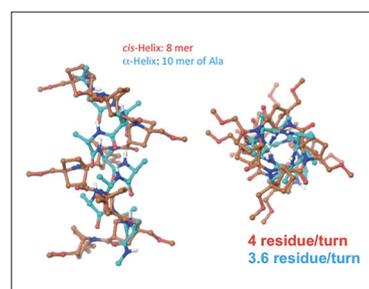
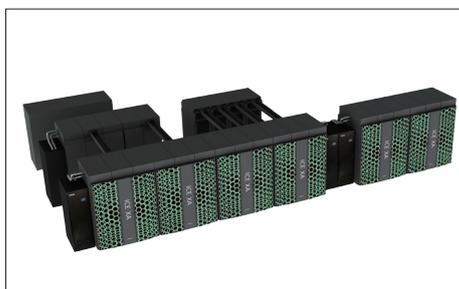


TSUBAME ESJ.



HPCとビッグデータ・AIを融合する グリーン・クラウドスパコンTSUBAME3.0の概要

Overview of Tsubame3.0, Green Cloud Supercomputer
for Convergence of HPC, AI and Big-Data

ChainerMN: スケーラブルな 分散深層学習フレームワーク

ChainerMN: Scalable Distributed Deep Learning Framework

創薬研究の基盤になる化学構造創出技術と 効率的GPUコンピューティングとの連携

Creation of Chemical Structures Aimed for Drug Design, Facilitated
by Efficient GPU Computing

HPCとビッグデータ・AIを融合する グリーン・クラウドスパコン TSUBAME3.0の概要

松岡 聡* 遠藤 敏夫* 額田 彰* 三浦 信一* 野村 哲弘* 佐藤 仁** 實本 英之*
Aleksandr Drozd*

*東京工業大学 学術国際情報センター **産業技術総合研究所 人工知能研究センター

TSUBAME3.0は半精度以上で47.2PFlopsの理論演算性能を持ち、従来型HPCに加えてビッグデータ・AIなど幅広いワークロードを処理する日本最大級のスパコンとなる。また、高密度化と省電力化を追求し理論PUE値1.033を実現する。本報告では、TSUBAME3.0のアーキテクチャを紹介し、いままでのTSUBAMEシリーズにおける課題とそれを解決するための運用方針について述べる。

はじめに

1

東京工業大学学術国際情報センター（以下、「本センター」という）では「みんなのスパコン」を合言葉とし、使いやすさと高性能を両立したスーパーコンピュータTSUBAMEシリーズを構築・運用している。2006年に導入されたTSUBAMEシリーズの初代であるTSUBAME1.0以来、TSUBAMEは学内外で約2,000名のユーザに利用されるシステムとなっている。2010年に導入されたTSUBAME2.0では、TSUBAME1.2で部分的に導入されたGPUアクセラレータを本格的に導入することで、消費電力を抑えながらも約30倍の性能向上を実現した^[1]。これによりGreenest Production Supercomputer in the Worldの認定を受けた。TOP500 4位（2010年11月時点）となる性能により、樹枝状凝固成長シミュレーションで2ペタフロップスを実現しACM Gordon Bell Prizeを受ける^[2]など、大規模実アプリケーションにおいても実績をあげている。2013年にTSUBAME2.0のGPU部をK20Xに更新したTSUBAME2.5では、理論最大性能が5.6PFに達した。

これらのTSUBAMEの後継機となるTSUBAME3.0を2017年8月より稼働を開始した。TSUBAME3.0は同年1月に日本SGI社による提案に決定し、本センターと日本SGI社および関連各社は協力し稼働開始に向けた準備を進めてきた。最終的なTSUBAME3.0の全体イメージを図1に示す。

TSUBAME3.0はこれまでのスパコンが対象としてきたHPC分野のみならず、ビッグデータやAIなどの処理基盤への活用を期待されている。SGI ICE XAを基にカスタマイズされた540台の計算ノードに搭載されるCPUは1,080基、GPUは2,160基となり、理論最大性能は倍精度で12.15ペタフロップス、半精度（以上）で47.2ペタフロップスになる。半精度は16bitで表現される浮動小数点数であり、ビッグデータやAIなどの分野で活用が期待されている。この半精度の演算はTSUBAME3.0に搭載するGPUアクセラレータでハードウェア実装されている。各計算ノードには容量2TBのNVMe対応SSDが搭載され、合計で1.08PBの容量は今後のビッグデータ解析基盤に活用されることが期待され

る。高速ストレージも容量15.9PB、データ転送速度150GB/sと強化されており、今後増大するビッグデータ処理基盤での活用が期待される。

最新のGPUを搭載するTSUBAME3.0は優れた省電力性を持っている。144台の計算ノードを使用してHPLベンチマークの性能測定を行った。電力効率に最適化したパラメータ設定では1.998PFLOPSの性能を141.6kWで達成し、14.11GFLOPS/Wattの電力効率で2017年6月のGreen500 List^[3]で世界第1位に認定された。

システムの冷却方式も効率化している。TSUBAME2.0/2.5の間接水冷およびTSUBAME-KFCの液浸・温水冷却等の経験より、冷却塔の使用による省電力性と、主要な熱源であるCPUとGPUのみを直接水冷、他のコンポーネントを間接水冷とすることによる高い安定性とメンテナンス性を実現する。

本稿では、TSUBAME3.0のシステム概要を示し、運用方針について報告する。

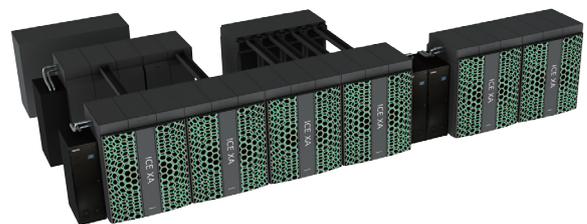


図1 TSUBAME3.0 全体イメージ

TSUBAME3.0の構成

2

2.1 計算ノード

TSUBAME3.0は540台の計算ノードがあり、その全ての構成は同一である。各ノードはTSUBAME2.0と同様に、最新のCPUやGPUアクセラレータ、大容量メインメモリ、高速なネット

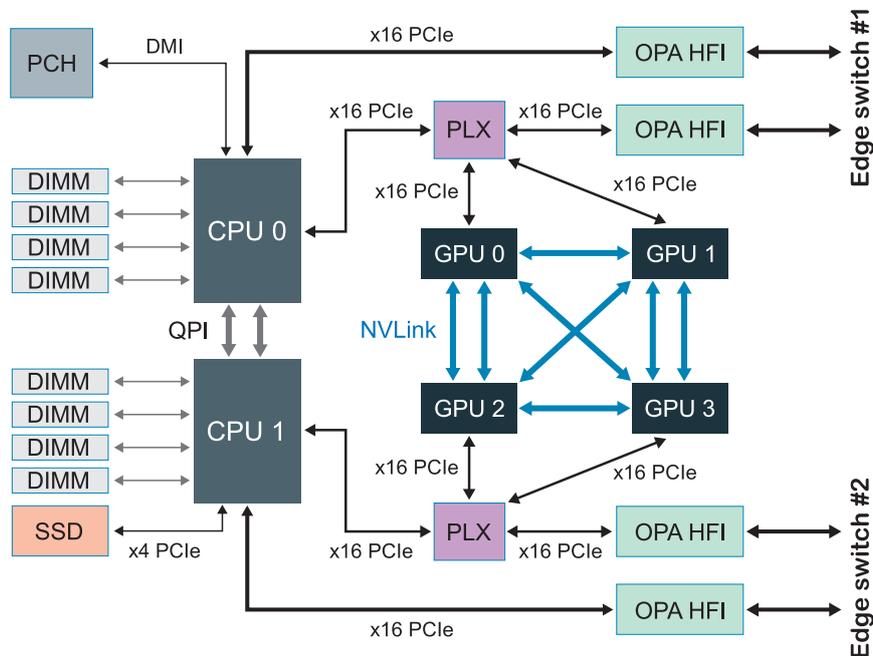


図2 計算ノードのブロックダイアグラム

ワークインターコネクト、高速なSSD等から構成される。CPUはIntel Xeon E5-2680 V4 Processor (Broadwell-EP、14コア、2.4GHz)を2基搭載し、DDR4-2400 ECC REG DIMM 32GBモジュール8枚により容量256GiB、バンド幅154GB/sとなる。GPUにはNVIDIA Tesla P100 for NVLink-Optimized Servers (16GB HBM2@732GB/s、5.3TFLOPS@FP64、10.6TFLOPS@FP32、21.2TFLOPS@FP16)を4基搭載する。なおTSUBAME1.2で導入したTesla S1070(GT200)、TSUBAME2.0のTesla M2050(Fermi)、TSUBAME2.5のTesla K20X(Kepler)に引き続き4世代目となる。ネットワークインターコネクトはIntel Omni-Path ArchitectureのHost Fabric Interface (HFI)を4基搭載し、ノードあたり片方向400Gbps、双方向800Gbpsのインジェクションバンド幅を持つ。各ノードには2TBの大容量SSDが搭載され、NVMeサポートにより高速なアクセスが可能である。特にネットワークインターコネクトとSSDはビッグデータやAI分野での処理の高速化に重要な要素である。

図2にTSUBAME3.0の計算ノードのブロック図を示す。ノード内の各プロセッサ数が偶数であり、上下でほぼ対称な構造である。PCI-ExpressインターフェイスはGen3であり、1レーンあたり片方向1GB/sの転送速度を持つ。Broadwell EPが持つPCI-Expressレーンは40であり、16レーンがOmni-Path HFIに、16レーンがPLXのPCI-Expressスイッチ経由で2基のGPUとOmni-Path HFIに接続される。残りの8レーンは4レーンずつ、SSDの接続に用いられる。2CPUで計4スロット装着可能であるが、そのうちの1スロットのみSSDが装着されている。

