

AMR 法を導入した格子ボルツマン法による イルカのフリースイミングの複数 GPU シミュレーション

A Multi-GPU Simulation with AMR Method for a Dolphin Free-swimming Using Lattice Boltzmann Method

- 山口大輝, 東工大, 〒 152-8550 東京都目黒区大岡山 2-12-1, E-mail : yamaguchi@sim.gsic.titech.ac.jp
渡辺勢也, 東工大, 〒 152-8550 東京都目黒区大岡山 2-12-1, E-mail : watanabe@sim.gsic.titech.ac.jp
青木尊之, 東工大, 〒 152-8550 東京都目黒区大岡山 2-12-1, E-mail : taoki@gsic.titech.ac.jp

Daiki Yamaguchi, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8550, Japan

Seiya Watanabe, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8550, Japan

Takayuki Aoki, Tokyo Institute of Technology, Meguro-ku, Tokyo 152-8550, Japan

A simulation for a dolphin free-swimming has been studied on multi-GPU of TSUBAME3.0. Lattice Boltzmann Method with cumulant collision operator is employed for stable computation for high Reynolds number flows. For more efficiency and accuracy, tree based AMR (Adaptive Mesh Refinement) method has been introduced. Solid body and fluid coupling problem is solved by using direct-forcing immersed boundary method. The computational time is approximately 24 hours for 2.86 physical seconds with 32 GPUs. The swimming speed of a dolphin reaches 68 percent (5.7 m/s) of the experimental result (8.4 m/s). The thrust force is still larger than drag force and the maximum speed is estimated to be 10.0 m/s (larger than 8.4 m/s). Currently the upstroke and downstroke of fluke are assumed to be same and we conclude that the previous deformation model should be improved.

1. 緒言

イルカは尾びれを動かして水中を高速で遊泳できるが、そのメカニズムは未だ完全には明らかになっていない。動物学者の Gray(1936) は、イルカが 20 ノット (10.3 m/s) で泳いだという観測から、イルカにかかる抗力を求めたところ、イルカの筋肉量から計算される推進力よりも大きくなる、つまり、イルカは 20 ノットという高速では遊泳できないという結論を出した。これは Gray のパラドックス⁽¹⁾と呼ばれる。今日では、Gray のパラドックスは誤りであったという見方が強いが、Gray の問題提起以降、数多くの研究者がイルカの高速度遊泳のメカニズムの解明を行ってきた。このメカニズムが解明されれば、水中遊泳ロボットの高速化、効率化などに応用可能である。

数値シミュレーションを用いた水生生物の遊泳運動に関する研究は数多く行われている。イルカの遊泳をシミュレーションによって解析する場合、実験と同様に広い計算領域が必要となる。一方で、イルカの複雑な形状を表現するために、物体近傍では十分細かい格子解像度も必要になる。また、イルカは高速で遊泳する為、流れは高いレイノルズ数の乱流になり、数値安定性の問題が生じる。本稿では、メモリアクセスの局所性が高い格子ボルツマン法を複数 GPU を用いて実装し、また AMR(Adaptive Mesh Refinement) 法を導入することで、高精度かつ高効率にイルカのフリースイミングの計算を行う。

2. 計算手法

2.1 格子ボルツマン法

格子ボルツマン法は McNamara⁽²⁾ によって提案された、流体を各格子点上で衝突、並進する仮想的な分子の集合と見なし、その速度分布関数の時間発展方程式を解く手法である。支配方程式である離散化されたボルツマン方程式は以下のように表される。

$$f_{ijk}(\mathbf{x} + \mathbf{c}_{ijk}\Delta t, t + \Delta t) = f_{ijk}(\mathbf{x}, t) + \Omega_{ijk} \quad (1)$$

添字 ijk は速度分布関数の方向で、 $(i, j, k) \in \{-1, 0, 1\}^3$ であり、 \mathbf{c}_{ijk} は速度分布関数の速度で、 $c = \Delta x / \Delta t$ とすると、 $\mathbf{c}_{ijk} = (c_i, c_j, c_k)^T$ である。 Ω_{ijk} は衝突項であり、本稿では、SGS モデルを用いない陰的な LES として作用し、高レイノルズ数でも数値安定性の高い Cumulant モデル⁽³⁾を使用した。

