

コロナ禍と半導体不足

伊藤 和人

情報メディア基盤センター長

世界的な半導体不足になっている。半導体は電子部品材料であるが、材料としての半導体が不足しているわけではない。半導体を材料とする電子部品の製造が需要に追いつかない、つまり電子部品不足の状況を「半導体不足」と称している。半導体を材料とする電子部品には発光ダイオード (LED) やトランジスタなど多くの種類がある。トランジスタは、信号の増幅や電流のオンオフ制御などを行う。多数のトランジスタを組み合わせた電子回路を数 mm 角の半導体チップの表面上に作成したものを集積回路 (IC)、トランジスタ数が多い IC を大規模 IC (LSI) という。LSI は一種類ではなく、トランジスタの数や接続方式を変えることで、演算、データ処理、記憶、制御など様々な機能を持つ多種の LSI が作られている。PC やスマホの中でアプリを実行しているマイクロプロセッサや、データを記録するメモリも LSI である。こういった LSI が部品として PC、スマホ、ゲーム機のほか、テレビ、エアコン、洗濯機などの家電製品に組み込まれており、LSI はなくてはならないものとなっている。自動車にもエンジン制御やカーナビなど、1 台あたり 100 個ほどの LSI が用いられている。半導体不足により、自動車の納車待ちが長期化しているという。

マイクロプロセッサやメモリなどの汎用的な一部の種類の LSI を除き、組み込み先の製品種類ごとに専用の LSI が設計・製造されている。必要な機能の性能を高める一方で不要なトランジスタを省き、LSI の高性能化と小型・低消費電力・低価格を両立させるためである。1 つの LSI に用いられるトランジスタの数は数百万、数千万になり、十億を超えることもある。このような LSI を最小配線幅が 10~100nm (1mm の 1 万~10 万分の 1) 程度の最先端微細加工技術で製造できる工場を作るには数千億円規模の経費が必要であり、LSI を大量生産できる企業数は世界的にもごく限られている。そのため、多くの LSI は工場を所有する企業 (ファウンドリという) に委託して製造されている。コロナ禍でテレワーク用に PC が売れ、外出を控えるため家庭用ゲーム機が売れるなど、いわゆる巣ごもり需要により様々な製品に用いられる LSI が多数必要となった。また、外出控えで自動車の販売が減ると予想されたが実際には売れ行きは下がらず、多くの車載用 LSI が必要になった。ところがファウンドリの製造能力には限りがあり、急に工場を増やすことも難しい。結果としてファウンドリの取り扱いになっている。必要な種類の LSI が必要な数だけ調達できない「LSI 不足」が現在の半導体不足である。

LSI は情報ネットワーク機器にも用いられている。埼玉大学でも令和 2 年度末に体育館に無線 LAN を設置した際に機器納入が危ぶまれた。半導体不足が原因と思われるが、幸い予定通りに設置を完了することができた。半導体不足は実はこれまでも何度か起きており、やがて解消すると思われるが、新型コロナウイルス感染拡大の影響はこんなところにも及んでいる。

『最適な制御』に必要なものとは？

大久保 潤

1. 制御について考える

《制御》という言葉は色々な分野で使われる。この言葉に対応する英単語である《control コントロール》と言い換えたほうが想像しやすいかもしれない。

機械やロボットを《コントロール》する。

感染症が広がらないように《コントロール》する。

車や飛行機を《コントロール》する。

最後の例は《操作》と訳したほうが適切かもしれないけれども、ここではこれらに共通するものは何かを考えてみたい。

なお、社会における制御をメディア論や哲学と絡めて議論した書籍もある [1]。制御やコントロールは工学の話だけではなく、いまや社会で生活する全ての人に関係している。制御やコントロールだと意識されることなく、人間の心理を上手く利用して特定の行動を促す研究も進んでいる。例えば特定の情報を知らされない状況や、逆に特定の情報のみを与え続けられる状況などの情報のフィルタリングや取捨選択も、一種の《制御》と捉えることができる。知らず知らずのうちに《制御》されていると考えると非常に怖い気もするが、一方で予算をかけない効果的な政策に結びつく点も注目されている [2]。

以下で述べていくのは主に機械の制御に関するものではあるものの、これをきっかけにして、日常生活における《制御》について考えてみてはいかがだろうか。

2. 何かを制御するために必要なものは？

何かを《制御》するためには、何が必要だろうか？ そして、何かを《制御》するとは、そもそもどのようなことを意味しているのだろうか？

まずは、制御する対象に関する知識が必要となる。例えばボールを投げて目的の場所に当たたいとする。ボールをどのような角度で、どのような速度で投げるとどのくらい飛ぶのか、それは理科や物理での知識を使うと予想できる。重力の影響、そして場合によっては空気抵抗の影響を考慮して方程式を立てれば、ボールがどのように飛んでいくのかがわかる、ということである。つまり、方程式を立てることさえできれば、投げ方を色々変えた場合の軌道を計算できる。軌道を計算できれば、目的の場所に向かう投げ方を選べばよい。場合によっては目的の場所から逆向きに方程式を解くことで、最初の投げ方を決めることもあ

注) 本稿は、筆者の研究の一端をできるだけわかりやすく紹介することを意図した《読み物》です。

る。このように、対象に関する知識を持つことで《目的の場所にボールを当てる》という制御が可能になる。

対象を知ること。これは色々なところに通じる話だろう。

もちろん、方程式には様々なパラメータがある。重力加速度、空気抵抗の強さ、ボールの重さなど。正確に測定できてわかっているパラメータもあれば、わからないものもある。パラメータにずれがあると、当然、求めた軌道も変わる。そこで、わからないパラメータを現実と擦り合わせる作業が必要になる。こういった作業をシステムやパラメータの《同定》と呼ぶ。地味な作業ではあるものの、これも大切な作業である。

もう少し違う形で対象を知ることでもある。例えば車の運転。アクセルを踏むと、進む。ハンドルを回すと、曲がる。車も当然のことながら機械なので、すべてを方程式で書くことができるはず……ではあるが、駐車場にバックで車を入れる時に、方程式を意識して車を運転している人はおそらくいないだろう。意識をすると逆にぶつけてしまいそうで、怖い。

