

学生フォーミュラの空力開発における 計算資源不足による結果への影響について

サウサンプトン大学 航空宇宙工学科
田崎 雄大



1. 背景
2. 設定の範囲
3. 結果
4. 結論



課題

- CFDを正しく理解して扱うのはハードルが高い：乱流モデルの物理的背景など
- 計算資源がいる：車全体を正しいメッシュ数で回すと1億4千万セルとかになる
- Softwareへのアクセスがない：STARCCM+が使えない、Ansys Fluentのメッシュ数に制限があるなど



それでも空力開発にCFDは必要

CFDを通じて得られる情報

- コンセプトの比較
- SCL, SCD, 空力バランスの絶対量
- エアロマップによる姿勢への空力特性の調査
- パラメータ最適化



少ない計算資源で何が正しく得られる？



理想（ざっくりとですが）

① ラップタイムシムをもとに目標決定



② モデリング



③ CFDの結果は目標を達成してる？ No → ②に戻る

Yes └→ ④ 風洞実験は同じ傾向を示す？ No → ②に戻る & CFDの設定の評価

Yes └→ ⑤ 製作して搭載



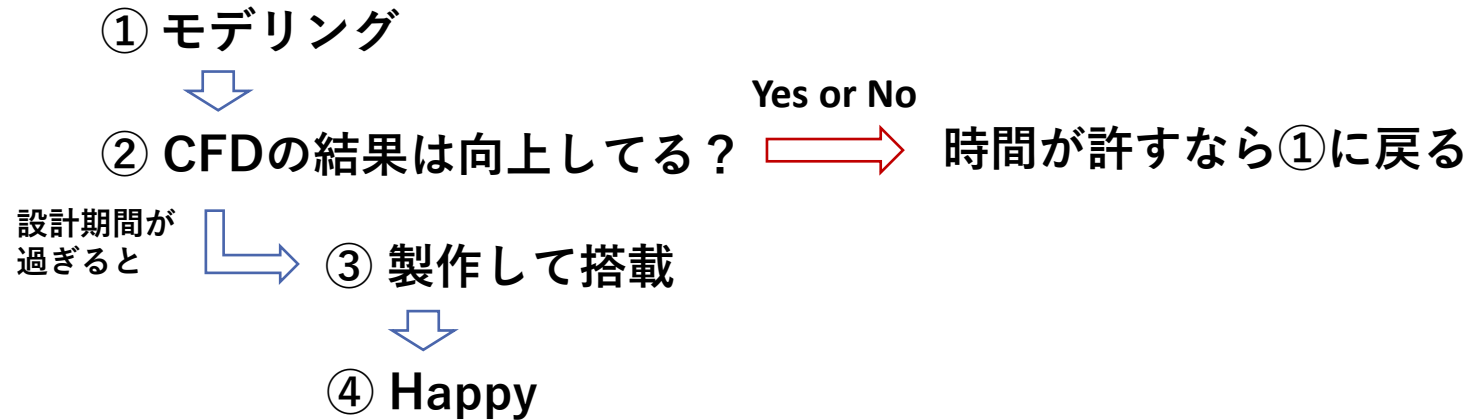
⑥ 実車の挙動はよくなった？ No → ①に戻る & 風洞実験の評価 & CFDの設定の評価

Yes └→ Happy & 次の改善点を検討し同じサイクルを回す

実際は学生フォーミュラではこんなサイクルはできなくて



現実（※僕の妄想）



問題点

- 前提であるCFDによる相対評価が正しいという仮定が崩れると**成り立たない**
- CFDの正当性を確認する時間、および方法がない（CFDが正しい確証があるならこの話は参考にならないかも）
- 空力性能の評価を正しく行えるフェーズが存在しない（風洞の話は後半に少し）

CFDが間違ってた時の具体的な影響は？



相対比較への影響

- 流れ場が正しく評価できないと、**間違ったコンセプトに誘導されてしまう**
- サイドポッドの流入&タイヤの後流などはCFDの結果に大きく依存する

絶対量のミスリードによる影響

- Lap time simなどでエアロの正当性を評価するとき、SCLが真値と違うと**結論が変わる**かも
- 空力バランスのミスリードによる空力的に不安定な設計

パラメータ最適化

- 剥離ギリギリのラインを攻めてるつもりが現実では剥離してる
- ガーニーフラップが機能していない
- ドライバーのパフォーマンスや気候の変動を考慮すると実車試験での細かい設定は現実的でない



パフォーマンスチームからの要求

- エアロによる重量増20kgに対してタイムの短縮を正当化するにはSCLが-3必要

CFDを回すと

- 初期設計がSCL=-2.8だったのでダウンフォースを増してSCL = -3.2にした
- エアロつけることでタイムは短縮するはず

でもCFDが正しくなくて

- 変更後の真値はSCL = -2.6だった
- エアロつけることでタイムは増える

結論が違ってくるのは困る



問題点

- 前提である、CFDによる相対評価が正しいという仮定が崩れると成り立たない
絶対値は間違っても相対評価はできるかもしれない
- CFDの正当性を確認する時間、および方法がない（正しい確証があるならこの話は参考にならないかも）
お手本（RANSでできる限界）があればいい
- 空力性能の評価を正しく行えるフェーズが存在しない