2.2 Direct-forcing immersed boundary method

Direct-forcing immersed boundary method は直交座標系で複雑形状物体を扱う際に有効な手法である。Fig.1 で示されるように、物体と流体の境界をラグランジュ的なマーカーの集合で表現し、周囲の流体から速度を内挿することでマーカー上での速度を求める。その速度から non-slip 条件を満たすような体積力を計算し、周囲の流体に分散させることで、物体と流体の相互作用を表現する。内挿と分散は離散化された 3 点参照のデルタ関数で行う。

本稿では体積力の計算を反復的に行うことで non-slip 条件を精度良く満たすことができる Multi Direct-Forcing Method(MDFM)⁽⁴⁾を採用した。物体に加わる力は体積力の反力の合力であり、モーメントと合わせて以下のような式で表される。

$$\begin{aligned} \mathbf{F} &= - \sum_b \mathbf{F}_b \Delta V_b \\ \mathbf{T} &= - \sum_b (\mathbf{x}_b - \mathbf{x}_c) \times \mathbf{F}_b \Delta V_b \end{aligned} \quad (2)$$

\mathbf{F}_b , ΔV_b はそれぞれ b 番目のマーカー \mathbf{x}_b にかかる力、及び検査体積であり、 \mathbf{x}_c を回転の中心とする。本稿では物体を三角形ポリゴンで表現し、その重心にマーカーを設置する。検査体積は三角形の面積 Δs_b と格子幅 Δx の積であり、 Δs_b は以下の条件を満たす必要がある。

$$\Delta s_b < \frac{\Delta x^2}{2} \quad (3)$$

2.3 物体の運動

物体の並進及び回転の運動方程式を解くことで、その運動をシミュレーションする。物体の重心速度を \mathbf{v}_g 、角速度を $\boldsymbol{\omega}_g$ 、体積を V 、質量を m_g 、慣性テンソルを \mathbf{I}_g 、流体の密度を ρ とすると、運動方程式は

$$\begin{aligned} \frac{d}{dt} (m_g \mathbf{v}_g) &= \mathbf{F} + (m_g - \rho V) \mathbf{g} \\ \frac{d}{dt} (\mathbf{I}_g \boldsymbol{\omega}_g) &= \mathbf{T} \end{aligned} \quad (4)$$

である。式 4 をオイラー法による時間積分で更新する。

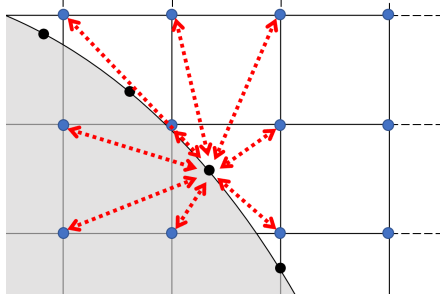


Fig. 1: Direct-forcing immersed boundary method. gray area: structure. blue points: fluid. black points: Lagrangian marker. red arrow: interpolating velocity and spreading force.

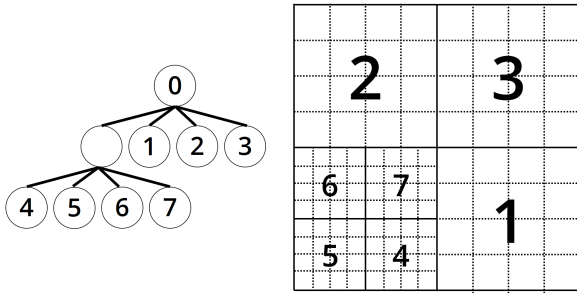


Fig. 2: Data and mesh structure of Cartesian computational domain.