TSUBAME3.0に搭載されるTesla P100 GPUはSXM2モジュール

版であり、20GB/sのデータ転送速度を持つNVLinkをそれぞれ4リンク備え、GPU間の直接データ転送に使用される。

表1はTSUBAME2.5とTSUBAME3.0の計算ノードの比較である。値は全て理論値である。GPUは約4年前に更新されているにも関わらずGPU性能の進化の速さにより大きく向上しており、特に半精度性能が際立つ。容量およびバンド幅の系統はビッグデータ処理に重要であるため大幅に増強されている。

指標	T2.5	T3.0	倍率
CPU			
コア数 × 周波数 (GHz)	35.16	72.8	2.07
メモリ容量 (GiB)	54	256	4.74
メモリバンド幅 (GB/s)	64	153.6	2.40
GPU			
CUDA コア数	8064	14,336	1.78
FP64 (TFlops)	3.93	21.2	5.39
FP32 (TFlops)	11.85	42.4	3.58
FP16 (TFlops)	11.85	84.8	7.16
メモリ容量 (GiB)	18	64	3.56
SSD			
容量 (GB)	120	2,000	16.67
READ (MB/s)	550	2,700	4.91
WRITE (MB/s)	500	1,800	3.60
ネットワーク			
転送速度 (Gbps)	80	400	5.00

表1 TSUBAME2.5とTSUBAME3.0の計算ノードの比較。数値はノード内合計

2.2 インターコネクト

計算ノード、各種ストレージ、各種管理ノード等はIntel Omni-Path Architectureで構成されるネットワークで相互接続されている。特にバンド幅が要求される計算ノード群はフルバイセクションバンド幅Fat-Treeネットワークで接続されている。階層はEdgeスイッチとDirectorスイッチの2階層である。計算ノードは9台毎にシャーシに格納され、各シャーシにはEdgeスイッチを2台格納している。各Edgeスイッチは計算ノード9台からの内部リンクが計18本、さらに3台のDirectorスイッチへ計18本の光ファイバーのリンクが接続されている。

スーパーコンピュータにおける大規模計算では、ノードのインジェクションバンド幅だけではなく、システム全体のネットワークポロジが大きな影響を与える。TSUBAME3.0ではTSUBAME2.0に引き続き、フルバイセクション・ファットツリーと呼ばれるトポロジーを用いている。本トポロジーの特徴として、ネットワークに接続する全ポートの任意の組み合わせにおける通信路が理論的には重複なく接続することが可能であり、ネットワークの特定箇所の通信が他の通信の性能に影響を与えることないという点があり、アプリケーションおよび利用ノード数・通信形態が多様である大学のスパコンにとって有利な接続形態である。TSUBAME2.0/2.5では、通信経路表が固定であるスタティックルーティングを用いているため、実際には通信の組み合わせによってスイッチ間ネットワークの同一経路を重複して利用することによる性能劣化が見られた^[4]。TSUBAME3.0では、通信経路の混雑状況に応じて動的に通信経路を選択するアダプティブルーティングを用いるため、このような問題が解消されると期待している。



図3 TSUBAME3.0のグローバルストレージ

2.3 ストレージ

2.3.1 グローバルストレージ領域

スーパーコンピュータにとって、巨大なデータを蓄え、高速にアクセスできるストレージは必要不可欠な要素である。特にTSUBAME2.5では、全計算ノードから等しくアクセス可能なグローバルストレージ領域を備え、これがユーザの利便性に大きく寄与した。TSUBAME3.0でもこれを踏襲し、全計算ノードから等しくアクセス可能なグローバルストレージを有する。TSUBAME3.0では障害範囲の局所化と運用の柔軟性確保のため、実効容量約5.3PBのファイルシステムを3セット用意し、合計容量は約15.9PBに達する。実際に導入されたグローバルストレージを図3に示す。

2.3.2 スクラッチ領域

現在のスーパーコンピュータでは、ビッグデータ等のデータ処理において、大量のファイルI/Oリクエストが発行される。これらのI/Oリクエストを高速に処理できることは、現在のスーパーコンピュータのワークロードにおいて重要になっている。グローバルストレージを用いたこれらのI/Oは、大きなファイルサイズでなされることが一般的であるが、ワークロードの種類によっては、小さなファイルサイズのI/Oリクエストを大量に発行される場合がある。I/Oリクエスト数が増大すると、Lustreのような並列ファイルシステムではMDS性能が性能低下の要因となるとともに、これらの性能改善は難しい。これらの問題を解決するために、多くのスーパーコンピュータにおいてBurst Bufferと呼ばれるシステムが導入されている。しかしながら、現状のBurst Buffer技術はコストパフォーマンスに欠けると判断し、TSUBAME3.0での導入は見送ることとした。

以上の議論から、TSUBAME2.5までと同様に、多量のファイルI/Oを効率的に処理するため、各計算ノードに装備された2TB NVMe SSDをスクラッチ領域として提供している。しかしノード単体からしかアクセスできない領域のみでは、マルチノードにまたがったジョブに対しての利便性が不足している。それをカバーするために、BeeGFS^[5]と呼ばれる並列ファイルシステムを動的に生成(BeeOND^[6])する技術を導入している。ユーザジョブに割り当てられたノードにまたがったSSD上で並列ファイルシステムを提供することにより、ジョブ内からのみアクセス可能なグローバルスクラッチ領域を構築することとした。

ファシリティの整備

3

TSUBAME3.0の導入に当たっては、高密度・高効率を目標とし、スーパーコンピュータだけでなく、基本的なファシリティの再整備も必要になった。特に、スーパーコンピュータが整備される情報棟は現代のデータセンター設備としては貧弱であり、目標を達成することが困難であった。また、本学として初めて旧システムとなるTSUBAME2.5とTSUBAME3.0の2台のスーパーコンピュータの並行運用の可能性を検討しているため、設置スペースや電力事情に大きな制約が生じた。これらの問題を解決するために、情報棟の改修を行った。また、世界最高レベルの冷却効率による運用コストの低減を目指し、スーパーコンピュータ調達時において冷却設備の条件を厳しくした。

3.1 サーバルーム

近年のスーパーコンピュータは、性能向上を実現するために床面積の拡大を必要とする傾向にあった。一方TSUBAMEシリーズでは、システムの高密度化を進め床面積の削減に努めてきた。高密度化は単純な床面積の削減以外に、電力効率・冷却効率の向上にもつながる。特に今回TSUBAME2.5を保持したままTSUBAME3.0を整備するため、システム設置スペースの最小化は重要な課題であった。さらには、計算設備の高密度化に加えて、

後述するような冷却効率の向上を当初から目的としていた。このため、TSUBAME3.0では3.3節で述べるようにプロセッサを省エネな手法で直接液冷するシステムの導入を検討していた。このような方式は省エネ性では優れる一方で、パイプや冷却水そのもののためにラック重量が増加する傾向にある。

上記の検討の結果、1ラック当たりの積載重量が約1t/m²になりうる事が想定された。このような厳しい床耐荷重条件を満たす部屋が情報棟になかったことおよび、TSUBAME2.5の並行運用を予定していたことを踏まえ、TSUBAME2.5のストレージを設置していた部屋を以下の条件のもとサーバールームを改修した。

- 今後の積載荷重の増大に備え、部屋全体として100tの荷重に耐えうる床とする
- 床耐荷重の最大化とシステムコストの最適化のため、フリーアクセスフロアではなく、直接スラブの床とし、配線・配管を天井部に設置する
- 可能な限り空間を大きくするため、既存の天井部を撤去する

これらの改修を行い、図4に示す145m²ほどの床面積のサーバールームを整備した。最終的に、このような高密度化対策によって、ラックの重量による設置制限が大幅に緩和し、図5に示すように、TSUBAME3では既存のTSUBAME2.5の計算ノード設置スペースに対して半分以下の床面積でストレージを含むすべての資源を配置できたばかりでなく、将来的な拡張の余地を十分に確保することが可能になった。



図4 TSUBAME3.0のサーバールーム(左:施工前 右:施工後)

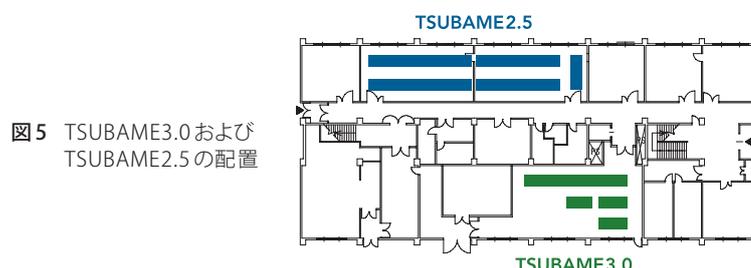


図5 TSUBAME3.0およびTSUBAME2.5の配置

3.2 電源設備

TSUBAME2よりもさらなる省エネ性を実現する一環として、TSUBAME3.0では配電電圧の高圧化を検討し、3相4線式の415Vもしくは3相3線式の480Vの2種類を検討した。一般的には、高電圧であれば配電口は小さく、かつ配線コストの低下につながる。しかしながら、導入する設備は高電圧仕様のスーパーコンピュータ本体だけでなく、管理用の一般的なサーバやネットワーク機器も導入する必要がある。そのため、単相電圧として240Vを容易に取り出すことができる3相4線式の415Vを選択した。単相240Vが利用できるのであれば、ほぼすべてのIT機器が対応しており、調達コストの低減を図ることができる。これらの検討を踏まえ、設備を施工する本学施設運営部との調整やトランス等の導入コストおよび将来の利用計画検討の結果、受電容量2MWの3相4線420Vの電源設備を整備することになった。最終的にTSUBAME3.0の消費電力はスーパーコンピュータ本体で1MW弱となり、電力面においても、3.1節に示したようにスペース面においても、将来の拡張に余裕が生じた。