じゃあお手本作って
相対評価がどこまで成立するかみてみよう



目的

- 現実的なセッティングとその制限の提案
- データベースの作成

モチベーション

- 現状役に立つデータベースがない（日産の資料は正しいけど計算資源的に現実的でない）
- 計算資源と時間に余裕がないと、データベース作成ができない

具体的なゴール

- 乱流モデルの比較
- メッシュ数による相対評価への影響の調査
- 使用するうえでの注意事項、および制約のまとめ



- ソフトウェア：SolidWorks Flow Sim, Ansys Fluent, Openfoam, クラウド系のSimScaleなど
- 追記（30/7/2023）：Simscaleの日本展開は先日発表されてた
- 人的資源：CFDのセッティングに最低 1 人割ける
 - 計算資源：Windowsのワークステーション or ノートパソコン = 8-16 CPU core
 - 最大計算時間：24h / sim
 - 知識：流体力学と数値計算の基礎

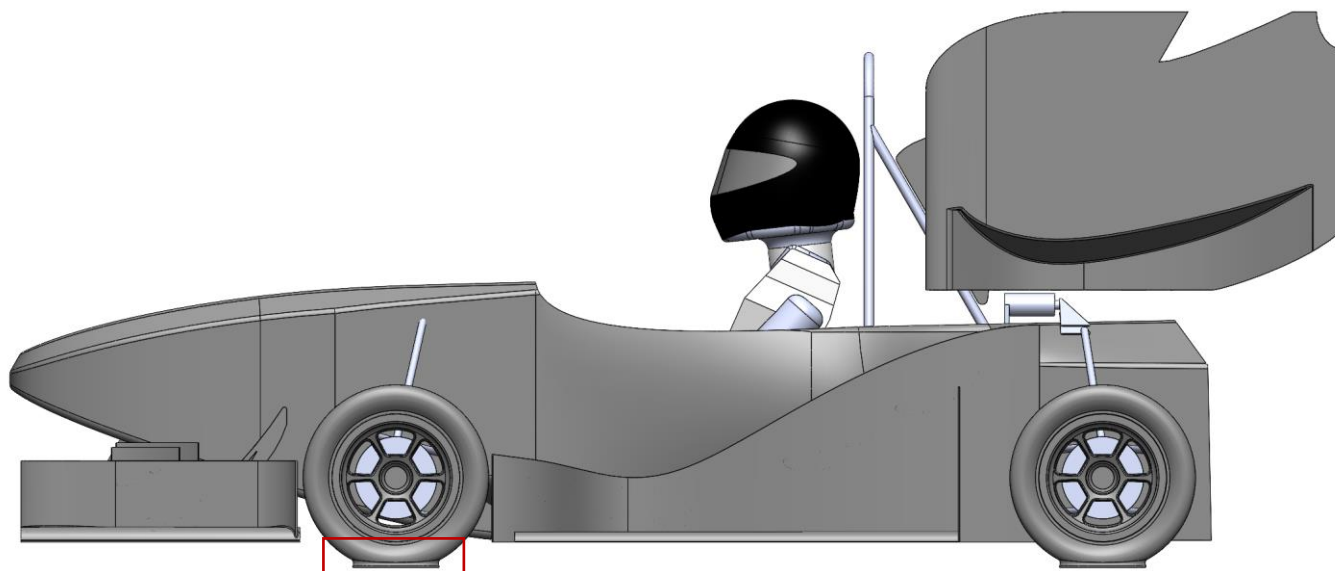
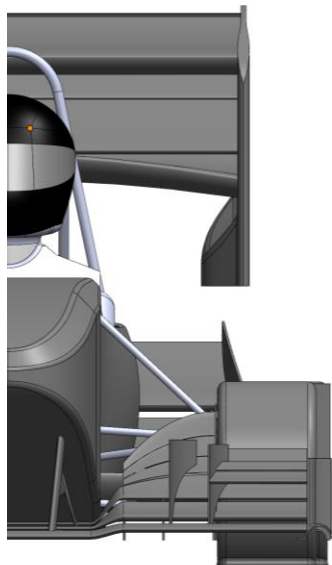
目次