2.4 Adaptive Mesh Refinement 法

複雑な形状の物体を表現するためには多数のポリゴンが必要になり、式3からも分かる通り、格子幅も細かくする必要があります。しかし、全計算領域で高解像度の格子を取ることは計算の非効率性に繋がる。効率化、省メモリ化を考えると、物体近傍には細かい格子を割り当て、遠方では解像度を粗くする必要があります。そこで、本稿では Adaptive Mesh Refinement 法を採用した。

Fig.2 に示される通り、木構造アルゴリズムに従い、格子間隔を $1/2, 1/4, 1/8 \dots$ と再帰的に分割することにより、任意の局所空間に任意の格子解像度を割り当てることができる。2次元の場合は1つのリーフに対応する領域が細分化されると、4つのリーフが生成される四本木のデータ構造になる。3次元の場合は8つのリーフが生成される八本木のデータ構造になる。

木構造アルゴリズムに基づき複雑に細分化された AMR 格子を3次元的に領域分割するため、Fig.3 のように、空間充填曲線により1次元空間にマッピングする。空間充填曲線に従ってリーフの数を数えていき、各 GPU が計算するリーフ数が均一になるように曲線を切断し、領域分割を行う。

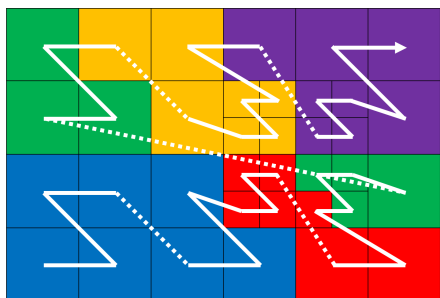


Fig. 3: Dynamic domain partitioning using space-filling curve.

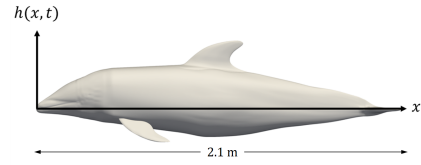


Fig. 4: Coordinate of deformation.

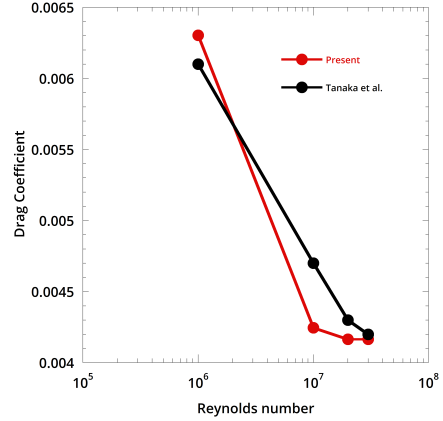


Fig. 5: Drag coefficient of dolphin against Reynolds number. red: cumulant LBM. black: CFD result (Tanaka et al.) (5).

3. イルカのモデル

本稿では、イルカのモデルとして Tanaka ら⁽⁵⁾ によって測定及び作成されたカマイルカの3次元形状モデルを用いた。測定値に基づき、イルカの体長 L は 2.1 m, 表面積 S は 1.87 m^2 , 体積 V は 0.111 m^3 とし、密度 ρ は水と同じ 1000.0 kg/m^3 とした。

通常の座標系とは別に、Fig.4 に示されるような鼻先を原点とした座標系を定義し、実験的な観測⁽⁶⁾⁽⁷⁾ から得られた遊泳時の背骨の変形の近似式 $h(x, t)$ をモデルに加えることで、イルカの変形を定義する。

$$h(x, t) = A(x)w(x, t)$$

$$A(x) = A_{\max} \left(0.21 - \left(\frac{x}{L}\right) + 1.1 \left(\frac{x}{L}\right)^2 + 0.35 \left(\frac{x}{L}\right)^8 \right)$$

$$w(x, t) = \sin \left(2\pi \left(\left(\frac{x}{L}\right) - \left(\frac{t}{T}\right) \right) \right) \quad (5)$$