どのくらいアクセルを踏むとどのくらい進むのか、ハンドルをどのように回すとどんな具合で曲がるのか。これらを教習所の練習で試行錯誤しながら、徐々に感覚的に身につける。これはつまり、データに基づいて、入力と出力の関係性を作ってしまうことに対応する。《ブラックボックスモデル》と呼ばれたり、最近の機械学習だと《エンドツーエンド》の学習とも呼ばれたりする「中身が現実には即しているかどうかは気にしない」というこの方法も、制御対象に関して知る、という意味合いがあり、とても役立つものである。

さて、制御する対象に関する知識が手に入ればそれで十分だろうか？

制御のためにはもう一つ、重要なものがある。それは《どのような制御をしたいのか》という目的である。

3. 「良い」の基準

制御の《目的》なんてわかりきっている、と考えるかもしれない。でも、果たして《目的》は自明なものだろうか。

例えば、今いる場所から自分の家に帰りたいとき、何を考えるだろうか。

電車やバスを使った帰路を考える。その際、どの経路を選ぶべきか。電車の遅延情報が入っているから、こちらの経路だと時間がかかりそうである。別の経路だと早く帰ることができるけど、少し値段が高い。いやいや、別のこちらの経路を選べば、途中で寄りたいお店があるから、ついでに買い物をすることができる……。このように色々なことを考えるのではないだろうか。単純に、今いるところから直線的に目的地に向かえるわけではない。時間を優先するのか、それとも切符代などの金額を優先するのか。このように考えていくと、実は《目的》も定まっているわけではない。

人工知能の要素技術である機械学習でも、そして政策決定などの広い分野での活用が期待されているデータサイエンスでも、実はこのような《目的》をどのように設定するのかで

結果が大きく変わることが知られている。

工学における《制御》においても様々な《目的》を考えることができる。図1に、よく用いられる《目的》を示した。ポイントは「良い」経路とはどのようなものを数値的に表すことである。数値で表現できれば、「良さ」が最も高くなるような経路を選び、その経路を通るように制御をすれば良い。

ここで考える「良さ」は以下の3つである。

- ① 最終的に、目的の場所にできるだけ近い場所に到着できれば「良い」
- ② 途中で通りたくない場所を避けつつ、通りたい場所を通れると「良い」
- ③ できるだけ少ない労力（エネルギー）で済むほど「良い」

ここでは「良さ」を定義したが、普通はその逆の「コスト」を定義する。コストが高いほど避けたい経路、という意味である。①の場合は『終端コスト』などと呼ばれる。もちろん目的の場所に到着できなければ困るけれども、完全に一致させなくてもよい場合も多い。ロボットの制御で、ミリメートル単位の誤差はあってもよい場合もある。その場合には、目的地周辺でコストを小さくしつつ、目的地からある程度離れると高いコストがかかるように設定すればよい。②は『ランニングコスト』などと呼ばれる。ロボットアームの制御などで、構造的に無理な姿勢の場合にはコストを高くしておけばよい。③は少し理解しづらいかもしれないが、非常に重要なコストである。例えばモーターを動かす場合に、大電力を使えば素早く正確に制御できるかもしれない。ただ、そのために高い電気代を必要とするのは効率が悪い。できる限り効率的に、目的を達成するほうが好ましい。

工学における制御にも様々なものがあるが、よく用いられる問題設定はここに述べた形

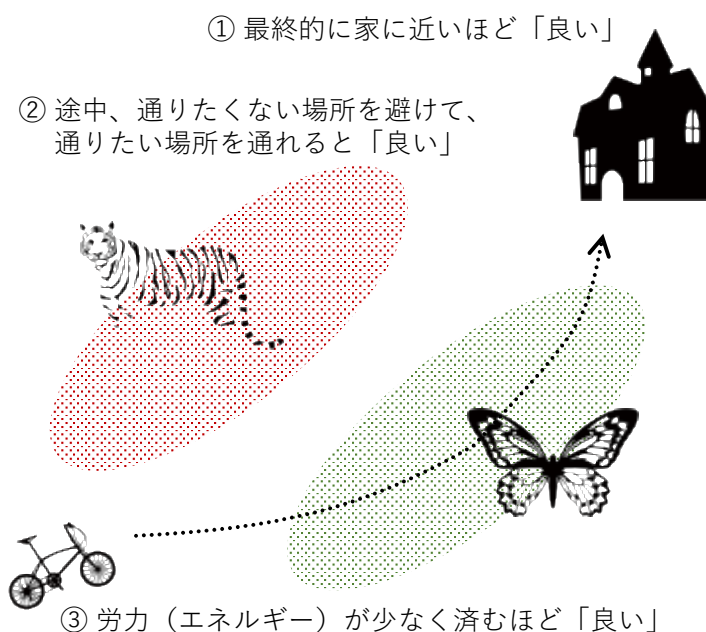


図1 「良い」経路を通るように《目的》を決める

である [3]。この《目的》、つまり《良さ》をどのように設定するかによって、制御の方法が変わることは想像しやすい。

以上の説明のように、機械に制御をさせるためには《目的》を数値で表す必要がある。その方法も人によって様々であり、設定によって結果も様々である。《目的》を決めると簡単に言っても、実は悩ましい問題がたくさんある。

少し話は変わるが、どのように《目的》を設定すれば良いかわからない場合もある。例えば卓球をしていて、どのようにラケットを操作すれば相手に適切な球を返せるのだろうか。このように腕を動かすと良さそう、逆にこうすると悪そう、ということがわかればコスト関数を設定できるが、実際に考えてみるとなかなか難しい。コスト関数さえわかれば、最適な腕の動かし方を計算させて、あとはそのように腕を動かす訓練をすれば良いだけに…。

そこで今は『逆強化学習』と呼ばれる研究分野などにおいて、経路に基づいて逆にコスト関数を見つけようとする研究もされている [4]。卓球選手の腕の動かし方は、きっと「何らかのコスト関数」を最小にするように、つまり相手に良い球を打ち返すという《目的》を達成するようなものになっているはずである。よって、画像などをデータとして利用して、その腕の動かし方をする場合のコスト関数を算出するのである。これによって、このような腕の動かし方は良くないはず、などの知見を得ることもできる。