1. 背景

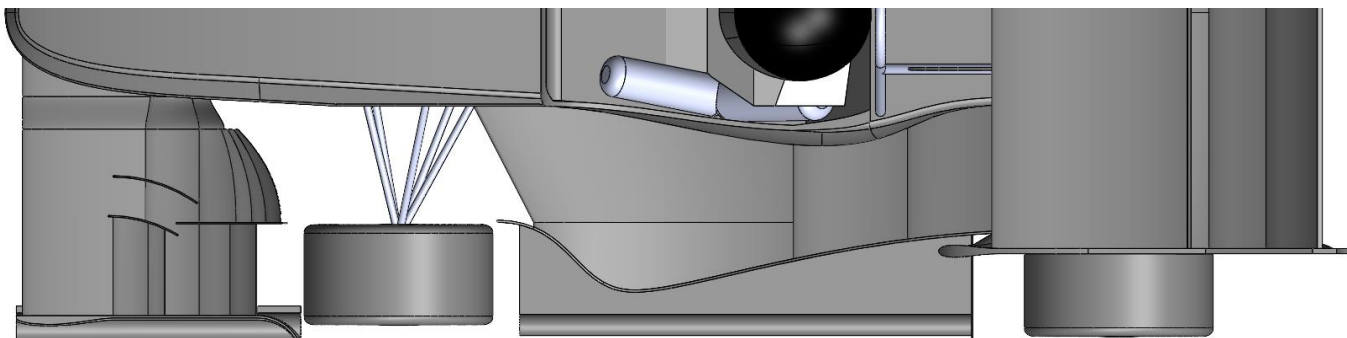
2. 設定の範囲

3. 結果

4. 結論

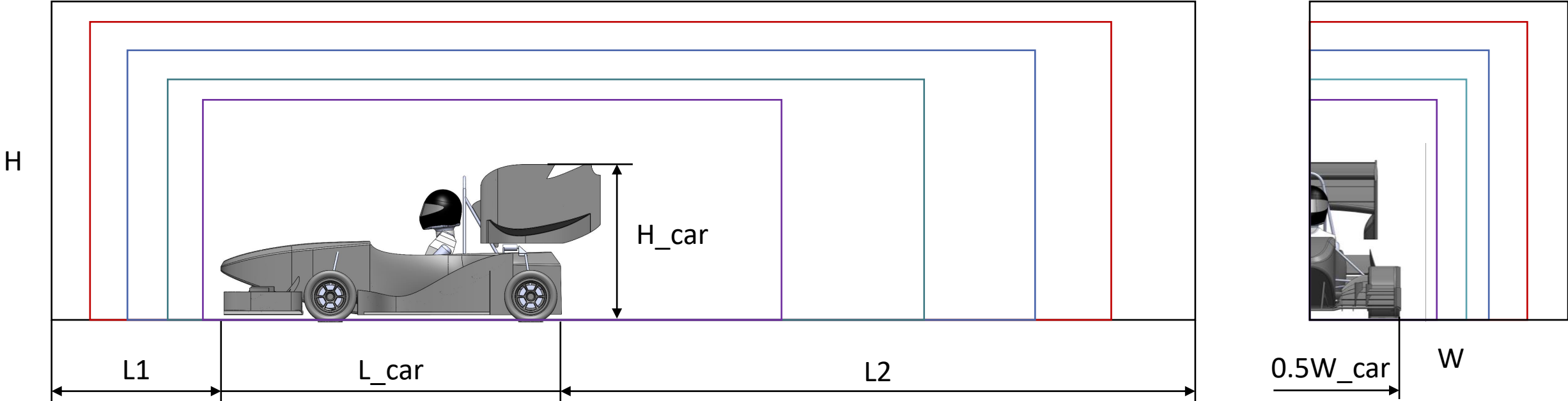


コンタクトパッチ

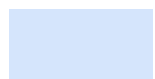
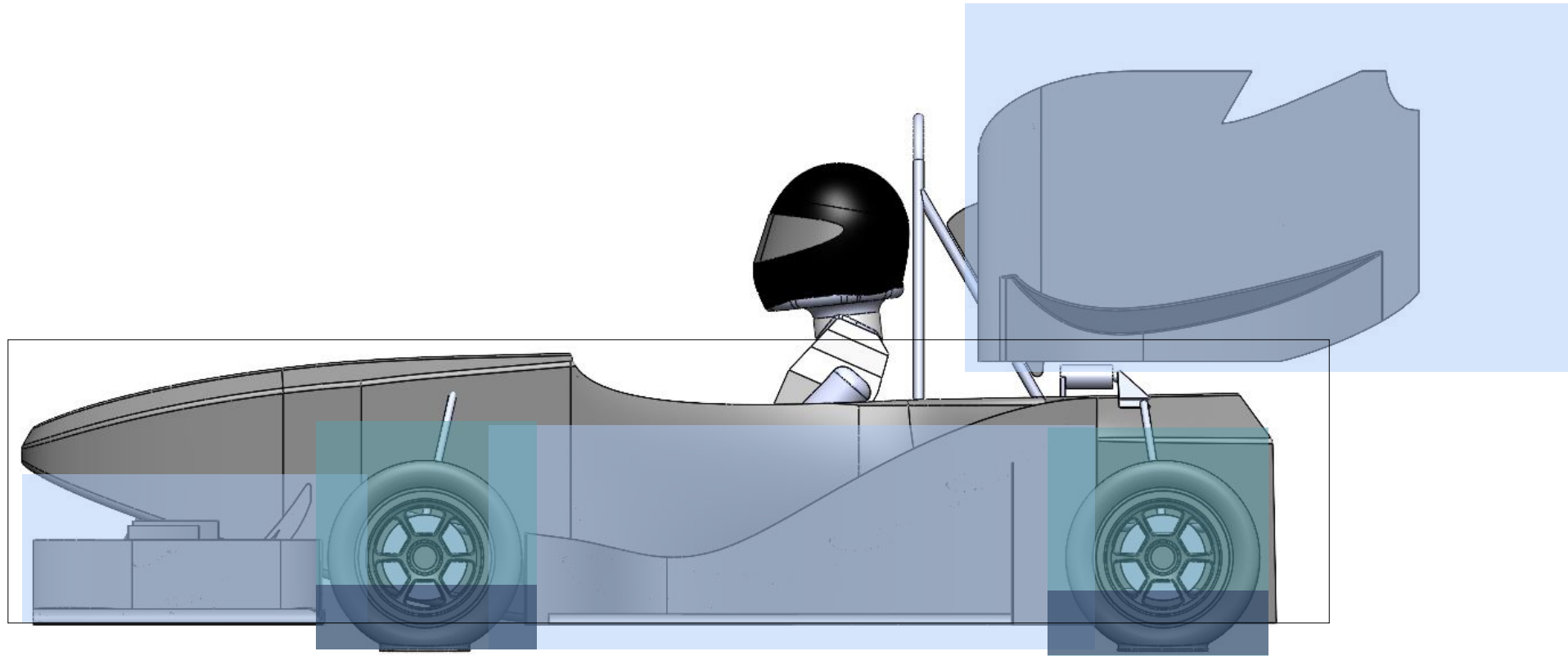