3.3 冷却設備

スーパーコンピュータの運用コストを低減するためには、運用コストの大きな割合を占める電力使用量を削減する必要がある。このためにはスーパーコンピュータ本体の電力低減も重要であるが、冷却設備の電力低減も重要になる。例えば、TSUBAME2における年平均PUEは1.28であり、この場合全電力に占める冷却等電力は、約22%にもなる。TSUBAME2においては年間電力利用料金が1億円以上となるため、2,000万円以上が冷却で消費されているといえる。

この課題に向け本センターでは、TSUBAME2で導入した水冷・空冷ハイブリッドシステムの運用や、TSUBAME-KFCでの油浸冷却などの省エネ冷却技術の検証を長年行ってきた^[7]。特に下記のような技術検討を行った：TSUBAME2のような空冷も用いる手法の場合、冷却水を低温（10℃前後）とする必要があり、そのためのチラーが冷却にかかる消費電力を押し上げていた。TSUBAME-KFCのような直接液冷により、冷却液の温度をより高温（30℃以上）とし、消費電力削減が可能である。一方、サーバ全体を冷却液に沈める方式では、特に大規模運用時において、設置面積・メンテナンス性などの課題が大きい。

以上を踏まえ、TSUBAME3.0の冷却においては、主要発熱源であるCPUおよびGPUについては、空気を介さない直接液冷とした。計算ノードの内部を冷却水パイプが通る方式である。冷却水の冷却のためには、クーリングタワーを用いたフリークーリングを年間を通じて行う。以上によりTSUBAME3.0の理論的な想定PUEは1.033となり、TSUBAME2で約22%を占めた冷却電力が、3%程度へと大きく削減される見込みである。

運用

4

TSUBAMEシリーズは、これまでパソコンや小規模クラスターで計算をしていたユーザも、気軽にスパコンを利用できる可用性の実現を目標としてきた。TSUBAME3.0においても、これまでの運用を継承しつつ、さらに可用性を高めるため、ジョブスケジューラに基づく単純な計算資源提供に加え、以下のようなサービスの提供を行う。

4.1 コンテナ技術を用いたノードの動的な分割

TSUBAME2.0/2.5では全体の約1/3のノードにおいてVM技術を用いて計算ノードをGPU部分とCPU部分に分割し、それぞれを仮想的に別の計算ノードとして資源提供することで、GPUを中心に利用しCPUはさほど利用しないユーザや、GPUを利用せずCPUのみを利用するユーザの需要に応えてきた。しかしながら、VMおよびスケジューラの資源分割が固定であるため、利用者の需要と合致しない資源が一時的に遊休化するなど、資源利用率の面で課題があった。また、当時のVM技術の制約上、性能を犠牲にすることなくGPUを仮想化することが出来なかったため、GPUジョブはベアメタルノードで実行しつつ、そのノードの中にCPUジョブのためのVMを作成する形をとる必要があり、ノード内の3基のGPUを分割することはできなかった。

TSUBAME3.0では、コンテナ技術のひとつであるDockerを用いて、すべての計算ジョブをコンテナの内部で起動することを計画している。DockerコンテナからはGPUおよびOmni-Path HFIに直接アクセスすることが可能であり、またコンテナから利用できるデバイスを制限することもできるため、TSUBAME2.5よりも柔軟に資源の分割が可能である。例えば図6に示すように、ノード内の一部のGPUのみを切り出すようなノードの分割を行うことができる。さらに、スケジューラと連携してコンテナの作成を動的に行うことにより、TSUBAME2.5で行われていた静的なパーティショニングは不要であり、需要に応じたノードの切り出しが行えるために、システム全体の計算資源が有効に活用可能となる。

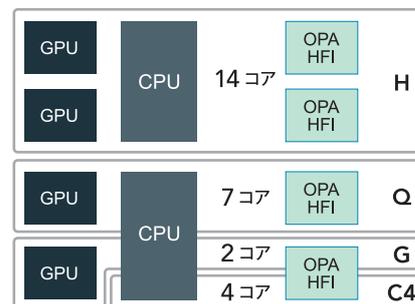


図6
TSUBAME3.0
計算ノードの
資源分割イメージ

4.2 資源のより効率的な分配

TSUBAME2.0/2.5を運用するうえで、継続的な運用改善にも関わらず、計算資源の効率的な利用を阻害するいくつかの要因が残っていることが分かってきた。その一つは上述の固定的なノード分割であり、TSUBAME3.0における改善についてはすでに述べた。

ほかにも、予約利用の利用方法が柔軟性にかけることが分かってきた。予約方式とは、通常のジョブスケジューラによる計算資源利用のほかに、ユーザが前もって日時を指定して利用する方式である。TSUBAME2では予約単位が1日(10:00～翌9:00)であったものを、TSUBAME3.0では1時間単位とする(本稿執筆時点では予約利用方式は運用準備中)。

また、すでにTSUBAME2.5の運用期間中に、ジョブスケジューラのバックフィリングをより効率的に行うための運用改善を行った^[8]。これはジョブの実消費時間だけでなく、ジョブ投入時に用いた最長実行時間にも一部課金を行うものである。この方式を継続して用いることで、資源の効率的な分配につとめる。

またTSUBAME2と異なり、システム中の540台の計算ノードを役割によって固定に分割することを原則行わないなど、ユーザの利用形態やシステム負荷状況にあわせた利用がより簡便となるようにする。

4.3 コンテナ技術を用いたシステムイメージの分離

TSUBAME2.0/2.5は当初計画を上回り、6年以上にわたって運用が行われてきた。計算ノードで動作するOSディストリビューションはSUSE Enterprise Linux 11 SP3(2013年に更新)であり、2017年現在ではカーネルおよび各種ライブラリのバージョンが古く、TSUBAME標準のコンパイラ・ツールや各種ライブラリではビルドできないアプリケーションがAI分野を中心に出現しており、個別に新しいバージョンのソフトウェアをインストールして対応する^[9]ことをユーザに強いている。また、個別にセキュリティパッチ適用等は当然行ってはいるものの、システムイメージを古い状態に保つことはセキュリティの観点からも望ましくない。システムイメージを頻繁に更新できない要因は複数あるが、システムライブラリ更新に伴うアプリケーションの互換性喪失、特にISVアプリケーションの動作検証コストが膨大となることがあげられる。

TSUBAME3.0では、上述のコンテナ技術Dockerを用いることで計算ジョブ毎に用いられるシステムイメージと、TSUBAME3.0本体のシステムイメージを分離することができる(本稿執筆時点では運用準備中)。このため、検証済みのISVアプリケーション実行用イメージや、最新のライブラリがインストールされたイメージなど、ユーザの需要に応じたシステムイメージをジョブごとに選択して実行することができるようになる。また、過去のシステムイメージを保存することにより、ソフトウェアの更新が原因で再現実験が困難になるという問題も解消できる。セ

キュリティの問題から、ユーザが作成したDockerイメージを直接実行することは難しいが、本センターよりユーザの需要に合致するシステムイメージを複数提供することを計画している。

4.4 計算ノード・ストレージの外部ネットワークとの接続

TSUBAME2.5までは、計算ノードが外部ネットワークから隔離されており、必要な実験データやプログラムは基本的に外部からSFTP等で送り込む必要があった。TSUBAME3.0では、計算ノードからの外部ネットワークへの接続を選択的に許可し、SINET5に100Gbpsで接続することにより、外部のストレージサービスや計算資源との連携をより行いやすくする。

またこれらのネットワーク帯域を有効に活用し、TSUBAME3.0のグローバルストレージをNFSもしくはCIFSにより外部に直接公開することを検討している。

4.5 モニタリング情報の公開

TSUBAME2.5に引き続き、スケジューラ・ノード・電力のモニタリング情報や障害情報の公開^[10]を継続し、「見える化」を推進する。障害情報については、現在は人手による更新でCSVデータの提供を行っているものの、表記ゆれや非構造的記述により機械的・統計的処理には向かないデータであるという問題がある。TSUBAME3ではモニタリング情報や障害情報の統計的処理ができるよう、開示情報の構造化・オープンデータ化に努める。またユーザのジョブに課する情報についても、TSUBAMEポータルを通じて積極的に提供する。

4.6 アカウント申請のペーパーレス化・全国共同利用

学内ユーザに対し、TSUBAMEが提供するポータルwebページ(以下TSUBAMEポータル)を東工大ポータル(全学のポータルwebページ)と連携させたペーパーレスのアカウント申請などのサービスをTSUBAME2.5と同様に行う。また、国内の代表的な9つの大学計算機センターの一つとして、HPCIなどを通じて大学を縦断したアカウントサービスを引き続き提供する。

TSUBAME2.5においては、アカウントの属性(学内利用・TSUBAME共同利用・HPCIによる利用など)に応じて1個人に複数のアカウントを発行していたため、アカウントの使い分けをユーザに強いていたが、TSUBAME3.0では1個人1アカウントとして同一アカウントからそれぞれの属性に応じた利用を可能とする予定である。これにより実験のためのデータをアカウントごととストレージに配置するなどの不便が解消される見込みである。

4.7 グランドチャレンジ制度

TSUBAME2.0/2.5では、ピアレビューにより選ばれたユーザグループ（学内・学外）に計算資源を提供するTSUBAMEグランドチャレンジ大規模計算制度^[1]を運用し、これまでに延べ20のグループに対してTSUBAME2の全計算ノードを12～24時間にわたって提供、14のグループに対してTSUBAME2の1/3の計算ノードを1週間にわたって提供した。TSUBAME3においても同様の大規模実行機会を設ける予定である。

おわりに

5

本稿では、2017年8月に稼働開始した、TSUBAME3.0の概要について示した。TSUBAME3.0は、TSUBAME2.5で目指した低消費電力やクラウド技術の統合を踏襲しつつ、既存のHPC分野に留まらない、ビッグデータ解析やAIなどの応用分野にも利用できるシステムを目指している。