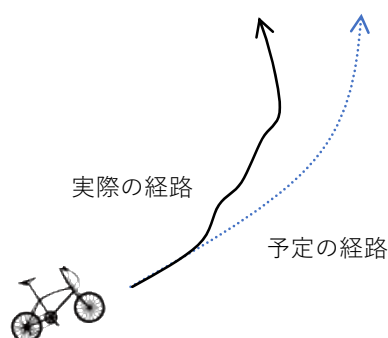
4. ほんの少しだけ先を予測する

制御する対象について知り、そして何を「良い」と考えて制御をするかを決めれば、制御方法を導くことができる……のだけれど、実際には制御方法を見つけることもなかなか難しい。色々な制御を試してみてどれが良い方法を調べてみようとしても、候補が多過ぎて、高性能な計算機でも全てを探索するためにもものすごい時間が必要になったりする。何かを

最初にすべての計画を立てる

「目的地に向かうためには右に10度 → 左に5度 → …」

※「ずれ」が蓄積してしまう



現在の状態を確認して、短く予定を立て直す

「目的地に向かうためには右に10度」→ 少しずれる

「ここから目的地へは、左に4度」

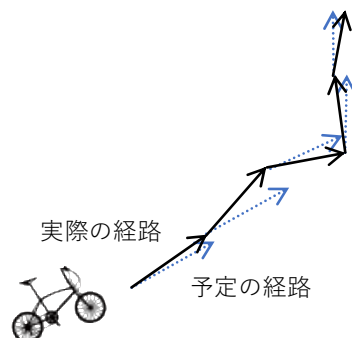


図2 短く制御入力を求めるモデル予測制御

制御しようと思って制御方法を探しているうちに、制御すべきタイミングを失ってしまった、なんてことになったら困る。

以前はリアルタイムに制御入力を求めることはなかなか難しかったものの、最近になって色々な技術が発展してきた。その一つが非線形モデル予測制御と呼ばれるものである [3]。簡単な問題（線形の問題）は数式で「解く」ことができるため制御が容易だったが、世の中には複雑な動きをする制御対象も数多くある。そして、そのような問題（非線形性）に対する一般的な解法はない。ではどうするのか。そこで、モデルを作って、ほんの少しだけ先の未来に制御対象がどのようになるのかをシミュレーションして「予測」しながら、少しずつ制御方法を探すというのが、非線形モデル予測制御のアイデアである。

そもそも目的地までのすべての制御入力を一度に求める必要はない。どうせ制御中に予想もつかない外からの力（ノイズ）が加わったりするのだから、少しだけ先を予測して、制御方法も少しずつ変えていけばよい。図 2 に示すように、現在の状態を確認しながら短く予定を立て直す。リアルタイムに制御を実行するこの方法で、ホバークラフトの姿勢制御なども含め、リアルタイム性が求められる制御が少しずつ可能になりつつある。

短く目標を定めて、現状を把握し、その都度修正をしていく。これは工学的な制御に限らずに大切な考え方かもしれない。

5. 制御の高速化を目指して

制御に関する研究は進んでいるものの、まだ大規模な対象をリアルタイムで制御することは難しい。「良い」制御方法を探すためには高い計算能力をもつ計算機があったほうが助かる。そのため、3D グラフィックスを高速に描画するためのチップである GPU を活用した制御入力の計算方法なども提案されている。この GPU を用いた計算方法は深層学習などの人工知能技術でも利用されているものの、チップそのものが高価であるばかりではなく、消費電力も大きい。一方で、IoT(Internet of Things)の時代、つまり色々なものに小さなセンサなどが搭載され、小さい計算能力をもつ機械が偏在する世の中になりつつある。すると、小さい計算能力しかもたない計算機で、いかに効率よく制御方法を見つけるか、という研究も重要となる。

まだ発展途上ではあるものの、「双対過程」の概念を利用した計算手法によって「予測」を効率よく行えることがわかりつつある。そして、この概念を制御の問題にも利用できることが筆者の最近の研究で明らかとなった[5]。経路積分と呼ばれる物理学で用いられる概念、そしてデータ解析の分野などで近年注目を浴びつつある Koopman 作用素の概念のつながりが見出され、結果として、計算量を減らせる場合があることを示された。今後、さらに具体的な計算方法を工夫するなどして、計算能力の小さい計算機でも可能な制御方法へとつなげていきたい。

制御は、工学の分野においては古くから研究されているが、まだまだ活発に研究されてい

る領域である。高速化、そして省力化は SDGs の観点からも望ましい。社会からの要求に応えるように、研究すべきことがどんどん増える。そのために、新しい数理的な手法を持ち込む、といった試みも必要になってきている。理論と実用のバランスを取りながら、着実に研究が進みつつあり、また、技術の社会実装も進んでいく。

最後に、もう少しだけ想像を広げてみよう。具体的な制御の方法の研究は着実に進む。ではそれで十分かと言えば、そうではない。本稿でも強調したように、制御の技術のポイントは「良い」制御とは何かという基準を決めることにある。特に工学だけではなく、社会や、少し怖いことではあるが個人の活動を《制御》することも出てきてしまうかもしれない。その際に、何を「良い」とするのかの議論は、工学だけで閉じる話ではない。人間の無意識を利用した制御は許されるのだろうか。場合によっては、そういった制御に対抗するための方法なども必要になってくるかもしれない。もちろん、無意識を利用するから悪い、というわけでは必ずしもないだろう。例えば無意識にゴミをきちんとゴミ箱に捨てるようになる、無駄な買い物を控える、などは好ましい《制御》かもしれない。大切なのは、しっかりと何が行われているかを理解して、必要な時には反対意見を述べつつ、議論をすることである。もちろん、制御をするためには制御対象を深く知る必要があるため、心理学の研究などの知見も大切になる。人間に関する学問である人文科学、社会科学と理学、工学などのつながりが、ますます重要になってくるはずである。

《制御》というキーワードで、本当に様々なことを考えることができる。対象を知ること、そして制御の目的を恣意的に決めることなど、制御の基本的な考え方を知っておくと、日常生活でのふとした出来事を制御の視点から眺められるようになるかもしれない。そうすると、なんとなく流されてしまう行動を自覚できるかもしれない。さらに、自分が本当にしたいことを見定めて、目的を達成するために積極的に行動できるようになる……かもしれない。

参考文献

- [1] 北野圭介：制御と社会，人文書院（2014）.
- [2] キャス・サンスティーン（田総恵子訳）：シンプルな政府，NTT 出版（2017）.
- [3] 大塚敏之：非線形最適制御入門，コロナ社（2011）.
- [4] 牧野貴樹 ほか編著：これからの強化学習，森北出版（2016）.
- [5] Ohkubo, J.: Connection among stochastic Hamilton-Jacobi-Bellman equation, path-integral, and Koopman operator on nonlinear stochastic optimal control, Preprint: arXiv:2012.05514.