追記 (30/7/2023)
このコンタクトパッチは
すごく質が悪いみたい。
大きすぎる。





	L1	L2	H	W
XL	0.5L_car	1L_car		
L	2L_car	5L_car		
M	2H_car	3H_car		
S	2W_car	3W_car		



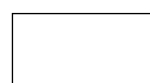
Aero refinement



Tyre refinement



Contact patch refinement



Chassis refinement



使用するソフトウェア

- Siemens STARCCM+

精度

- 2次精度（対流項のみ風上差分）
- 数値は倍精度（Double precision）を使用

境界層の設定

- 厚み：40mm（シャシ）, 20mm（エアロ、タイヤ、ドライバー）, 5mm（サス、ロールフープ）
- Y^+ は5Mのみ30 = 1 mm, それ以外は1 = 0.01 mm
- Prism Layerの層数は各メッシュごとに変更：最小が4, 最大が25

収束条件

- 全ての残差が 10^{-4} 以下

追記（30/7/2023）
残差を使わない収束条件も
意外とメジャーみたい



メッシュ成長率

- Volume mesh : Very slow
- Surface mesh: Slow

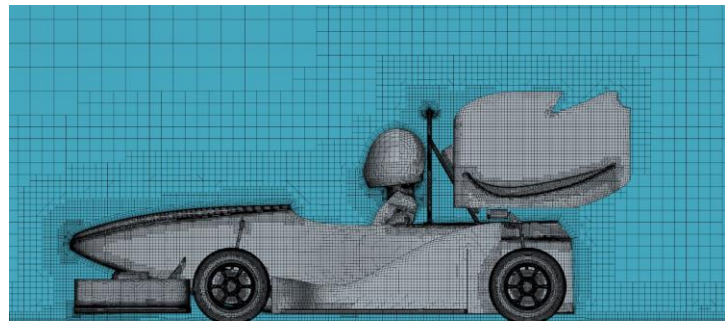
境界条件

- 直線走行を仮定しSymmetryとする
- Inlet : 10, 15 m/s, Turbulent intensity 5%, Viscosity ratio 10
- Outlet: Pressure outlet
- タイヤには接線方向の速度条件、床も10, 15m/s
- 側面、および上面はすべり条件