謝辞

TSUBAME3.0の設計には、東京工業大学学術国際情報センターが推進してきた文部科学省「スパコン・クラウド情報基盤におけるウルトラグリーン化技術」および「スマートコミュニティ実現のためのスパコン・クラウド情報基盤のエネルギー最適化の研究推進」、JST CREST (JPMJCR1303, JPMJCR1501)などのプロジェクトの研究成果が活用されている。

参考文献

- [1] 松岡聡、遠藤敏夫、丸山直也、佐藤仁、滝澤真一朗：TSUBAME 2.0の全貌。東京工業大学学術国際情報センター、TSUBAME e-Science Journal, No.1, pp. 2-4 (2010)
- [2] T. Shimokawabe, T. Aoki, T. Takaki, A. Yamanaka, A. Nukada, T. Endo, N. Maruyama, S. Matsuoka. Peta-scale Phase-Field Simulation for Dendritic Solidification on the TSUBAME 2.0 Supercomputer. IEEE/ACM SC11, 11 pages (2011)
- [3] TOP500 Supercomputer Sites: Green500 List for June 2017, <https://www.top500.org/green500/lists/2017/06/>
- [4] 野村哲弘、遠藤敏夫、松岡聡：TSUBAME2.0におけるMulti-rail InfiniBand ネットワークの性能評価、情報処理学会研究報告、Vol. 2012-HPC-137, No. 3, pp. 1-5 (2012)
- [5] Fraunhofer Center: BeeGFS - The Leading Parallel Cluster File System, <https://www.beegfs.io/>
- [6] Fraunhofer Center: BeeOND: BeeGFS On Demand,

<https://www.beegfs.io/wiki/BeeOND>

- [7] T. Endo, A. Nukada, S. Matsuoka. TSUBAME-KFC: a Modern Liquid Submersion Cooling Prototype towards Exascale Becoming the Greenest Supercomputer in the World. IEEE ICPADS 2014, pp.360-367 (2014)
- [8] 野村哲弘、佐々木淳、三浦信一、遠藤敏夫、松岡聡：TSUBAME2におけるスケジュール効率化への取り組みとユーザ動向の見える化、情報処理学会研究報告、Vol. 2015 HPC-150, No. 2, pp. 1-7 (2015)
- [9] 東京工業大学学術国際情報センター：TSUBAME2.5 実験的サービス
<http://tsubame.gsic.titech.ac.jp/labs>
- [10] 東京工業大学学術国際情報センター：TSUBAME2.5 統合監視ポータル
<http://mon.g.gsic.titech.ac.jp/>
- [11] 東京工業大学学術国際情報センター：TSUBAME グランドチャレンジ大規模計算制度
<http://www.gsic.titech.ac.jp/GrandChallenge>

ChainerMN：スケーラブルな分散深層学習フレームワーク

秋葉拓哉 福田圭祐

Preferred Networks, Inc.

深層学習がブレイクスルーとなった鍵の一つは、GPU を中心とした高性能な計算能力の活用である。分散処理によって更なる計算能力の活用を可能とすることは、現在の深層学習を高速化・大規模化するだけでなく、未解決の課題に挑戦していくためにも重要である。

本稿では、深層学習の分散処理に焦点を当て、我々の開発した分散深層学習フレームワーク ChainerMN の設計、実装及び評価を解説する。

はじめに

1

画像認識、自然言語処理、音声認識など、機械学習が応用される様々な分野において、深層学習が既存の手法を遥かに凌駕する予測性能を達成することが分かってきた。深層学習で用いるニューラルネットワークの基礎技術は非常に長い歴史を持ち、その発端は 1950 年代まで遡ることができる。長い歴史を持つニューラルネットワークの技術が、2010 年代に入り「深層学習」として上述のようなブレイクスルーを起こすことのできたのは、アルゴリズム、大規模データ、高性能な計算能力の 3 点の進歩全てを組み合わせることに成功したからであると考えられている。そして今日でも、この 3 点のうち 1 つでも欠いてしまうと、深層学習によって優れた予測性能を達成するのは難しい。本稿では、深層学習を支えるこれら 3 つの柱の 1 つである、計算能力の部分に注目する。

計算能力の高い GPU を訓練時に利用することは、深層学習の多くのタスクにおいて今や標準的になっている。しかし、GPU の進歩と共にニューラルネットワークのアーキテクチャが大規模化・複雑化していることもあり、例え最新の GPU を利用しても、未だに深層学習の訓練は長い時間がかかってしまう。訓練に長い時間がかかってしまうと、高い精度を達成するためのモデルやパラメータの試行錯誤の回数が限られてしまう。また、活用可能なデータサイズにも限界が出来てしまう。そこで、多数の GPU を並列に利用することによって計算を加速することが不可欠である。

我々は、かつて深層学習の創薬分野への応用を目的とし、Neural Fingerprint^[1]とその改良手法を用いた化合物の表現学習を、TSUBAME 産業利用トライアルユース採択課題（平成 27 年度）として実施した。機械学習の問題としては活性の有無の 2 値分類に帰着される。1 タスクあたり 10-20 時間と、CNN を用いた典型的な一般物体認識モデルの学習に比べれば短い、タスクの数が 72 個あったため、ハイパーパラメータチューニングを含めると非常に高い計算能力が求められた。それが本課題を

TSUBAME2 の課題として実施した理由の 1 つである。72 個のタスクは独立していたため、単純な並列実行で実験を実施できた。実験の詳細に関してはトライアルユース報告書を参照されたい。

一方で、独立ではない単一の学習をより高速にするための研究や実装も行ってきた。本稿で解説する ChainerMN もそのような取り組みの 1 つである。ChainerMN は深層学習フレームワーク Chainer の追加パッケージとして開発されており、インストールすることで Chainer に分散学習の機能を追加する。ChainerMN の開発において重視した特徴は以下の通りである。

1. **柔軟である**：Chainer は define-by-run アプローチに基づく柔軟性の高いフレームワークであり、ChainerMN もその柔軟性を損なわないように設計されている。そのため、動的ニューラルネットワーク、敵対的生成ネットワーク (GAN)、深層強化学習のような複雑なユースケースであっても簡単に分散学習を行うことができる。
2. **高速である**：深層学習における現実的なワークロードを最初から想定して利用技術を選択し設計を行うと共に、ハードウェアの性能をフル活用するための実装上の工夫を行っており、後述の通り他フレームワークとの比較実験にて最も高い性能を達成した。

本稿では、ChainerMN について解説する。まず、分散深層学習の基礎について解説し、次に ChainerMN の実装を解説する。最後に評価実験の結果を提示する。

分散深層学習の基礎

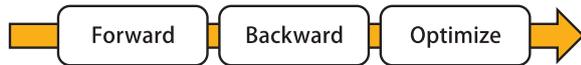
2

2.1 深層学習

ニューラルネットワークのパラメータを θ とおき、入力データ x に対するニューラルネットワークによる予測を $f(x; \theta)$ と表記する。

誤差逆伝播法と確率的勾配法 (SGD) 及びその派生アルゴリズムを用いた場合のニューラルネットワークの学習は、反復アルゴリズムによって行われる。1 反復は以下の 3 つのステップから成る。

1. 前向き計算
2. 後向き計算
3. 最適化



前向き計算のステップでは、まず入力データ x に対する予測 $f(x; \theta)$ を計算する。そして、 x に対する正解データ y との不一致の具合を表現する損失 E を計算する。ここでは、例えば交差エントロピー等の指標が用いられる。

後向き計算のステップでは、損失 E を減少させる方向へのパラメータ θ の勾配 $g = \frac{\partial E}{\partial \theta}$ を求める。出力の層から入力層へ向かって遡りながら、連鎖率を用いて全パラメータについての勾配を求めていく。

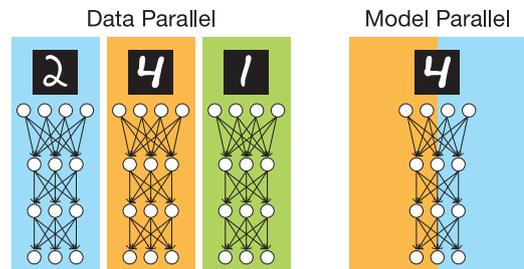
最適化のステップでは、勾配 g を用いて、パラメータ θ を更新する。最もシンプルな更新ルールは、 θ を $\theta - \eta g$ に更新するものである。ここで η は学習率と呼ばれるパラメータである。

実際には、1 回の反復で 1 つの学習データを用いるのではなく、複数の学習データに対して前向き計算・後向き計算を同時にを行い、それらの全データに対する勾配の平均を用いて最適化を行う。1 回の反復で用いる入力データの集合をミニバッチと呼び、そのサイズをバッチサイズと言う。バッチサイズは通常数十から数百である。

なお、以上で説明したのは標準的な教師有り学習の場合である。教師無し学習、半教師有り学習などの異なる枠組みにニューラルネットワークを適用する場合でも、以下で説明する並列化手法は適用可能であり、ChainerMN も利用可能である。

2.2 2つの並列化アプローチ

深層学習を分散処理によって高速化するアプローチは、大きく分けて、データ並列とモデル並列の 2 つがある。データ並列のアプローチでは、各ワーカーがモデルのレプリカを持ち、異なるミニバッチに対する勾配を計算する。それらの勾配を用い、ワーカーは協力してモデルをアップデートしていく。モデル並列のアプローチでは、各ワーカーがモデルの一部を持ち、1 つのミニバッチに対する計算を協力して行う。