ここで、測定値⁽⁷⁾⁽⁸⁾に基づき、尾びれの最大振幅 A_{\max} を $0.15L$, イルカが尾びれを振る周期 T を 0.270 s とした。

4. 検証計算

本計算の妥当性を検証するために、一様流の中にイルカを静止させ抗力係数を計算し、既往研究⁽⁵⁾との比較を行う。ここで、イルカの抗力係数は次式で表される。

$$C_D = \frac{F_D}{\frac{1}{2}\rho U^2 S} \quad (6)$$

F_D は抗力であり、流入流速 U は、レイノルズ数が $1.0 \times 10^6, 1.0 \times 10^7, 2.0 \times 10^7, 3.0 \times 10^7$, となるように、 $0.48 \text{ m/s}, 4.76 \text{ m/s}, 9.52 \text{ m/s}, 14.3 \text{ m/s}$ と設定した。計算領域は $14.0 \text{ m} \times 4.0 \text{ m} \times 2.0 \text{ m}$ とし、イルカ近傍には 2.0 mm の解像度の格子を、遠方には最大で 256 mm の

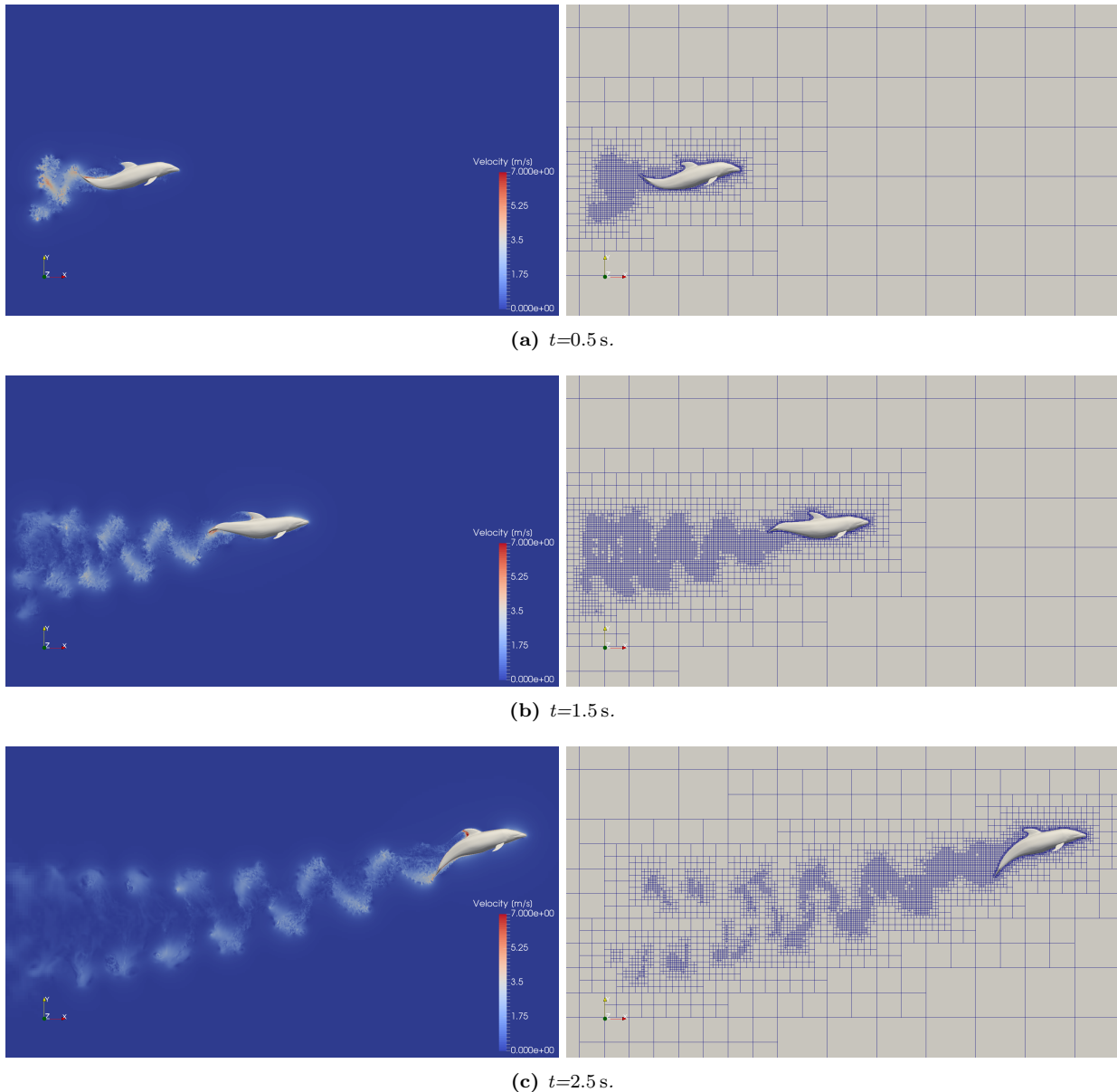


Fig. 6: Snapshot of velocity magnitude and block structured mesh.

格子を割り当てる．計算結果を Fig.5 に示す．同じモデルを使用した Tanaka らの結果と同様に，レイノルズ数が 1.0×10^5 から 1.0×10^6 の間で抗力係数が顕著に減少するという傾向が見られたことから，本研究の妥当性が示された．

5. イルカのフリースイミング

計算領域は $24.0\text{ m} \times 16.0\text{ m} \times 2.0\text{ m}$ とする．格子解像度は前節と同様に，イルカ近傍では 2.0 mm ，遠方では 256.0 mm とし，時間刻みは 2.0 mm の格子解像度に対して， $2.29 \times 10^{-5}\text{ s}$ とした．GPU は 32 台使用し，物理時間を 2.86 s 進めるのに，1 日程度の計算時間を要した．