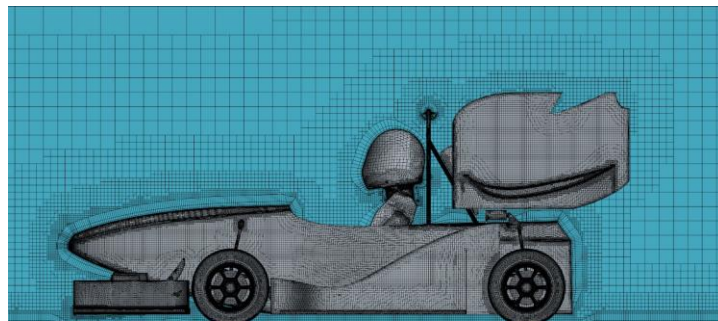
乱流モデル

- Realizable K-Epsilon
- K-Omega SST

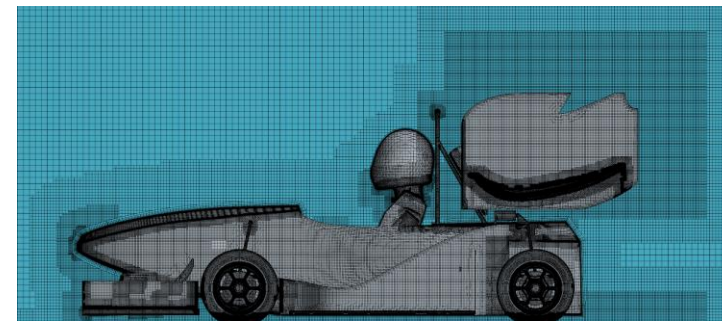
メッシュの密度と外観



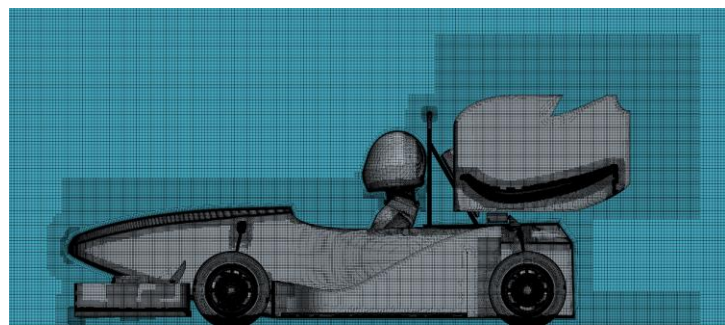
5M



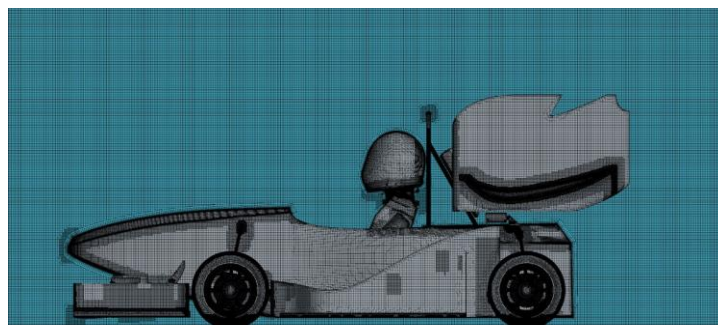
10M



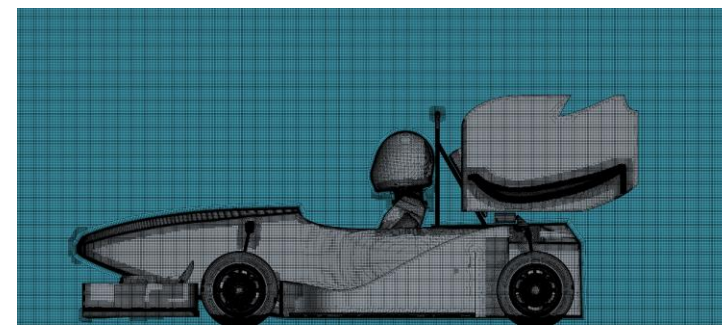
30M



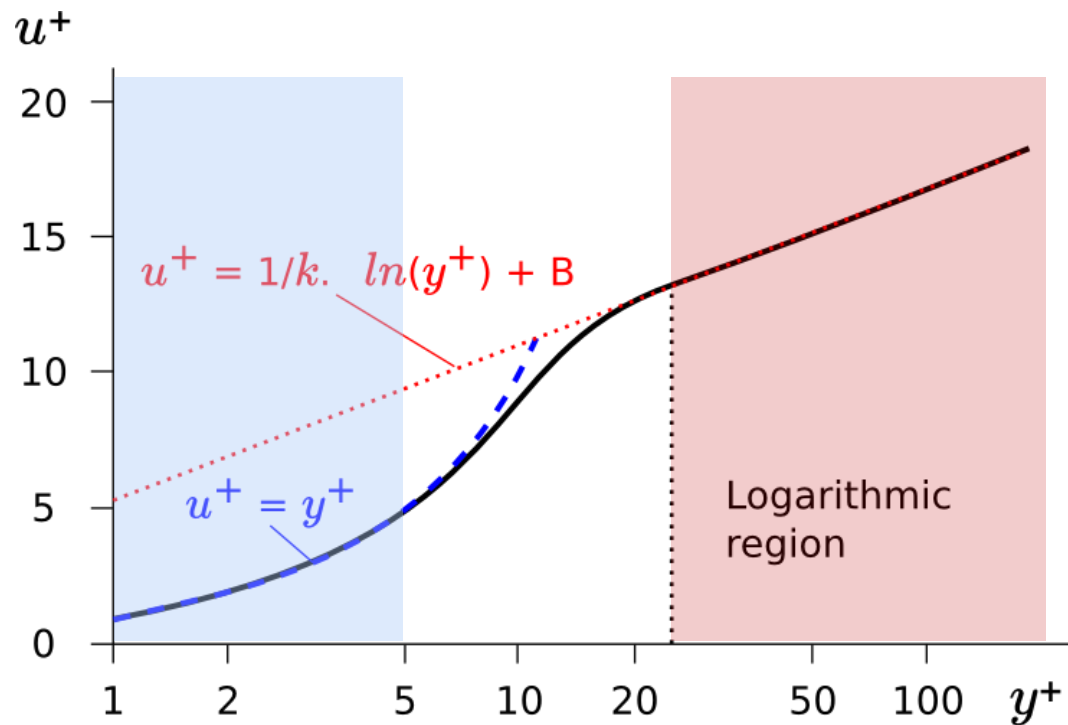
40M



55M



70M



普遍的な境界層の速度分布

境界層に支配される物理現象

- 壁面せん断応力 \equiv 摩擦抵抗
- 流れの剥離 \equiv 圧力抵抗, ダウンフォースに直結

➡ 要するにレースカーで見たい現象全部

剥離を捉えるなら

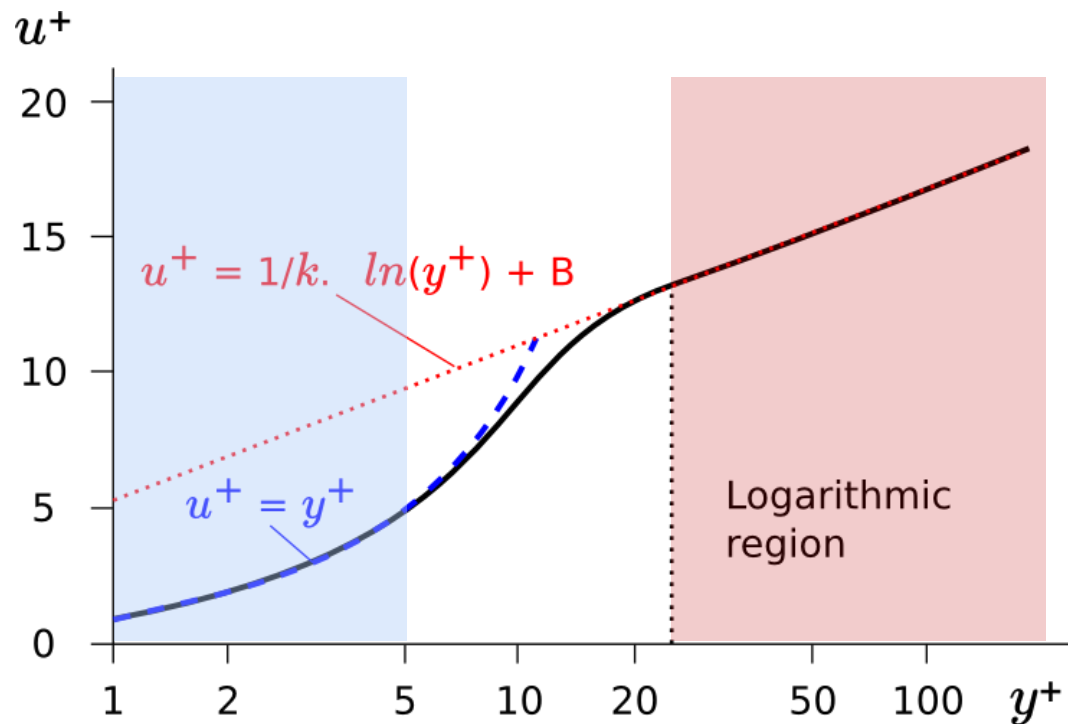
- 初めのメッシュは $y^+ = 1$ 付近にほしい
- 具体的に 10 m/s, 3 m の平板だと 0.01 mm に相当
- y^+ は無次元数で流速に合わせて実スケールは小さくなる
- $y^+ = 5$ までは許容

追記 (30/7/2023)