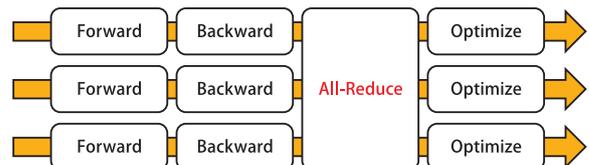


モデル並列のアプローチは、GPU のメモリが小さかった少し前の時代に活発に利用されていた。現在では、モデル並列の基本的な形での利用は少なく、データ並列のアプローチが一般的に用いられている。一方で、データ並列のアプローチの課題についても分かってきており、モデル並列に関しても新しい形態についての研究が進行している。また、モデル並列とデータ並列は同時に利用することも可能である。

2.3 データ並列における同期型と非同期型

以下では、現在一般的に用いられているデータ並列のアプローチに注目していく。データ並列には大きく分けて同期型と非同期型の 2 つがあるが、ここでは同期型についてまず説明する。同期型のデータ並列の分散学習における 1 反復は、以下の 4 ステップから成る。

1. 前向き計算
2. 後向き計算
3. 通信
4. 最適化



最初に紹介した通常時の 1 反復に対し新たに「通信」のステップが加わっている。このステップではワーカー同士が通信を行い、全ワーカーが独自に計算した勾配の平均を計算し、その平均を分配する。全ワーカーは通信によって手にした勾配を用いて

モデルを更新する。各ワーカーが処理するバッチサイズを b とし、ワーカーの数を n とすると、通信によって得る勾配はバッチサイズ bn での勾配に相当する。分散処理により、1イテレーションでより多くの訓練データに対する勾配を計算したこととなり、勾配の品質が向上し、学習が高速化する。

一方、非同期型では、通常のワーカーに加え、パラメータサーバと呼ばれる特別なワーカーを用いる。パラメータサーバはモデルのパラメータを管理する。通常のワーカーは前向き計算・後ろ向き計算により勾配を得次第、勾配をパラメータサーバに送付する。パラメータサーバは受け取った勾配を用いてモデルの更新を行う。ワーカーは新たなモデルのパラメータを受け取り、また勾配の計算を行う。

ChainerMN の実装

3

3.1 APIデザイン

Chainerは、define-by-runを特徴としているフレームワークである。define-by-runとは、スクリプト言語の柔軟性を活用し、実行時に学習モデルと計算フローが定義されるモデルである。一方でdefine-and-runと呼ばれるアプローチでは、予めネットワークの構造を定義し、その後でデータを入力して計算を行うモデルで、潜在的に性能最適化がしやすい一方で、柔軟性に欠けると言われている。

Chainerは、define-by-runアプローチにより、複雑なニューラルネットワークを柔軟に定義したり、実行中に変更を行うことも可能であるプログラミングモデルを提供している。これにより研究者や技術者が新しいモデルや複雑なモデルを容易に試行錯誤でき、機械学習の研究開発に適したフレームワークであるといえる。

開発にあたっては、既存のChainerプログラムからの移植が容易であること、Chainerの柔軟性に制限を加えないことを目標にAPIを設計した。

ChainerMNは、Optimizerと呼ばれる部分に追加コンポーネントを加えることによって既存のChainerプログラムに分散機能を追加する仕組みとなっている。それに加えてデータ並列のためにデータを配分するScatter処理を加えるだけで、基本的な移植が可能となっている。それ以外の部分(Iterator、Updater、Evaluator)は、基本的なユースケースでは変更は不要である。このAPI設計により、様々なChainerプログラムをdefine-by-runの利点を活かしつつ少ない手間で移植できるようになっている。

3.2 データ並列・同期型の並列計算モデル

現在のChainerMNは、データ並列・同期型の計算モデルを採用している。それぞれの計算モデルについては既に述べた。

まず、既存の深層学習アプリケーションを容易に拡張可能であり、並列化による学習の高速化が高い確度で見込める利点があることから、データ並列アプローチを採用した。データ並列は、大まかに言って通常の深層学習アプリケーションのミニバッチサイズを増加させることに相当し、既存プログラムのアルゴリズム・コードを大きく変更することなしに適用可能であるという利点がある。ただし、後に述べるように、学習されたモデルの精度については注意が必要である。

また、同期型と非同期型については、実装ごとに様々な戦略が取られてきており、タスクや設定によって結果が変わってくるため、どちらが望ましいかは自明ではない。文献^[3]では、非同期型は収束に対する安定性が低く、同期型のほうが早く収束するという実験結果が示されている。また、非同期型モデルには一般にパラメータサーバを用いた実装方式が用いられる一方、同期型ではいわゆるAllreduce通信が利用可能であり、MPIの最適化された実績のある集団通信メカニズムの恩恵を受けることができる利点もある。これらの点を踏まえ、我々はChainerMNにおいてまずは同期型を採用することとした。

3.3 実装と最適化

執筆時点でのChainerMNは、データ並列・同期型モデルを採用しているため、HPCアプリケーションとしての通信パターンは比較的単純である。大きく分けて、学習や評価の結果である勾配を交換するための主要部分であるAllreduce処理と、学習に先立って必要なデータの配置等を行うためのScatterなどの補助部分がある。

Allreduce部分は、学習のイテレーションごとに呼び出され大量のデータを処理する必要があることから、特に速度が要求される部分である。我々は、MPIのみを用いるのではなく、NVIDIA社が開発しているNCCLライブラリを併用することで速度向上を図った。NCCLはノード内のNVIDIA製GPU間での高速なAllreduce処理などを提供する、高度に最適化された通信ライブラリである。ノード内のGPU間通信を行うNCCLと、ノード間の通信を行うMPIを階層的に組み合わせて利用することによって、より高速なAllreduce処理を実装することができた。

(注：本稿執筆中にNCCLライブラリバージョン2がリリースされた。NCCL2ライブラリはInfinBandを用いたノード間通信もサポートしており、MPIによらず高度に最適化されたAllreduce通信を実現できる。)

評価

4

4.1 評価における留意点

分散深層学習の難しさの1つとして、スループットの向上が常にそのまま学習効率の向上になるわけではない、という点が挙げられる。例えばデータ並列のアプローチでは、GPU数を増やすほどバッチサイズが大きくなるが、ある程度以上バッチサイズを大きくすると学習に悪影響があり、結果として得られるモデルの精度が徐々に下がっていく。この理由としては2点が指摘されている。まず、同じエポック数（即ち、各学習データを同じ回数利用する）の学習を行う場合、イテレーション数が減ってしまうため、モデルが成熟しないことがある。加えて、勾配の分散が小さくなることにより、性質の悪い局所解（sharp minima と呼ばれる）に進みやすくなってしまい、結果として得られるモデルの汎化性能が悪くなってしまふ、という現象も知られている^[2]。

このような分散深層学習の課題を考慮せずにスループットのみを報告するベンチマーク結果は意味が無いと言える。通信パターンは単純なAllreduceが主であるため、バッチサイズを上げたり同期頻度を下げたりすることにより表面上は高い性能とスケーラビリティを得ることは容易である。しかし、そういった設定では有用な学習結果を得ることはできない。今回も、GPUのメモリにはかなり余裕がある状況ながら、各GPUの担当バッチサイズを小さめに抑えることで、十分な精度のモデルを得ることができる設定としている。

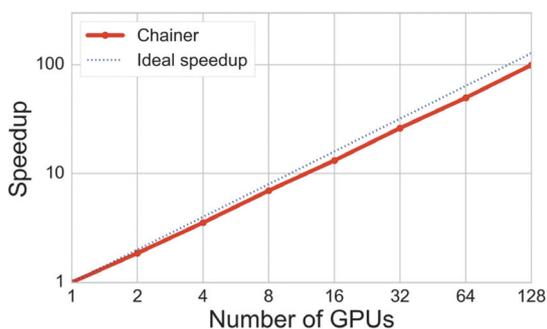


図1 Training Speedup for ImageNet classification (ResNet-50)

4.2 評価結果

図1は、画像認識のベンチマークとして広く用いられているImageNetデータセットを、ResNet-50^[4]と呼ばれるモデルで学習したときの性能評価である。評価は、ChainerMN、MXNet、CNTK、TensorFlowについて1~128GPUの環境上で比較した。詳細な実験の設定と、本稿に掲載していない図表については我々が公開しているブログ記事^[5]で述べているので参照されたい。Chainerは、1GPUではMXNetおよびCNTKに劣る性能となっている一方、4GPU以上では最速となった。ChainerはPythonで書かれたdefine-by-runスタイルのフレームワークであるため、CPUにおけるCUDAカーネル発行やその他の処理がボトルネックとなりやすく、C++で記述されたMXNet、CNTKよりも不利だと考えられる。しかし、NCCLとMPIを組み合わせた階層的な通信によって、4GPU以上ではChainerが最速となっている。

なお、TensorFlowがこのような低い評価結果となったのは、gRPCを用いたクライアント・サーバモデルを採用しているという点が理由として考えられる。クライアント・サーバモデルは、集団通信のAllreduceと比較してオーバーヘッドが多く、また実験の時点でgRPCには巨大なメッセージで性能が低下するという現象が確認されていた^[6]。

図2は、ChainerMNを用いて分散学習を行った時の計算時間と精度を示しており、8GPU時と比較して128GPUでも精度を維持したまま学習を高速化できていることがわかる。これは、我々の実験設定が、スケーラビリティを優先した病的な実験設定ではなく、機械学習的な側面から見て健全な実験設定になっていることを示している。

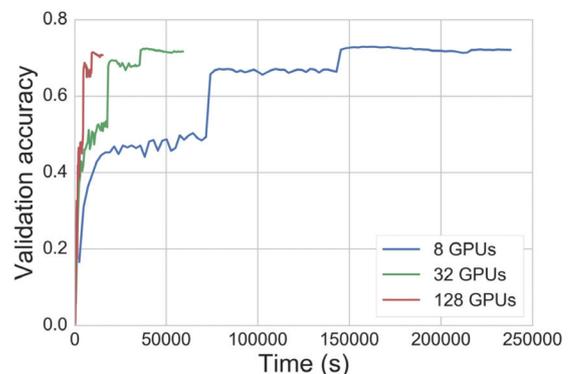


図2 Achieved Accuracy for ImageNet classification

本稿では、分散深層学習の基礎と ChainerMN の設計、実装及び評価について解説した。Chainer と ChainerMN は深層学習に関する研究開発を加速させることを主眼に開発しており、高い柔軟性とスケーラビリティの両方を兼ね備える。今後は、モデル並列、通信と計算のオーバーラップ、ワーカー間の非同期な計算、勾配の圧縮による通信効率化、耐障害性などの課題に取り組み、改善を続けていく予定である。