シミュレーション結果を Fig.6, Fig.7 に示す．イルカは真っすぐには進まず，斜め上方向に進んでいくことが伺える．その進行方向の遊泳速度の時刻歴は Fig.8 のような結果となった．既往研究⁽⁵⁾によると，体長 2.1 m のイルカが周期 0.27 s で尾びれを振って 8.4 m/s で遊泳したことが確認されている．本稿では $t=2.87\text{ s}$ の時点で，その 68% 程度の速度が出ていることが分かる．

これ以上の速度は出せない，つまりこの速度で抗力と推進力が釣り合っているかを確認するために，推進力の推定を行った．

推進力と抗力の時刻歴を Fig.9 に示す．抗力は 4 節で求めた抗力係数を線形補間し，各時刻での抗力係数を求めることで算出した．横軸は時刻を周期で除した値であり，灰色の半周期が downstroke，白色の半周期が upstroke である．

推進力の半周期ごとの平均値は 389 N 程度ではほぼ横ばいになっており，抗力は遊泳速度の二乗に比例して増加していき，イルカにかかるトータル力は時間とともに減少していることが伺える．しかし，9 周期目でも合力の平均値は 221 N と正值であり，イルカは加速していることが分かる．Fig.9 から予測される最高速度は 10.0 m/s であり，これは測定値の 8.4 m/s を上回っている．シミュレーションでは upstroke と downstroke で得られる推進力が同程度であるが，既往研究⁽⁹⁾によると，イルカは upstroke よりも downstroke の方が尾びれを大きく振っており，upstroke の推進力は downstroke の推進力より小さくなる．これはイルカの変形式 (式 5) が upstroke と downstroke の非対称性を考慮していないためだと考えられる．

遊泳時にイルカがどの部位で推進力を得ているかを調べるために，イルカにかかる力の分布を Fig.10 に示した．

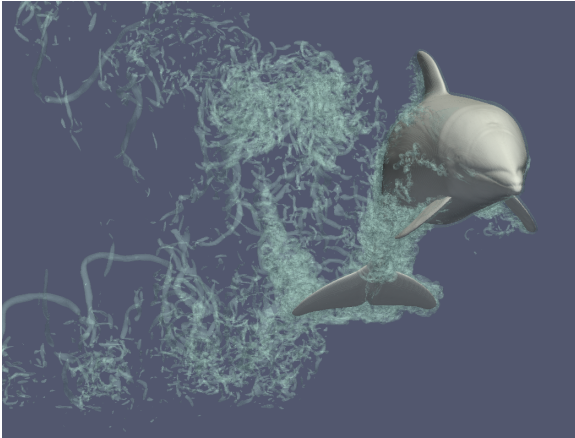


Fig. 7: Vortex structure around a dolphin.