Blended wall function とか使えば
そこまで気にしなくていいかも

より現実的には

- どの乱流モデルも DES と比較すると剥離予測ができない



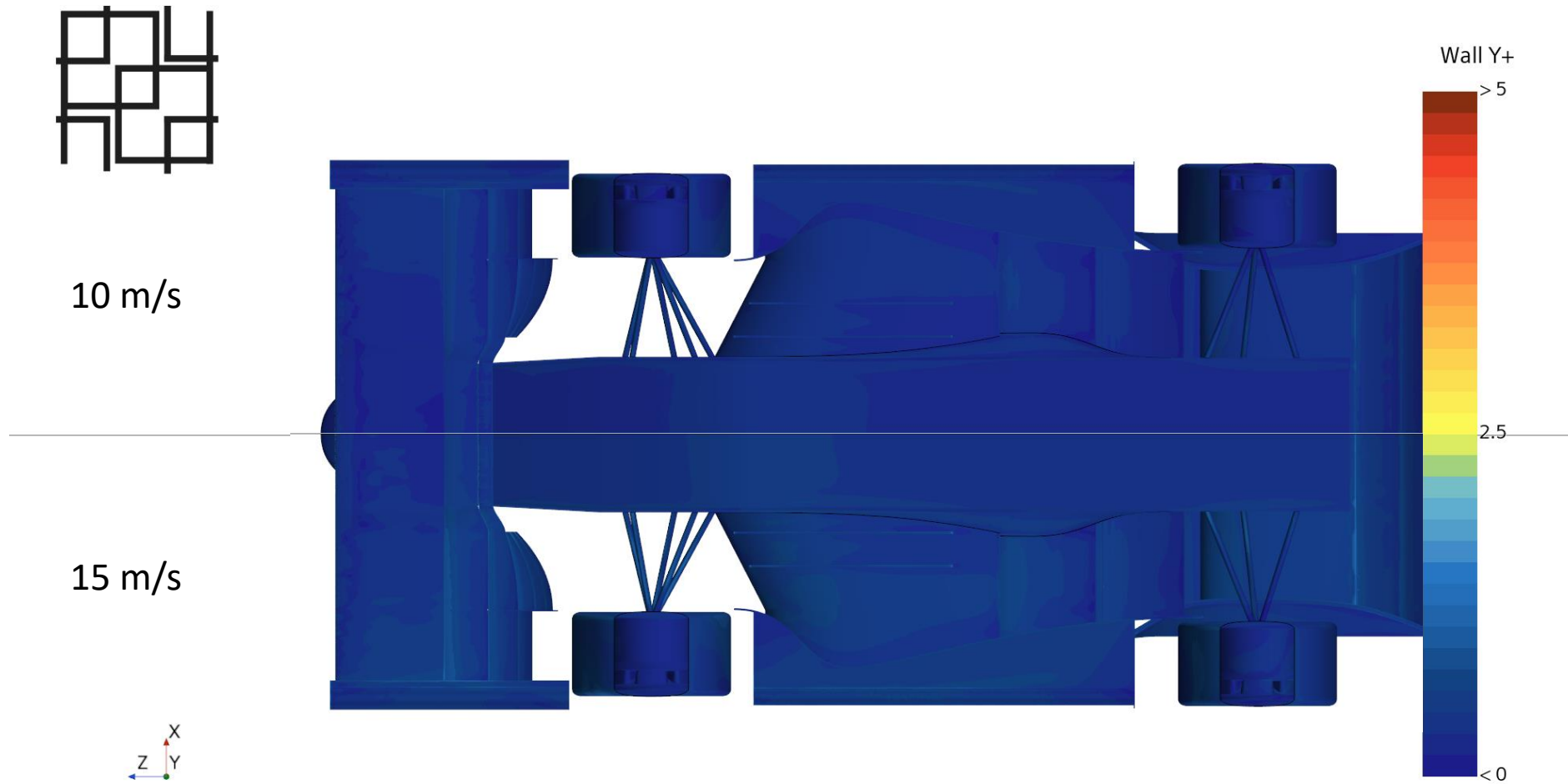
普遍的な境界層の速度分布

メッシュ的な話

- 第一層の厚みは指定可能
- 境界層厚みも指定する必要がある
- 残りは成長率かPrism layerの層の数
- 成長率は1.2以下が好ましい
- 20層ぐらいあると問題ない

計算資源の制約下における対処

- y^+ を諦める
- 成長率を1.5などにする
- 境界層厚みを薄く設定する



$y^+=1$ をターゲットにしている



ソフトウェア的な制約

- 乱流モデルがK-Epsilonしかない：一般にK-Epsilonモデルは剥離流れの予測には不向き