深層学習は既に複数の応用分野で欠かすことの出来ない存在となっている一方、現在も急速に進歩を続ける技術であり、国際的な研究開発競争は激化の一途を辿っている。深層学習の研究開発を効率的に進めるためには、時に数TBにも及ぶ学習データを保存・活用できるストレージ、膨大な線形代数演算を高速化するアクセラレーター、そして巨大な勾配データを交換するための低遅延かつ広帯域なネットワークなどの強力な計算基盤が不可欠である。

ビッグデータスパコンとして世界トップの計算効率、トップクラスの計算性能、高速なインターコネクト、強力なネットワークファイルシステムとローカルストレージを有するTSUBAME3.0とChainer/ChainerMNを組み合わせることによって、世界での研究開発競争において大きなアドバンテージを有することができると考えている。

参考文献

- [1] Dahl, G. E., Jaitly, N., & Salakhutdinov, R. (2014). Multi-task neural networks for QSAR predictions. arXiv preprint arXiv:1406.1231
- [2] Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., & Tang, P. T. P.. On large-batch training for deep learning: Generalization gap and sharp minima. ICLR'17.
- [3] Pan, X., Chen, J., Monga, R., Bengio, S., & Jozefowicz, R. Revisiting Distributed Synchronous SGD. ICLR'16 workshop track.
- [4] Kaiming He et al. "Deep Residual Learning for Image Recognition", CVPR 2016.
- [5] <https://research.preferred.jp/2017/02/chainermn-benchmark-results/>
- [6] <https://github.com/tensorflow/tensorflow/issues/6116>

創薬研究の基盤になる化学構造創出技術と効率的GPUコンピューティングとの連携

尾谷 優子 大和田 智彦

東京大学大学院薬学系研究科薬科学専攻

新しい化学構造を持った化合物は新しい生物活性の機能を持ちえる。そのため化学構造の創出技術は創薬にとって重要である。一方で、直感に頼らず新しい化学構造をどう生み出すのか、あるいは熱的に変換する溶液中の化学構造はそもそもどう考えれば良いのかという古典的な問題が依然存在する。この記事では有機合成化学の立場で、規定された三元構造をとる2種類の分子、すなわち(1)非天然アミノ酸から作られるヘリックス分子と(2)回転可能な一重結合を多数持つフレキシブルな構造の脂質メディエーターを構造化した誘導体の創出について解説する。

今日、実験有機合成化学者は、自由エネルギーに基づく分子像を持つことが求められ、動的構造も考慮した化学構造の創出技術を考えるように迫られているが、効率的なGPUコンピューティングの助けなくしては到達し難く、GPUコンピューティングの積極的な活用が不可欠である。GPUコンピューティングと有機合成化学との組み合わせの例を紹介したい。

はじめに

1

私達は、薬学系の大学で実験有機化学・メディシナルケミストリー研究に携わる者で決してコンピューターサイエンスに詳しいわけではない。単なるTSUBAMEの利用者に過ぎない。一方で私達のような利用者の存在が今後のTSUBAMEの活用という視点で意味を持つかもしれない。

抗体医薬を含めすべての医薬品は有機化合物であるという主張が正しいとすると、薬学や創薬は生体分子と呼ばれる有機分子と医薬品という有機化合物の相互作用を研究・最適化する分野であると考えられる。すべての相互作用は水を含め、様々な溶媒(周辺)環境中で起きているため、結合の生成・消滅に伴うエンタルピー項だけではなく、溶媒の再配向に伴うエントロピー項を考慮した自由エネルギーに基づく動的挙動(「動画」のようなもの)を研究する必要がある。一方で有機化学者は化学構造を考える際、スナップショットのような「静止画」で構造をとらえる傾向がある。この「動画」と「静止画」から得られる情報は決して同じではない。このギャップを埋めるのが分子シミュレーションであり、溶媒分子を含めた分子構造の網羅的探索やトラジェクトリー解析等には効率的なGPUコンピューティングが不可欠である。そのため私達も分子シミュレーションを化学構造創出技術として活用する試みを5年程前に開始した。TSUBAME2.5を利用して得られた計算結果の多くが未発表なため本稿で紹介できない部分が多いが、実験的な視点から構造創出研究について紹介し、どのような研究にTSUBAME2.5を利用して分子シミュレーションを適用しているかを紹介し、化学構造の創出技術としてGPUコンピューティングにどれほど大きな期待を寄せているかを伝えたい。

生体でない規則構造を創り出す

2

2.1 二環性β-プロリンのホモオリゴマーの規則構造化の鎖長依存性の調査

アミノ酸が複数化学結合したペプチドは我々の生体の中で、高活性なホルモン等の生理活性物質としての機能を有する。そのためペプチドをそのまま、もしくは修飾し、医薬品に用いることも既に行われているが、ペプチド化学の最大の問題点は、生理活性を生み出す最小の活性構造が多くの場合不明であることだ。一方、非天然アミノ酸ペプチドを用いた安定な規則構造の創製は、ペプチド構造化の新たな基盤技術であり、また医薬化学への応用が期待される。

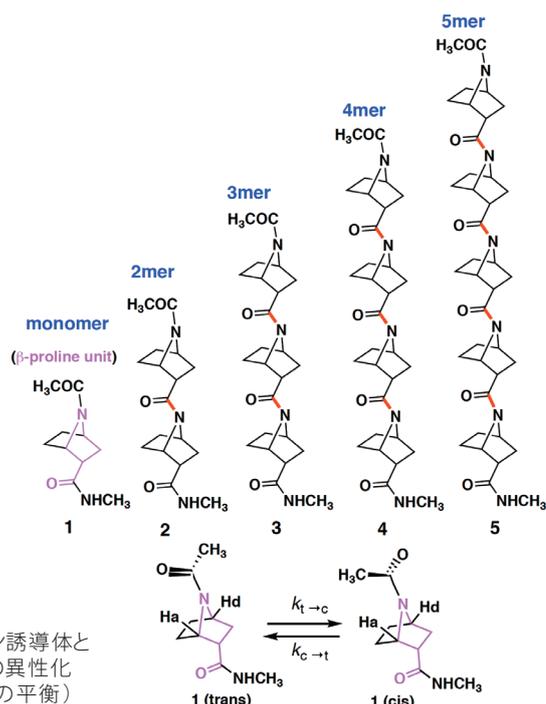


図1 人エプロリン誘導体とアミド結合の異性化(transとcisの平衡)

2.2 水素結合によるアミド窒素の ピラミッド化方向の変化の観測

タンパク質やペプチドを形作る重要な結合であるアミド結合の非平面化、すなわち窒素原子のピラミッド化が、タンパク質の構造化に寄与している可能性が古くから現在まで議論されている。タンパク質やペプチドのアミド結合は平面構造が優勢であり窒素原子は sp^2 性を帯びているため、通常ピラミッド化の度合いは非常に小さい。二環性の水素結合性の β -プロリンオリゴマーのアミド窒素はそのひずんだ環構造によりアミド結合が平面構造を取らず、アミド窒素がピラミッド化したピラミッドアミド構造を取る構造的な特徴を持つ^[1]。橋頭位側鎖が非水素結合性である $-CH_2OMe$ 基を持つ2量体((*S*)-2)と、水素結合性である $-CH_2OH$ 基を持つ2量体((*S*)-2^h)の結晶構造を比較したところ、前者ではアミド置換基の傾きの方向と橋頭位側鎖・C末側置換基の向きが逆である*anti*-ピラミッド化構造をとるのに対し、後者ではアミド置換基の傾きの方向と側鎖・C末側置換基の向きが同じである*syn*-ピラミッド化構造を取っていた。また、側鎖と隣接するアミドカルボニル基の間に分子内水素結合を形成していた(図4)^[8]。

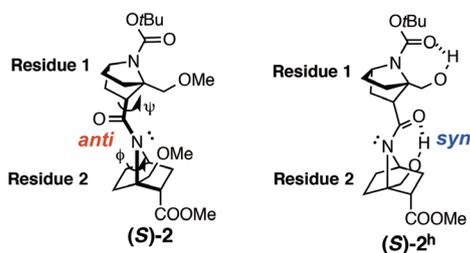


図4 非水素結合性アミノ酸2量体と水素結合。
アミノ酸2量体の結晶構造の模式図

さらに、*anti*-および*syn*-窒素ピラミッド構造は窒素の反転により速い平衡関係にあるが、側鎖の違いにより安定な異性体に変化するという結晶中で見られた傾向は溶液中でも同様であることがVCDにより示された。また、 $-CH_2OMe$ 基を持つ非水素結合性アミノ酸について、単純な構造を持つ単量体アセチルアミドをDFT計算により評価したところ、*anti*-ピラミッド化構造が*syn*-ピラミッド化構造に比べて0.2 kcal/mol 安定であり、本アミノ酸が単量体アミドにおいても同様の傾向を持つことが示唆された。

核磁気共鳴分光法(NMR)および赤外円二色性分光法(VCD)により溶液構造を調査したところ、有機溶媒中で分子内水素結合が確かめられた。さらにVCDにより、非水素結合性オリゴマーでは*anti*窒素ピラミッド化構造が優勢であり、水素結合性側鎖を持つオリゴマーの場合は*syn*窒素ピラミッド化構造が優勢であるという結晶と同様の傾向を示し、アミド基と側鎖の水素結