学生実験「IoT 環境モニタリング」の実践

長谷川 有貴

1. はじめに

工学部電気電子物理工学科の実験科目である、3 年生対象の電気電子物理工学実験 III では、准教授以上の学科教員がそれぞれ 1 テーマずつ実験テーマを設けており、学生は全 23 テーマの中から、1 テーマ 4 週(4 日)×3 テーマを受講する。この科目は、4 年生からスタートする卒業研究の予行練習の意味合いを兼ねている面もあり、基本的には、各教員の研究分野に関連したテーマが設定されている。

私の研究分野では、植物工場における栽培効率の向上や味覚センサやガスセンサを用いた食品の品質評価などに関する研究を行っており、温度、湿度、二酸化炭素濃度などの環境モニタリングが必須となる。そこで私が担当する実験では、「IoT 環境モニタリングに関する実験」をテーマとして設定した。この実験では、受講学生に研究活動の一部を体験してもらうことを目的として、ハードディスクや電源を持たず、一枚の基板に最低限の機能のみを搭載したワンボードマイクロコンピュータの一つである Raspberry Pi Zero WH (以下、ラズパイ) と、さまざまなセンサを組み合わせた環境モニタリングシステムの提案と実装を題材としている。さらに、「IoT (Internet of Things)」化するため、センサで計測したデータは Twitter などの SNS を介してインターネット上に定期的に発信されるようにして、遠隔地からもセンサ情報が確認できるようにしている。

この実験では、学生が 2 名一組となり、環境モニタリングの必要性や対象を理解した上で、自ら課題を設定し、解決する力を身につけてもらうことを意識し、どのような場所で、どのようなセンサを使用してなにを測定するか、さらに得られたデータをどのように解析して得られた情報を表現するかを、すべて学生たち自身が設計、実装している。学生たちの自由な発想からさまざまな実験が生まれ、コロナ禍における講義室の換気状況や学生の集中度、図書館内の照度と快適さの評価と比較、喫煙所の利用状況とガス濃度の関係など、さまざまな環境モニタリングが行われてきた。本稿では、2021 年度前期に受講した学生たちが提案して実施した実験・研究内容とその結果の一部をいくつか紹介する。

2. テーマ「IoT 環境モニタリングに関する実験」の概要

ある曜日の 3-5 限にかけて、4 週連続で行われるこの実験では、まず 1 週目に、IoT 環境モニタリングのイメージを持ってもらうための簡単な講義と、ラズパイの基本操作、ブレッドボードを利用した回路の組み方やプログラム作成の練習を行う。2 週目には、ラズパイとセンサの接続とデータ取得の練習として I2C (Inter-Integrated Circuit) 通信可能な温度湿度気圧センサモジュールを実装し、センサから得られた情報を実験用に作成した Twitter

アカウントでツイートする機能を付与する。ここまでで、IoT 環境モニタリングに必要な環境は整い、あとは学生自身で、測定したい環境と必要なセンサを考え、実験目的、センサの設置場所、使用するセンサの型番、配線図、プログラムなどを含む企画書を提出してもらい、企画書の内容を教員が確認した上で必要なセンサ等をすぐに発注する。3 週目には、いよいよ、企画書に基づいてラズパイとセンサを接続し、実装していく。選定したセンサによっては実装のための設定やプログラミングに時間がかかる場合があるが、なんとか 5 限までに実装を済ませたら、測定現場に設置し、最終週となる 4 週目までの 1 週間の測定を行う。ラズパイに、ブレッドボードを介して温湿度気圧センサモジュール (BME280) および焦電型赤外線センサモジュール (人感センサ, SB612A) を接続した例を図 1 に示す。4 週目には、データを回収し、得られたデータの解析、比較を行った結果をプレゼン資料およびレポートとしてまとめ、本実験が完了する。

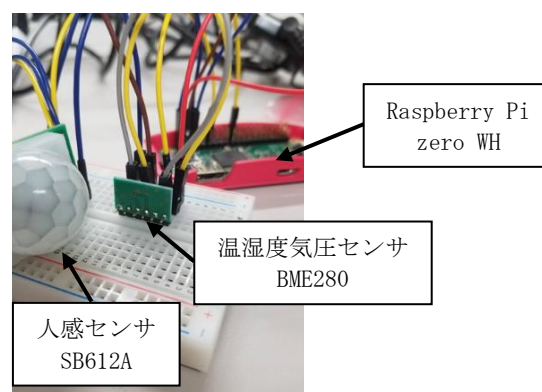


図 1 ラズパイとセンサの接続例

3. 実践例

まず、工学部講義棟 50 番講義室の環境状態の把握を目的として、講義室後方入り口付近にセンサを設置し、講義室への人の出入りと温度、湿度、気圧、二酸化炭素濃度の変化を 1 週間測定した例について紹介する。

厚生労働省により定められている建築物環境衛生管理基準^[1]では、空気調和設備を設けている居室における良好な状態として、二酸化炭素の含有率(濃度)を 100 万分の 1000 以下(= 1000 ppm 以下)、温度を 17 °C~28 °C、相対湿度を 40%~70%などの値が提示されており、特に、二酸化炭素濃度は、換気の目安になることから、新型コロナウイルス感染症防止対策の基準としても用いられるため、最近は病院やレストランなどでも簡易な二酸化炭素センサが設置されているところも多い。さらに、二酸化炭素濃度が高くなると、集中力の低下、眠気、頭痛、耳鳴りなど、人体にも影響を及ぼすことが知られており、そのモニタリングは非常に重要である。

講義室での測定結果例を図 2 に示す。この図は、13:30~17:50 まで対面授業(図中の枠内)が実施されていた、ある 1 日における講義室内の二酸化炭素濃度変化と人感センサの反応回数(カウント数)を示している。図から、講義のない