- 境界層メッシュ（Prism layer）が存在せず、境界層はlog法則の壁関数でしか扱えない
- Windowsでしか回らない（HPCに突っ込めない）
- デフォルトの関数が少ない（圧力係数や揚力係数は自分で定義しないといけない）
- 用意されていない場の関数は定義できない（ y^+ ）
- 各スキームの詳細が不明（一次精度なのか二次精度なのか、緩和係数の設定など）
- 単精度計算なのか倍精度計算なのか？（ChatGPT曰く倍精度、でもネットで裏がとれない）
- Iterationに対する残差が出てこない？（travelは対流時間）
- 遅い（あと大規模計算のスケーラビリティが不明）



乱流モデルと境界層以外は大きな問題はない

目次

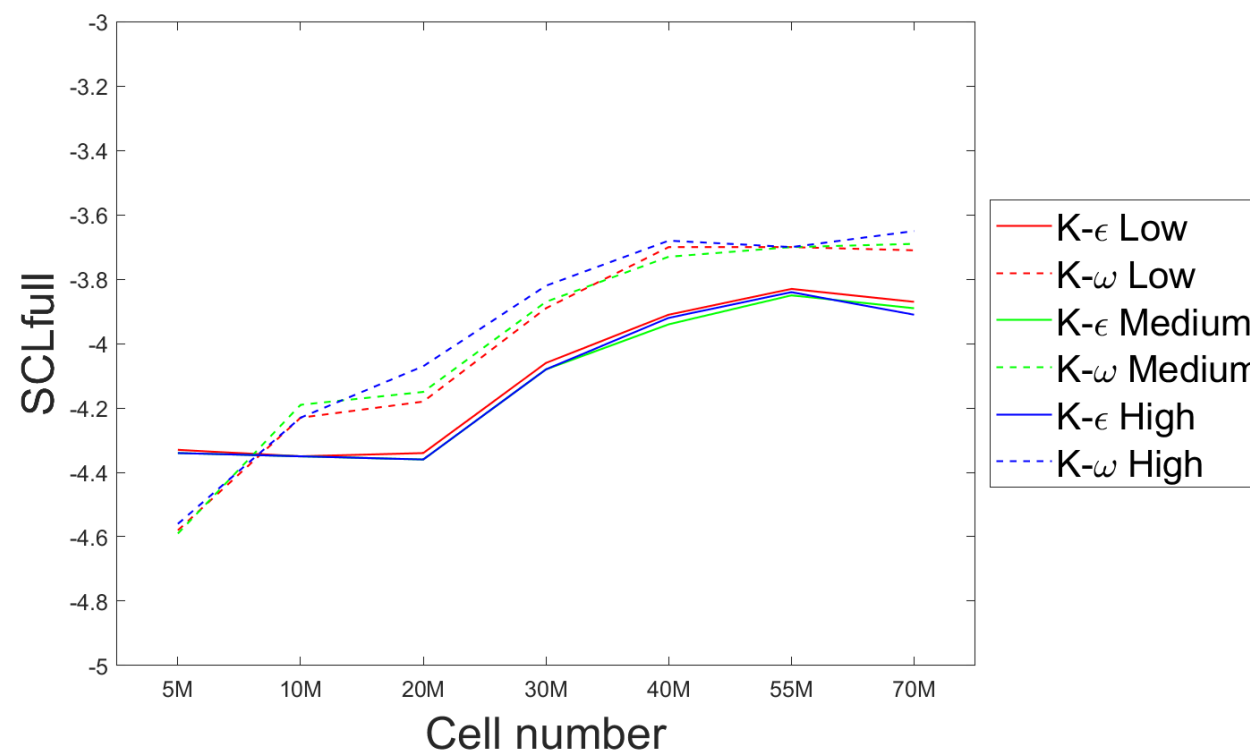
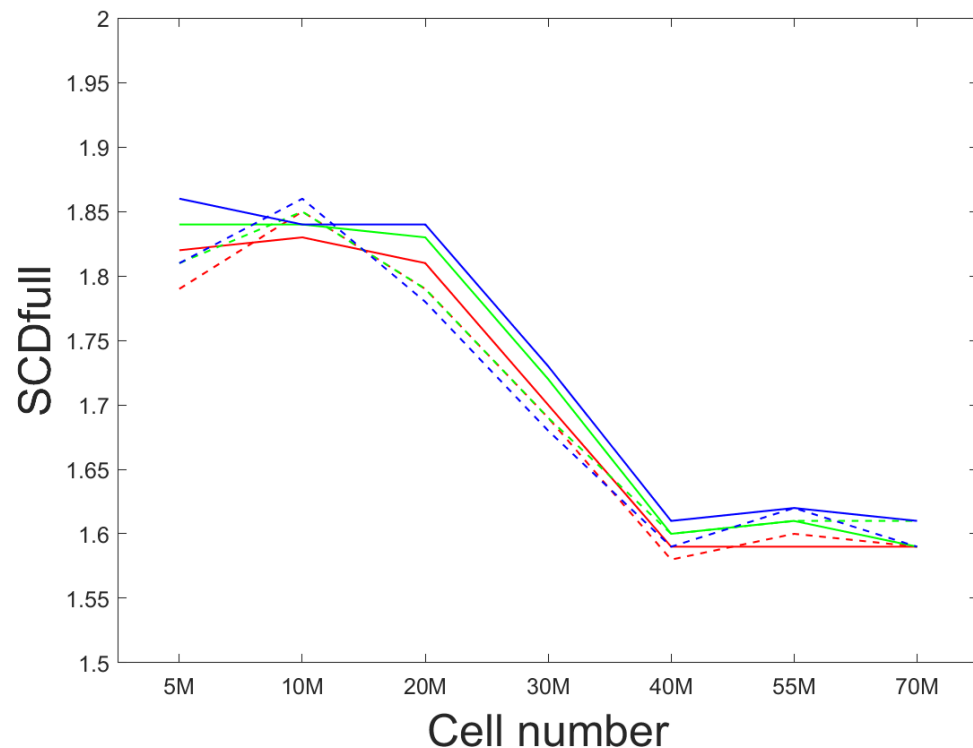
1. 背景