合が窒素ピラミッド化の方向を制御することを初めて明らかにした。

アミド窒素のピラミッド化という分子の「柔軟性(自由度の増加)」は、たった1つの水素結合で剛直性を帯びることを本研究は示唆しており、タンパク質の構造変化や構造構築における分子構造の「柔軟性(自由度の増加)」が、規則性や剛直性を作り出す際、協同的に働く可能性を示唆している点で重要な知見であると言える。

2.3 アミドのcis/trans平衡の動的固定

そもそも橋頭位に置換基を導入した二環性 β -プロリンのホモオリゴマーは溶媒、温度、濃度、鎖長に依存しない規則構造を形成することが分かっている。橋頭位の置換基を1位にすれば*cis*-アミドが^[5]、橋頭位の置換基を5位にすれば*trans*-アミドが形成される^[7]。8量体の構造を2量体、3量体の構造から推定したヘリックス構造を図5に示した(図5)。

cis-アミドのヘリックスは一残基あたり、2.21 Åの上昇、*trans*-アミドのヘリックスは一残基あたり、4.21 Åの上昇を起こす。これらの値が、天然型アミノ酸が作る α -ヘリックス(1.53 Å/residue)に比べると長い値である事が分かる。このことは、通常6アミノ酸残基でカバーしていたタンパク質表面のホットスポットを、*cis*-アミドのヘリックスは4残基、*trans*-アミドのヘリックスは2残基でカバーできる事が分かる(図6)。

Comparison of Bicyclic *cis*-Helix and α -Helix

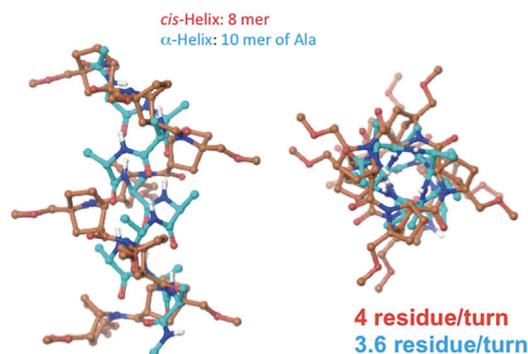


図6 アミドの*cis*ヘリックス(オレンジ)と α -ヘリックス(ブルー)の比較。規則性は生体のヘリックス構造と大きく異なっていて、短い残基数で生体ヘリックスと同じ長さを作りだせる。

このように天然アミノ酸が作れない規則構造を創成することで、タンパク質-タンパク質相互作用を阻害するペプチド性分子が*trans*-アミドの場合わずか2残基で形成できることを意味している。実際、ある種のタンパク質-タンパク質相互作用の阻害活性を持っている事が分かってきた(未発表)。当然、動的挙動を考慮した結合様式や結合過程の解析はより有効な分子設計に不可欠である。

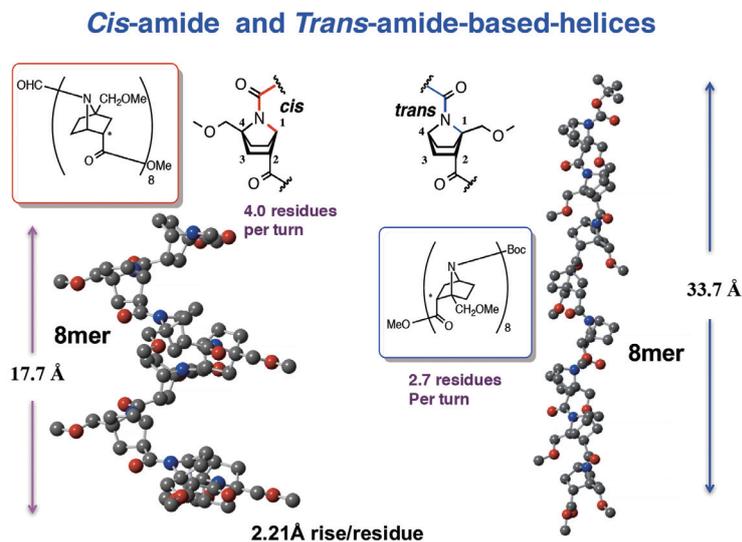


図5 アミドのcisとtransの異性化を動的に固定すると発生するヘリックス構造

脂質誘導体に基づく メディシナルケミストリー

3

私達の研究室では機能性物質の創製、特に膜タンパク質の機能を調整する物質として、リン脂質の誘導体を設計合成している。

3.1 配座固定した内因性リゾリン脂質アナログの合成と3次元構造の特性

リゾホスファチジルセリン(LysoPS)はジアシルリン脂質であるホスファチジルセリン(PS)を親分子としてPSの二つの脂質酸エステルのうち一方が加水分解され、産生されるリゾリン脂質である(図7)^[10]。細胞膜上に存在するGタンパク質共役受容体(GPCR)を介してシグナル伝達に關する脂質メディエータであることが解明されつつある。LysoPSの生理機能についてはラット腹腔マスト細胞の脱顆粒促進^[10]、ヒト末梢T細胞の増殖制御、神経細胞の突起進展促進等の複数の報告がなされているが未だ不明な点が多く残されている。2006年にLysoPSがリガンドとするGPCRの一つであるGPR34が現出された。更に2012年、我々の共同研究者により2つのオーファンGPCRsがLysoPS受容体であることが検出された。LysoPSはL-セリン、リン酸ジエステル、グリセロール、長鎖脂肪酸部位から構成されていて、自由回転可能な結合を多く含む構造柔軟性が高い分子であるため^[11]三つの受容体を同時に活性化することが可能である。私達はグリセロール部位のコンフォメーションを固定したLysoPS誘導体をデザインし、合成法を確立した(図7)^[12]。LysoPS受容体の活性化に必要な構造要求性を探索した。受容体のなかで特に

GPR34(LPS1)およびP2Y10(LPS2)をターゲットとして研究を行った。これらの構造の類似した誘導体の中ではGPR34に対して活性がない誘導体、活性が強い誘導体があり、またP2Y10に対して活性がない誘導体、活性が強い誘導体がありこの相違を活性構造の検索に利用しようとしている。グリセロール部位の平面性が受容体の選択性に関係しているのではないかという仮説を提出した。この仮説の検証には分子シミュレーションが有効であるが、このような小分子の取り得る構造空間の網羅的な探索ですら大きなGPUパワーが必要である事が分かって来ている。この辺の詳細は公開できる段階になって改めて紹介したい。また最近、有機合成化学と構造生物学と分子シミュレーションの組み合わせが受容体-リガンド認識の分子像の解明にとっても有効である事を経験した^[13,14]。

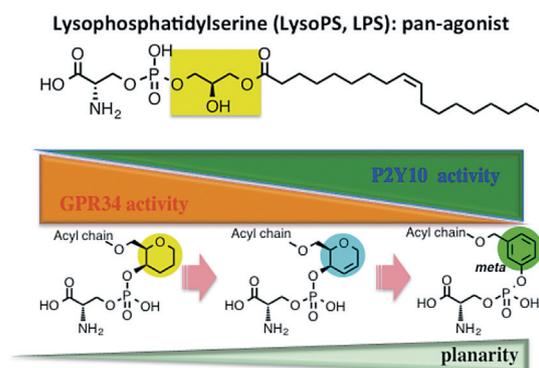


図7 生理活性脂質(リゾリゾホスファチジルセリン(LysoPS))のグリセロール部位のコンホメーション固定と平面性と受容体選択性の関係。

効率的な GPUコンピューティングとの連携

4

統計的な調査ではなくあくまで個人の経験に基づく見解であるが、有機化学者が行う有機化学反応、特に有機合成反応の平均的な反応時間（反応が終結したと見なされるまでの時間。この反応時間は必ずしも最適された最短時間を意味しない点にご注意願いたい）は18時間（3/4日）である。18時間以上かかる有機化学反応は遅く感じ、18時間以内、特に1時間以内の反応は早く感じる。分子シミュレーションを実験有機化学・メディシナルケミストリーに採り入れていく際、計算時間への感覚も反応時間の感覚に似てくると思われる（これは実感している）。進展著しいGPUコンピューテーションはこの化学者の持つ体内時計に計算時間を近づけていて、今後益々実験研究者の利用が拡大するであろう。計算手法のノウハウだけではなく情報学などIT創薬の根底にある知識や体系的な理解を普及させる教育や多くのデータポイントの解析手法を利用者と共同開拓することも今後益々必要かつ重要になっていくであろう。

謝辞

本稿で紹介した東京工業大学クラウド型グリーンスーパーコンピュータシステムTSUBAME2.5で実施した計算について未発表分も含め、“High Performance Computing Infrastructure (HPCI)”プロジェクト研究課題名“創薬ターゲットタンパク質の内在性リガンドの合成改変のための立体構造予測”（hp150177）（大和田）、研究課題名“分子シミュレーションを用いる生体内リガンドの網羅的構造化”（hp170146）（大和田）、平成28年度TSUBAME女性利用者支援制度 研究課題名“架橋プロリンを用いたアミノ酸配列からの構造モチーフの創製”（16J0003）（尾谷）のご支援を頂いた。また研究費として科学研究費補助金、新学術領域「理論と実験の協奏による柔らかな分子系の機能の科学」研究課題名：プロリン型の非天然アミノ酸の柔らかいアミド結合に基づいた規則構造制御（平成26-27年度）（JP 26104508）（尾谷）、基盤研究B、研究課題名：タンパク質表面での反応を制御するペプチドの構造機能化（2014-2016）（JP 26293002）（大和田）のご支援を受けた。