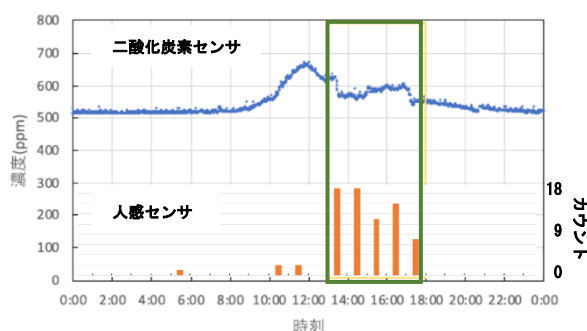


図 2 講義室における二酸化炭素センサと人感センサの測定結果例

午前 10 時以降に人感センサに反応が見られ、この反応と同時に二酸化炭素濃度が上昇していることがわかる。また、対面授業時間中は、二酸化炭素濃度に変動はあるものの、600 ppm 程度の濃度を推移しており、空調を適切に稼働させ換気が十分に行われている様子が確認された。

次に、全学講義棟 1-206 教室において、授業形態と学生の集中度との関連を調べることを目的として、講義室後方に温度、湿度、気圧センサ、照度センサ、そしてカメラを設置し、約 10 日間に渡って測定した例を紹介する。

大学教育における学生の集中度と講義形態との関連については、従来からさまざまな意見があるが、アクティブ・ラーニング (Active Learning、以下 AL) の導入との関係、さらにはコロナ禍にあって、講義形態の多様化などの面からも学生の集中度を評価することは重要となる。講義を受講中の学生の表情、身振り手振りといった非言語的行動が、内的な理解状況を表していること^[2]や、学習時の姿勢から集中度を推定する研究^[3]などが行われており、これらの研究手法を参考に、測定対象と環境を決定して実験を行った。

この実験では、講義室の後方から講義室内の様子を 20 分ごとにカメラで撮影し、その姿勢から集中度を求めた。集中度は、首をかしげていれば-1、頭を抱えていれば-2 のように点数化し、1 講義時間中にその合計が-3 以下の学生を講義に集中していない非集中状態と判断した。その結果、授業への参加人数と着席位置によって、集中度は異なり、前列に着席している学生が多い講義では集中度が高く、講義室全体に後方まで学生が着席している講義では集中度が低くなることがわかった。さらには、板書形式の講義のほうが、スライドを用いた講義よりも集中度が高く、教員が一方に話をする講義よりも、学生が話す、発表する機会を設けた、AL 形式を導入している講義のほうが、集中している学生の割合が圧倒的に多いことも示された (図 3)。



図 3 撮影された講義室内の写真の例

なお、収集したセンサデータからも、集中度との関係を比較、検討したが、これらの環境情報と集中度との明確な関係は見られなかった。一件目の実験例で紹介したように、二酸化炭素濃度は、集中力の低下と関係していることが知られているため、二酸化炭素濃度の測定が有効だった可能性が考えられる。

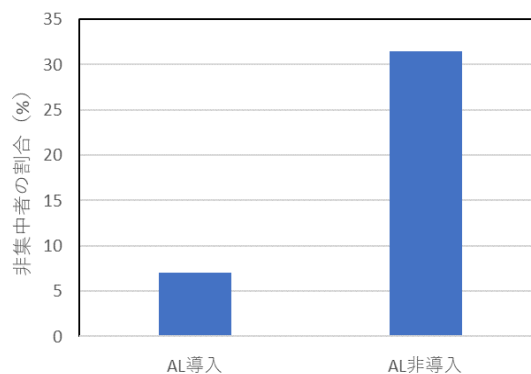


図 4 アクティブ・ラーニングの導入有無による非集中者数割合の比較

このほかにも、図書館の2階、3階の東側にあるラーニングコモンズの温湿度や照度を測定し、各階における勉強のしやすさなどの快適さを評価と比較したり、喫煙所の空気質をガスセンサと二酸化炭素センサで測定し、空気質の状態と利用者の滞在時間の関係について検討したり、と学生 アイディアから、さまざまな実験が生まれ、興味深いデータが収集、分析されており、今後も折に触れて紹介する機会を設けていく予定である。

4. まとめ

本稿では、実験科目の一環として実施している実験テーマ「IoT 環境モニタリング」での実践例を紹介した。IoT 化についてはほとんど触れなかったが、ラズパイをインターネットにつなぐことで、さまざまなセンサ情報を SNS 等で発信することができ、さらに環境を整えれば、センサ情報から周囲の換気扇や空調をコントロールするなどのフィードバック制御も可能となる。このような、より実用的な技術やシステムを、学生たち自身が、たった4週間、厳密には半日×4日間という短い期間で実装し、解析まで行えるため、学生にとっても有意義な経験となり、実験終了後に取り組む卒業研究などにも活かしてもらえることを期待している。また、この実験を通して、新たな着眼点やセンサの利用などが示されることもあり、私自身にとっても発見が多く、引き続き、このテーマの実践を続けていく予定である。

今後も、実験中であるとのお断りのメモとともに、学内のさまざまなところにセンサを設置する可能性があり、皆様のご理解とご協力をお願いしたい。

謝辞

実験科目の一テーマとして本実験を実施し、さまざまなアイディアで環境モニタリングを行い、実験データから有意義な結果を導き出してくれた学生さんたちに感謝の意を表します。また、実際に測定を行うにあたり、センサ類の設置を許可していただき、ご協力いただきました、埼玉大学図書館、埼玉大学教育推進室、埼玉大学工学部係、および設置場所を利用されている学生さん、教職員の皆さんに感謝の意を表します。今後も学内のどこかでセンサ類の設置を見かけることがあるかもしれませんが、引き続きのご協力をよろしくお願いいたします。

参考文献

- [1] 厚生労働省、建築物環境衛生管理基準について、
<https://www.mhlw.go.jp/bunya/kenkou/seikatsu-eisei10/>
- [2] 横川和章、有馬道久、教授場面における非言語的コミュニケーション-理解状況の表出と判断-、教育心理学研究、第34巻、第2号、pp. 120-129 (1986)
- [3] 亀井 諭、学習時の姿勢と行動計測による集中度合いの推定、中央大学大学院研究年報理工学研究科編、第45号 (2015)