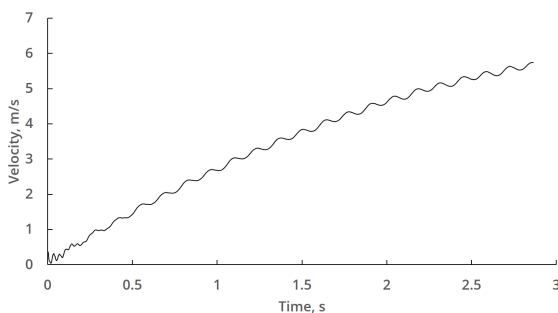


Fig. 8: Time history of swimming velocity.

x 軸は推進方向であり, 図左がイルカを横から見た図, 図右上が背びれ側から見た図, 図右下が腹側から見た図となっている. 時刻は Fig.9 の 8 周期と 9 周期の間であり, upstroke と downstroke において, 推進力がそれぞれ最大, 最小となる時刻とした.

イルカは推進力の大部分を尾びれを振ることで得ていることが分かる. また, 胴体部の変形こそ異なるが, Peak, Valley 時の尾びれの迎え角は upstroke と downstroke において, それぞれ一致しており, 前節で指摘した対称性が表れている.

6. 結言

AMR 法を導入した格子ボルツマン法の複数 GPU 計算を用いて, イルカのフリースイミングのシミュレーションを行った. 一様流中にイルカのモデルを設置し, 抗力係数を既往研究と比較することで妥当性を確認した. イルカの遊泳速度から抗力を計算し, 推進力の推定を行った.

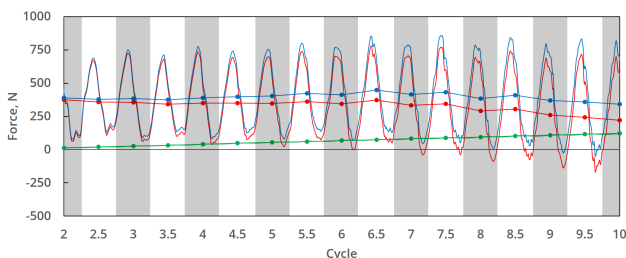


Fig. 9: Forces acting on a dolphin and these average of a half cycle. red line: total force. blue line: thrust force. green line: drag force. purple line: gravity and buoyancy. each color's circle: average of a half cycle (upstroke or downstroke). gray region:downstroke. white region: upstroke.

た. イルカのモデルにかかる力の分布から, 尾びれが推進力の大部分を生み出していることを確認した.

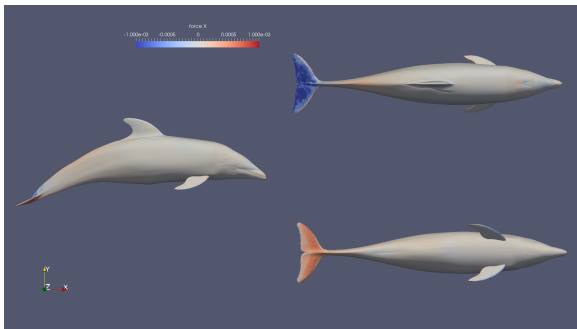
謝辞

本研究は科学研究費補助金・基盤研究 (S) 課題番号 19H05613 「自由界面を含む混相流の革新的数値流体シミュレーション」, 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 (HPCI) 課題番号 hp190130 および学際大規模情報基盤共同利用・共同研究拠点 (JHPCN) 課題番号 jh190054 から支援を頂いた. 計算には東京工業大学学術国際情報センターのスパコン TSUBAME3.0 を利用させて頂いた. 記して謝意を表す.