参考文献

- [1] An Evaluation of Amide Group Planarity in 7-Azabicyclo[2.2.1]heptane Amides. Low Amide Bond Rotation Barrier in Solution. Yuko Otani; Osamu Nagae; Yuji Naruse; Satoshi Inagaki; Masashi Ohno; Kentaro Yamaguchi; Gaku Yamamoto; Masanobu Uchiyama; Tomohiko Ohwada *Journal of the American Chemical Society*, **2003**, 125, 15191-15199.
- [2] Alpha, alpha-Disubstituted Amino Acids Bearing a Large Hydrocarbon Ring. Peptide Self-Assembly through Novel Hydrophobic Recognition Ohwada, T.; Kojima, D.; Kiwada, T.; Futaki, S.; Sugiura, Y.; Yamaguchi, K.; Nishi, Y.; Kobayashi, Y. *Chemistry-A European Journal*, **2004**, 10, 617-626.
- [3] Oligomers of beta-Amino Acid Bearing Nonplanar Amides form Ordered Structures Yuko Otani, Shiroh Futaki, Tatsuto Kiwada, Yukio Sugiura, Atsuya Muranaka, Nagao Kobayashi, Masanobu Uchiyama, Kentaro Yamaguchi and Tomohiko Ohwada *Tetrahedron*, **2006**, 62, 11635-11644.
- [4] Nonplanar Structures of Thioamides Derived from 7-Azabicyclo[2.2.1]heptane. Electronically Tunable Planarity of Thioamides Hori, T.; Otani, Y.; Kawahata, M.; Yamaguchi, K.; Ohwada, T. *Journal of Organic Chemistry*, **2008**, 73, 9102-9108.
- [5] Water-stable Helical Structure of Tertiary Amides of Bicyclic β -Amino Acid Bearing 7-Azabicyclo[2.2.1]heptane. Full Control of Amide Cis-Trans Equilibrium by Bridgehead Substitution. Masahiro Hosoya; Yuko Otani; Masatoshi Kawahata; Kentaro Yamaguchi; Tomohiko Ohwada *Journal of the American Chemical Society*, **2010**, 132, 42, 14780-14789.
- [6] Secondary Structure of Homo-thiopeptides Based on a Bridged β -Proline Analogue: Preferred Formation of Extended Strand Structures with Trans-Thioamide Bonds Yuko Otani, Tetsuharu Hori, Masatoshi Kawahata, Kentaro Yamaguchi and Tomohiko Ohwada *Tetrahedron*, **2012**, 68 (23), 4418-4428 Special Issue (Symposium in print) on Chemistry of Foldamers

- [7] Robust Trans-Amide Helical Structure of Oligomers of Bicyclic Mimics of β -Proline: Impact of Positional Switching of Bridgehead Substituent on Amide Cis-Trans Equilibrium.
Siyuan Wang; Yuko Otani; Xin Liu; Masatoshi Kawahata; Kentaro Yamaguchi; Tomohiko Ohwada
Journal of Organic Chemistry, **2014**, 79 (11), pp 5287-5300.
- [8] Hydrogen Bonding to Carbonyl Oxygen of Nitrogen-Pyramidalized Amide-Detection of Pyramidalization Direction Preference by Vibrational Circular Dichroism Spectroscopy.
Siyuan Wang, Tohru Taniguchi, Kenji Monde, Masatoshi Kawahata, Kentaro Yamaguchi, Yuko Otani, Tomohiko Ohwada
Chemical Communications, 2016, 52, 4018 - 4021,
- [9] Molecular Dynamics Study of Nitrogen-Pyramidalized Bicyclic β -Proline Oligomers: Length-Dependent Convergence to Organized Structure
Yuko Otani, Satoshi Watanabe, Tomohiko Ohwada, and Akio Kitao
J. Phys. Chem. B, 2017, 121 (1), pp 100-109. Publication Date (Web): December 5, 2016
- [10] Synthesis and Evaluation of Lysophosphatidylserine Analogs as Inducers of Mast Cell Degranulation. Potent Activities of Lysophosphatidyl- threonine and Its 2-Deoxy Derivative
Masazumi Iwashita, Kumiko Makide, Taro Nonomura, Yoshimasa Misumi, Yuko Otani, Mayuko Ishida, Ryo Taguchi, Masafumi Tsujimoto, Junken Aoki, Hiroyuki Arai, Tomohiko Ohwada
J. Med. Chem., 2009, Volume 52, Issue 19, pp 5837-5863.
- [11] Structure-Activity Relationships of Lysophosphatidylserine Analogs as Agonists of G-Protein-Coupled Receptors GPR34, P2Y10, and GPR174
Masaya Ikubo, Asuka Inoue, Sho Nakamura, Sejin Jung, Misa Sayama, Yuko Otani, Akiharu Uwamizu, Keisuke Suzuki, Takayuki Kishi, Akira Shuto, Jun Ishiguro, Michiyo Okudaira, Kuniyuki Kano, Kumiko Makide, Junken Aoki, and Tomohiko Ohwada
J. Med. Chem., 2015, 58, pp 4204-4219.
- [12] Conformational Constraint of the Glycerol Moiety of Lysophosphatidylserine, an Emerging Lysophospholipid Mediator, Affords Compounds with Receptor Subtype Selectivity
Sejin Jung, Asuka Inoue, Sho Nakamura, Takayuki Kishi, Akiharu Uwamizu, Misa Sayama, Masaya Ikubo, Yuko Otani, Kuniyuki Kano, Kumiko Makide, Junken Aoki, Tomohiko Ohwada
Journal of Medicinal Chemistry, 2016, 59 (8), pp 3750-3776.
- [13] Probing the Hydrophobic Binding Pocket of G-Protein-Coupled Lysophosphatidylserine Receptor GPR34/LPS1 by Docking-Aided Structure-Activity Analysis
Misa Sayama, Asuka Inoue, Sho Nakamura, Sejin Jung, Masaya Ikubo, Yuko Otani, Akiharu Uwamizu, Takayuki Kishi, Kumiko Makide, Junken Aoki, Takatsugu Hirokawa, and Tomohiko Ohwada
J. Med. Chem., 2017, 60 (14), pp 6384-6399.
- [14] Structural insights into ligand recognition by the lysophosphatidic acid receptor LPA₆
Reiya Taniguchi, Asuka Inoue, Misa Sayama, Akiharu Uwamizu, Keitaro Yamashita, Kunio Hirata, Masahito Yoshida, Yoshiki Tanaka, Hideaki E. Kato, Yoshiko Nakada-Nakura, Yuko Otani, Tomohiro Nishizawa, Takayuki Doi, Tomohiko Ohwada, Ryuichiro Ishitani, Junken Aoki, Osamu Nureki
Nature, 548, 356_360 (17 August 2017).

● **TSUBAME e-Science Journal vol.16**

2017年11月13日 東京工業大学 学術国際情報センター発行 ©
ISSN 2185-6028

デザイン・レイアウト：キックアンドパンチ

編集：TSUBAME e-Science Journal 編集室

青木尊之 渡邊寿雄 板倉有希

住所：〒152-8550 東京都目黒区大岡山 2-12-1-E2-6

電話：03-5734-2085 FAX：03-5734-3198

E-mail：tsubame_j@sim.gsic.titech.ac.jp

URL：http://www.gsic.titech.ac.jp/

TSUBAME

TSUBAME 共同利用サービス

『みんなのスパコン』TSUBAME共同利用サービスは、
ピーク性能 12.15 PFlops、15120CPUコア、2160GPU搭載
世界トップクラスの東京工業大学のスパコンTSUBAME 3.0を
東京工業大学以外の皆さまにもご利用いただくための制度です。

TSUBAME 共同利用サービスの利用区分とカテゴリ

TSUBAME 共同利用サービスには「学術利用」と「産業利用」の利用区分があります。
「学術利用」は、HPCI(革新的ハイパフォーマンスコンピューティングインフラ)、
JHPCN(学際大規模情報基盤共同利用・共同研究拠点)の採択により無償でご利用になれる制度と、
東京工業大学学術国際情報センターが実施する有償の制度がご利用になれます。
「産業利用」は、HPCIまたはJHPCNの採択により無償でご利用になれる制度と、
東京工業大学学術国際情報センターが実施する有償の制度があり、
有償の制度には利用目的や利用成果を非公開とする「成果非公開」のカテゴリもあります。

TSUBAME 共同利用サービスの利用料金

TSUBAME3.0における計算資源は口数を課金単位としております。
1口は1000ノード時間積で、1計算ノード(28CPUコア、4GPU、256GBメモリ搭載)を
1000時間あるいは100計算ノードを10時間というように、
ご利用の用途に合わせて自由にご利用になれます。
TSUBAME 共同利用サービスの利用区分・カテゴリ・利用料金を下表に示します。

利用区分	利用者	制度	募集期間	申請および審査	成果	料金(税別)	
学術利用	他大学 または 研究機関 等	HPCI	年1回 10月頃	HPCI 運用事務局 (高度情報科学技術研究機構)	公開	無償	
		JHPCN	年1回 1月頃	JHPCN 拠点事務局 (東京大学 情報基盤センター)	公開	無償	
		TSUBAME 学術利用	随時 募集中	東京工業大学 学術国際情報センター	公開	1口 100,000円	
産業利用	民間企業	HPCI	実証利用	年1回 10月頃	HPCI 運用事務局 (高度情報科学技術研究機構)	公開	無償
			トライアル・ ユース	随時 募集中			
		JHPCN	企業共同 研究課題	年1回 1月頃	JHPCN 拠点事務局 (東京大学 情報基盤センター)	公開	無償
		TSUBAME 産業利用	随時 募集中	東京工業大学 学術国際情報センター	公開	1口 100,000円	
					非公開	1口 200,000円	

お問い合わせ

- 東京工業大学 学術国際情報センター 共同利用推進室
 - e-mail kyoyo@gsic.titech.ac.jp Tel.03-5734-2085 Fax.03-5734-3198
- 詳しくは <http://www.gsic.titech.ac.jp/tsubame/> をご覧ください。