分子動力学シミュレーションデータ解析パッケージの開発

松永 康佑

1. はじめに

DNA やタンパク質といった生体分子の分子動力学(Molecular Dynamics; MD)シミュレーションは、ミクロの分子世界の振る舞いを観察するために強力な手法であり、近年では計算顕微鏡(Computational Microscope)とも呼ばれ、実験計測を補完する観測手法として広く普及している[1]。MD シミュレーションで行なっている計算は非常にシンプルで、最初に (a)実験で観測された構造や自分でモデリングした構造を初期構造として計算機の中に構築する。(b) 原子間距離を計算してそこから各原子に働く力を計算し、 $F=ma$ に従って原子の座標を少し動かす。(c) (b)を何回も繰り返して原子・分子の座標の時系列データ(トラジェクトリと呼ばれる)を得るというステップから成っている。たったこれだけであるが、そこから生体分子の様々な振る舞いを定量的に観測することができる。例えば、どのような構造をとりえるのか？構造が変化しやすいか？どうやって構造変化するか？薬剤がどれくらい結合しやすいか？他のタンパク質と結合しやすいか？といった多くの静止室を定量化することができる。

タンパク質の物理化学の分野では、「タンパク質フォールディング」という問題がある。タンパク質はランダムな構造から自発的に決まった構造へ折れたたむ(フォールディングすること)が知られているが、なぜ短時間のうちに他の無数の構造の中から唯一の構造を探ることができるのだろうか？という問題である。これを調べるには、タンパク質がフォールディングする過程を MD シミュレーションして観察してみるのがひとつの手段であるが、10 年以上前は全原子モデルを用いた MD シミュレーションでタンパク質をフォールディングさせるくらい長い時間スケール(およそミリ秒)を達成することは夢物語であった。しかし最近では、京や富岳に代表されるスーパーコンピュータ、GPU、MD 専用計算機などを使えば、長時間の MD シミュレーションが可能となっており、多くの研究でタンパク質をフォールディングさせることに成功している。

一方で、それほど長時間の MD シミュレーションができるようになり、タンパク質のフォールディングや機能する際の構造変化のトラジェクトリが得られるようになると、別の問題が浮上してきた。それは、MD シミュレーション可能なモデルが複雑で長時間に及ぶにしたがって、得られるデータが複雑すぎて人間が解釈するのが不可能になっている、と言う問題である。例えばタンパク質フォールディングの例で言えば、シミュレーションするときちゃんとフォールディングすることは分かったが、なぜ効率良くフォールディングできるのか？という問いに答えるにはデータを解釈する必要がある。しかしながら、フォールディングシミュレーションのデータは長時間で非常に巨大になっており、人間には複雑すぎて問

題に答えることができない。そこで、最近の分子シミュレーション研究の分野では、データ解析が得意な研究グループが、早い計算機を持つ研究グループが行った長時間のシミュレーションデータを提供してもらい、データ解析だけ行ってその結果を報告する研究スタイルが確立してきている。つまり、高速な MD シミュレーションがコモディティ化するとともに、長時間シミュレーションの実現の研究よりも、元々の科学的問いに答えるためのデータサイエンスのほうへ比重がシフトしてきている。

本稿では、この問題を克服するために、我々が開発している MD シミュレーションデータ解析パッケージである MDToolbox.jl (<https://github.com/matsunagalab/MDToolbox.jl>)を紹介する。MDToolbox.jl は統計手法や機械学習を用いて MD シミュレーションデータから科学的知見を得ることを助けるパッケージであり、MD トラジェクトリの読み込み書き込みに必要な関数から、分子構造に関する特徴量の計算、更に特徴量を用いた機械学習タスクやモデリングの関数などを提供している。Julia 言語で開発されており [2]、BSD 3-Clause ライセンスのもとで Julia の公式パッケージとして承認を受けて version 0.1.0 を公開されている。以下では、MDToolbox.jl の特徴や応用例について解説する。

2. 背景知識：MD トラジェクトリと必要とされる解析

前節で説明したように、MD シミュレーションを行うと、原子・分子の座標の時系列データであるトラジェクトリがデータとして得られる。典型的な MD シミュレーションでは、原子数はおおよそ 10 万で 100 万～1000 万フレームの座標を保存するので、32 ビット浮動小数点で座標を保存したとすればトラジェクトリのデータサイズはおおよそ 1 TB ～ 10 TB となる。ではここからどのような解析を行えば、当初の科学的問いへ答えるための知見を得ることができるだろうか？大雑把にはこれを人間が理解できるくらいの少数の重要なアミノ酸残基(アミノ酸とはタンパク質を構成する 20 種類の分子ユニット)を捉えることができるくらい低次元に縮約することが目標となる。例えば、タンパク質フォールディングの問題で言えば、ランダムな構造から安定構造への進捗をよく記述できる低次元座標を探すことが目標となる。そのためには、途中で失敗せずにうまくフォールドしたデータと、失敗してランダム構造へ戻ってしまったデータとを抜き出して、それらを使って回帰や判別分析などを行うなどが考えられる。

上記のような解析を行うには何よりもまず、トラジェクトリから統計解析や機械学習へ渡す特徴量を計算することが求められる。生体分子系の特徴量としては、参照構造からの平均二乗変位(RMSD)、タンパク質を構成する特定の原子間距離、全てのアミノ酸ペアの原子間距離(距離マップやコンタクトマップと呼ばれる)、原子間の角度などが用いられる。これらを計算するためには、トラジェクトリデータのフォーマットから座標の読み込みを行い、そこから特定の原子を選択し、特徴量を計算するという一連の処理が求められる。

また、タンパク質フォールディングに限らず、モータータンパク質やポンプ、トランスポーターが機能する際には大きな構造変化が起こることで機能が発揮される。これらの動きは、

空間的に離れたアミノ酸が連携して動いたり、ローカルなイベントがグローバルな動きと連携することで実現される。これを総称してアロステリック制御と呼ぶ。アロステリック制御を解析するには、互いに動きが相関を持っているアミノ酸残基ペアを特定したり、その相関のネットワーク構造を捉えることが必要になってくる。

他に、統計解析として重要なものとして、標本データであるトラジェクトリから各状態の熱力学安定/不安定の推定量を求める、というものがある。これは自由エネルギー推定と呼ばれる。この推定は、実験で観測される熱力学量と比較するために必要であり、他にも構造がそれぞれ何%の割合で存在するのか？構造間のエネルギーバリアはどれくらい高くてその間の移動は起こり得るか？といった推定や、薬剤やタンパク質間の結合しやすさを知る上で重要である。