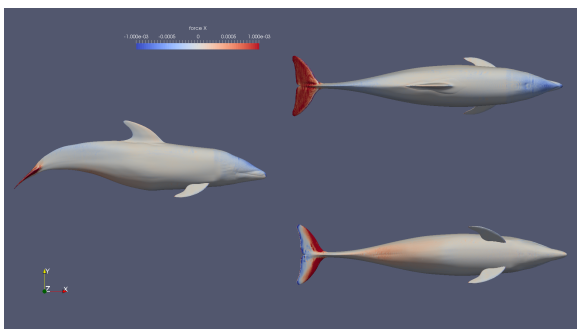
また, 東京工業大学・工学院の田中博人准教授には, イルカの詳細な形状モデルを提供していただき, 心から感謝の意を表す.

参考文献

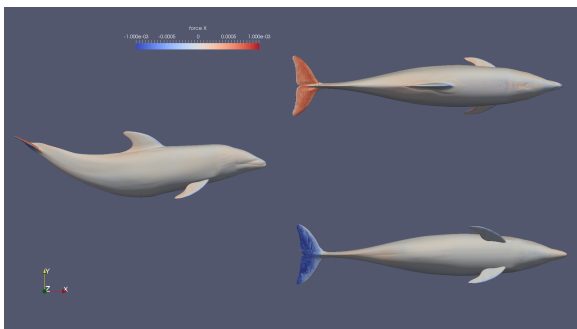
- (1) J Gray. Studies in animal locomotion: Vi. the propulsive powers of the dolphin. *Journal of experimental biology*, Vol. 13, No. 2, pp. 192–199, 1936.
- (2) Guy R McNamara and Gianluigi Zanetti. Use of the boltzmann equation to simulate lattice-gas automata. *Physical review letters*, Vol. 61, No. 20, p. 2332, 1988.
- (3) Martin Geier, Martin Schönherr, Andrea Pasquali, and Manfred Krafczyk. The cumulant lattice boltzmann equation in three dimensions: Theory and validation. *Computers & Mathematics with Applications*, Vol. 70, No. 4, pp. 507–547, 2015.
- (4) Zeli Wang, Jianren Fan, and Kun Luo. Combined multi-direct forcing and immersed boundary method for simulating flows with moving particles. *International Journal of Multiphase Flow*, Vol. 34, No. 3, pp. 283–302, 2008.
- (5) Hiroto Tanaka, Gen Li, Yusuke Uchida, Masashi Nakamura, Teruaki Ikeda, and Hao Liu. Measurement of time-varying kinematics of a dolphin in burst accelerating swimming. *PLoS one*, Vol. 14, No. 1, p. e0210860, 2019.
- (6) Kourosh Shoele and Qiang Zhu. Drafting mechanisms between a dolphin mother and calf. *Journal of theoretical biology*, Vol. 382, pp. 363–377, 2015.
- (7) Evgenii Vasil'evich Romanenko. *Fish and dolphin swimming*. Pensoft Publishers, 2002.
- (8) 内田雄介, 李根, 中村昌史, 田中博人, 劉浩. J027022 ハイスピードカメラを用いたイルカのジャンプの運動解析 ([j027-02] 生物規範メカニクス・システム: 生物の運動, 力学及びミメティクス (2)). 年次大会 2013, pp. J027022-1. 一般社団法人 日本機械学会, 2013.
- (9) 田中一郎, 永井實. 抵抗と推進の流体力学-水棲動物の高速遊泳能力に学ぶ-, 扇興社. 1996.



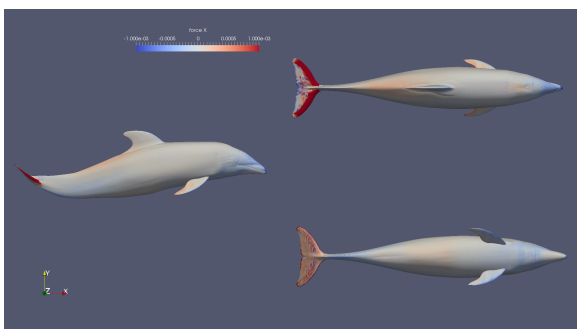
(a) Valley (upstroke).



(b) Peak (upstroke).



(c) Valley (downstroke).



(d) Peak (downstroke).

Fig. 10: Force distribution. left: side view, right top: top view, right bottom: bottom view of dolphin.