(1):25-32、2019
特集:人工知能(AI)と腎臓 Watoson for Genomicsによる がんゲノム診断支援	宮野 悟、東 大医科研	腎臓	日本腎臓財団	Vol.40 2018、 p10- 14
東京大学医科学研究所におけ るがんの臨床シークエンスシ ステム研究の背景	宮野 悟、東 大医科研	遺伝子医学 MOOK34号	株式会社メディカ ルドゥ	2018年11月30日
多領域シーケンスとがんの進 化シミュレーション-大腸が んの腫瘍内不均一性の解析を 例に-	新井田厚 司、東大医 科研	遺伝子医学 MOOK33号	株式会社メディカ ルドゥ	2018年4月15日

【今年度計画と成果】

平成30年度業務計画は、プロジェクト全体の連携を密としつつ円滑に運営していくため、運営 委員会や研究会の開催等、参画各機関の連携・調整にあたる。特に、プロジェクト全体の進捗状況を 確認しつつ計画の合理化を検討し、必要に応じて調査或いは外部有識者を招聘して意見を聞くなど、 プロジェクトの推進に資する。またニュースレター(電子版)発行、ホームページ運営、シン ポジウム等の開催及び参加、人材育成活動、広報活動やアウトリーチ活動への参加などを 通じて研究の進捗と連携を推進する。

プロジェクトで得られた成果については、積極的に公表し、今後の展開に資する。

成果は、運営委員会を8回、人材育成のための講習会を4回主催し、広報活動やアウトリーチ活動の一環として9件以上のワークショップやシンポジウム等に参加し他重点課題各機関との連携・ 調整に努めた。外部有識者を外部諮問委員として招聘し意見を頂いた。またニュースレター(電 子版)を6本発行、ホームページを適宜最新化、各1回のシンポジウムとシンポジウムを開 催した。

- 4-4. 実施体制
- 1. 業務主任者
  - (受託者 (委託先))

役職・氏名 国立大学法人東京大学 医科学研究所 教授 宮野 悟

E-メールアドレス: miyano@ims.u-tokyo.ac.jp

TEL: 03-5449-5615 FAX: 03-5449-5442

(再委託先)

役職・氏名 国立大学法人大阪大学 大学院基礎工学研究科 教授 和田 成生

E-メールアドレス: shigeo@me.es.osaka-u.ac.jp

TEL&FAX : 06-6850-6170

(再委託先)

役職・氏名 株式会社 UT-Heart 研究所 代表取締役会長 久田 俊明

E-メールアドレス: hisada@ut-heart.com

TEL: 03-3410-0216 FAX: 03-3410-0216

(再委託先)

役職·氏名 国立大学法人京都大学 医学研究科 教授 小川 誠司

E-メールアドレス: sogawa-tky@umin.ac.jp

TEL: 075-753-9283 FAX: 075-753-9282

(再委託先)

役職・氏名 学校法人自治医科大学 学長 永井 良三

E-メールアドレス: rnagai@jichi.ac.jp

TEL: 0285-58-7006 FAX: 0285-44-5019

2. 業務項目別実施区分

業務項目	担当機関	担当責任者	
(1)	東京都港区白金台4-6-1	ヒトゲノム解析センター長	
大量シーケンスによるがんの	国立大学法人東京大学	宮野 悟	
個性と時間的・空間的多様	医科学研究所ヒトゲノム解析セ		
性・起源の解明(サブ課題	ンター		
A)			
	京都府京都市左京区吉田近衛町	大学院医学研究科教授	
	国立大学法人京都大学	小川 誠司	
	大学院医学研究科		
(2)	大阪府豊中市待兼山町1-3	大学院基礎工学研究科 教授	
データ同化生体シミュレーシ	国立大学法人大阪大学	和田 成生	
ョンによる個別化医療支援	大学院基礎工学研究科		
(サブ課題B)			
	東京都文京区本郷 7-3-1	大学院工学系研究科教授	

	国立大学法人東京大学	高木 周
	大学院工学系研究科	
	大阪府吹田市山田丘1-1	大学院情報科学研究科 教授
	国立大学法人大阪大学	松田 秀雄
	大学院情報科学研究科	
	大阪府豊中市待兼山町1-3	大学院基礎工学研究科教授
	国立大学法人大阪大学	野村 泰伸
	大学院基礎工学研究科	
	大阪府吹田市山田丘1-8	歯学部附属病院 准教授
	国立大学法人大阪大学	玉川 裕夫
	歯学部附属病院	
(3)	東京都世田谷区野沢3-25-8	代表取締役会長
心臓シミュレーションと分子	株式会社UT-Heart研究所	久田俊明
シミュレーションの融合によ		
る基礎医学と臨床医学の架橋	栃木県下野市薬師寺3311-1	学長 永井 良三
(サブ課題C)	学校法人自治医科大学	
(4)	東京都港区白金台4-6-1	ヒトゲノム解析センター長
プロジェクトの総合的推進	国立大学法人東京大学	宮野 悟
	医科学研究所ヒトゲノム解析セ	
	ンター	

3. 経理担当者

(受託者(委託先))

役職・氏名	国立大学法人東京大学	医科学研	究所 事	務部	研究支援課
	外部資金戦略チーム	主任 前田	幸子		

E-メールアドレス:t-gshikin@ims.u-tokyo.ac.jp

TEL: 03-5449-5138 FAX: 03-6409-2017

(再委託先)

役職・氏名 国立大学法人大阪大学 大学院基礎工学研究科 研究協力係 係長 武中 典子

E-メールアドレス: ki-kenkyukyoryoku@office.osaka-u.ac.jp

TEL: 06-6850-6142 FAX: 06-6850-6145

(再委託先)

役職・氏名 株式会社 UT-Heart 研究所 研究開発部 代表取締役会長 久田 俊明 E-メールアドレス: kida@ut-heart.com

TEL: 03-3410-0216 FAX: 03-3410-0216

(再委託先)

役職・氏名 国立大学法人京都大学 医学系研究科 医学・病院構内共通事務部 経理・研究協力課外部資金掛 掛長 芳倉 清紀

E-メールアドレス: a40gaishi@mail2.adm.kyoto-u.ac.jp

TEL: 075-753-9483 FAX: 075-753-4347

## (再委託先)

役職・氏名 学校法人自治医科大学 総務部総務経理課

経理第3係 係長 高巣 嘉子

E-メールアドレス:keiri3@jichi.ac.jp

TEL: 0285-58-7022 FAX: 0285-40-8014

- 知的財産権の帰属
   「知的財産権は乙に帰属することを希望する。」
- 5. 委託契約書の定めにより甲に提出することとされている著作物以外で委託業務により作成し、甲に 納入する著作物の有無
  - (有・(無))