3. MDToolbox.jl の特徴

MDToolbox.jl は前節で紹介した背景のもとで求められる以下の機能を備えている。

1. MD データの IO : PDB、PSF、DCD、NetCDF のデータフォーマットの読み書き
2. 原子選択 : 独自型の導入により、原子番号や残基番号などで簡単に配列の部分セットを取り出し可能
3. 特徴量の抽出と相関 : 構造の重ね合わせによる RMSD 計算。特定の原子間距離や距離マップ、コンタクトマップ、各種の角度計算による特徴量抽出。各種の相関解析
4. 教師なし学習 : 主成分解析や独立成分解析などの次元縮約、クラスタリング
5. 自由エネルギー推定 : Weighted Histogram Analysis と Multistate Bennett Acceptance Ratio、カーネル密度推定
6. マルコフ状態モデリング構築のための関数 : カウントマトリックスの計算、カウントマトリックスからの最尤法による遷移確率推定、lag-time を用いて推定した遷移確率行列からの implied-timescale の計算、Viterbi アルゴリズム、Baum-Welch アルゴリズム

6 のマルコフ状態モデルとは、生体分子の構造ダイナミクスを粗視化して表現するモデリングのことを言う[3]。マルコフ状態モデルの構築では、最初にトラジェクトリのクラスタリングを行い似た構造を状態としてまとめて、その後でトラジェクトリから状態間の遷移確率を推定する。あとは生体分子の構造ダイナミクスを確率的な状態間の遷移として表現し様々な解析を行う。

MDToolbox.jl の特徴は、パッケージ内の関数同士が連携しやすいように設計されている点である。トラジェクトリを読み込んだ直後のデータ型は、`TrjArray` という独自型が採用されているが、これは Julia 固有の配列の派生型であり、原子選択がしやすいように設計されている。`TrjArray` 型のデータを特徴量を計算する関数へ渡すと、出力は Julia 固有の配列であり、それをその後な様々な解析へ引き続くことができる。例えば、トラジェクトリを読み込んで、原子選択を行い、特徴量として距離マップを計算し、それを主成分解析などの

次元縮約へ引き継いでプロットするという一連の連携が自然にできるようになっている。このような一連の解析を Jupyter notebook 上に対話的に行うことだけでなく(図 1)、バッチ処理として複数のノードを用いて並列で処理することも一部関数では行うことができる。

もう一つの MDToolbox.jl の特徴は、全ての関数が Julia 言語のみで書かれている点である。Julia 言語の特徴は優れた JIT コンパイラにあり高速に実行できることが知られている。これは先行研究として存在する Python ベースの MD トラジェクトリ解析パッケージ (MDTraj[4]やMDAnalysis[5])と比べて大きく異なる点である。例えばRMSDの計算では、RMSDが最小になるような回転行列を求める必要があり、現在では四元数ベースのQCP法を用いることが多い[6]。元々のQCP法のコードはC言語で書かれているが、これを例えばPythonへそのまま移植すると10倍以上遅くなってしまう。従って、既存のPythonパッケージでは、Pythonはグルーコードとしてだけ用いてQCP法はCやC++などで書かれていることが多い。一方でJuliaの場合だと、オリジナルのC言語のコードをほぼそのままJuliaへ移植することができ、実行速度もオリジナルの2倍以下である。全ての関数が一つの言語で書かれていることは、特にメンテナンスや開発効率の点でメリットがある。これは教育の観点からも同じことが言える。学生が関数を使うだけでなく、自分で中身に興味を持ったり、改造してみようと思った場合に、Pythonベースのパッケージでは、C++などで書かれた別の言語を習得しなくてはならず、ハードルが高い。しかしMDToolbox.jlの場合は、そのまま同じJulia言語で中身を改造することができる。

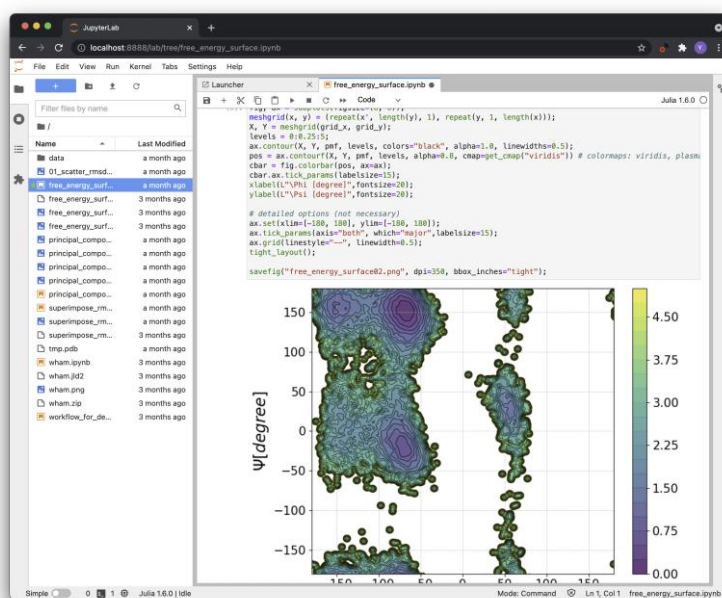


図 1 : Jupyter notebook 上で MDToolbox.jl を動かして対話的に解析の様子

4. MDToolbox.jl の応用例

MDToolbox.jl の応用例として、アデニル酸キナーゼ(Adenylate Kinase; AdK)というタンパク質の解析例を紹介する(この notebook は以下に公開されている https://github.com/matsunagalab/hmm_tutorials)。AdK は酵素であり、ある基質(ADP+AMP)が結合すると、それに特異的に応答し、オープン構造からクローズド構造へと大規模な構造変化を起こし基質に対する触媒能を得ることが知られている[7]。以下では、AdK の粗視化モデルの MD シミュレーション[8]を MDToolbox.jl で解析する流れを紹介する。

まず、MD トラジェクトリを読み込み、参照構造としてクローズド構造からの RMSD を計算する(図 2A)。この結果を時間フレームの関数としてみると、AdK が時間発展とともに非常に大きな構造変化を起こしており、クローズド構造とオープン構造を行ったり来たりしていることがわかる。次に特徴量として距離マップを計算し、これを主成分解析で処理して可視化する(図 2B)。この結果、RMSD で見えていなかった隠れた状態が複数見つかり、一番右のクローズド構造と、一番左のオープン構造以外にもいくつか中間構造があることがわかる。MDToolbox.jl の関数を連携して適用することがこのような解析を短時間で行うことができる。更に、主成分空間でクラスタリングや状態間の遷移を行うと、AdK 構造ダイナミクスのマルコフ状態モデルを構築して遷移に関する確率解析を行うことができる(図 3)。確率解析の詳細は本稿の範囲を越えるので割愛する。

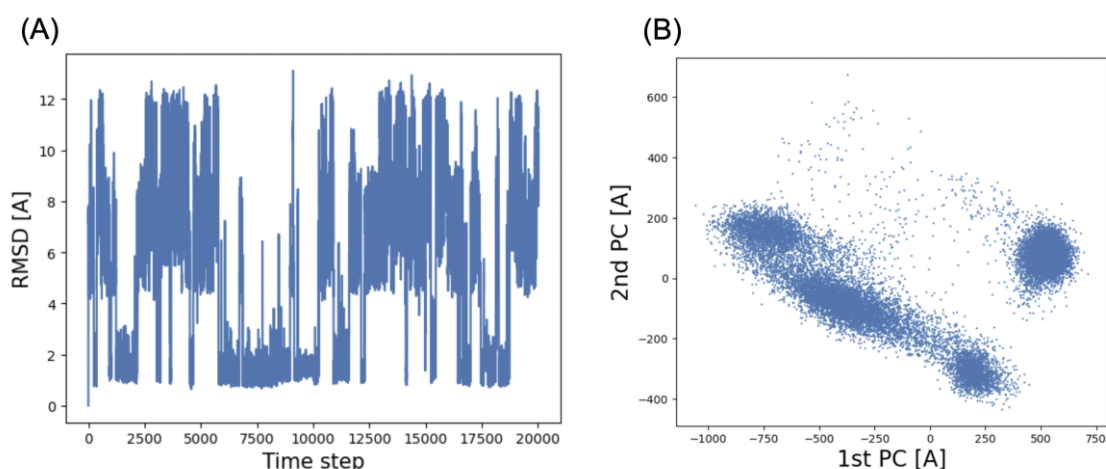


図 2 : アデニル酸キナーゼの(A)RMSD と(B)距離マップの主成分解析の結果

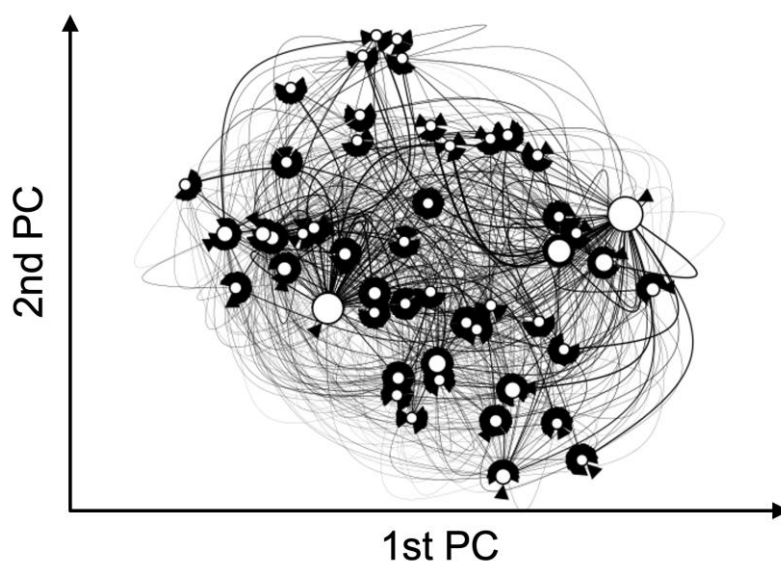


図 3 : アデニル酸キナーゼのマルコフ状態モデル

5. まとめ

本項では、我々が開発している MD シミュレーションデータ解析パッケージである MDToolbox.jl について解説した。今回は MDToolbox.jl 内で閉じた解析例について紹介したが、MDToolbox.jl で得られる特徴量などのデータ型は Julia 固有のものであり、Julia エコシステムの一つとして、他のパッケージと連携した応用も考えることができる。例えば、深層学習パッケージである Flux.jl や、他の機械学習系のパッケージと連携することでより高度な解析・モデリングへの展開が期待される。

6. 謝辞

本研究は以下の受託研究の成果の一部である。文部科学省 スーパーコンピュータ「富岳」成果創出加速プログラム「全原子・粗視化分子動力学による細胞内分子動態の解明」(HPCI 課題番号: hp200135)、JST CREST「高速原子間力顕微鏡 1 分子計測のデータ同化による生体分子 4 次元構造解析法の開発」(課題番号: JPMJCR1762)。

参考文献

- [1] Brooks, C. L., Case, D. A., Plimpton, S., Roux, B. et al. Classical molecular dynamics. *J. Chem. Phys.* **154**, 100401 (2021).
- [2] Bezanson, J., Edelman, A., Karpinski, S. and Shah, V. B. Julia: A Fresh Approach to Numerical Computing. *SIAM Rev.* **59**, 65–98 (2017).
- [3] Husic, B. E. and Pande, V. S. Markov State Models: From an Art to a Science. *J. Am. Chem. Soc.* **140**, 2386–2396 (2018).
- [4] McGibbon, R. T., Beauchamp, K. A., Schwantes, C. R., Wang, L.-P. et al. MDTraj: a

modern, open library for the analysis of molecular dynamics trajectories. 2.

- [5] Michaud-Agrawal, N., Denning, E. J., Woolf, T. B. and Beckstein, O. MDAnalysis: A toolkit for the analysis of molecular dynamics simulations. *J. Comput. Chem.* **32**, 2319–2327 (2011).
- [6] Theobald, D. L. Rapid calculation of RMSDs using a quaternion-based characteristic polynomial. *Acta Crystallogr. A* **61**, 478–480 (2005).
- [7] Matsunaga, Y., Fujisaki, H., Terada, T., Furuta, T., et al. Minimum Free Energy Path of Ligand-Induced Transition in Adenylate Kinase. *PLoS Comput. Biol.* **8**, e1002555 (2012).
- [8] Kobayashi, C., Matsunaga, Y., Koike, R., Ota, M., et al. Domain Motion Enhanced (DoME) Model for Efficient Conformational Sampling of Multidomain Proteins. *J. Phys. Chem. B* **119**, 14584–14593 (2015).