2. 設定の範囲

3. 結果

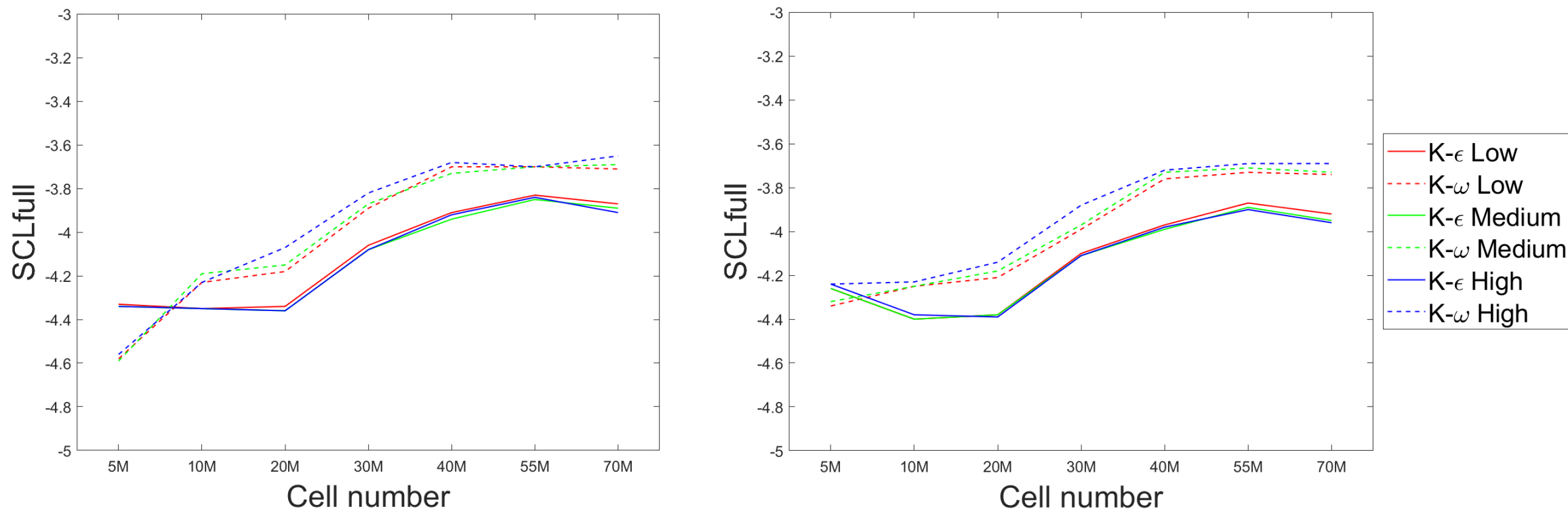
4. 結論

数値的結果の傾向：SCLとSCD



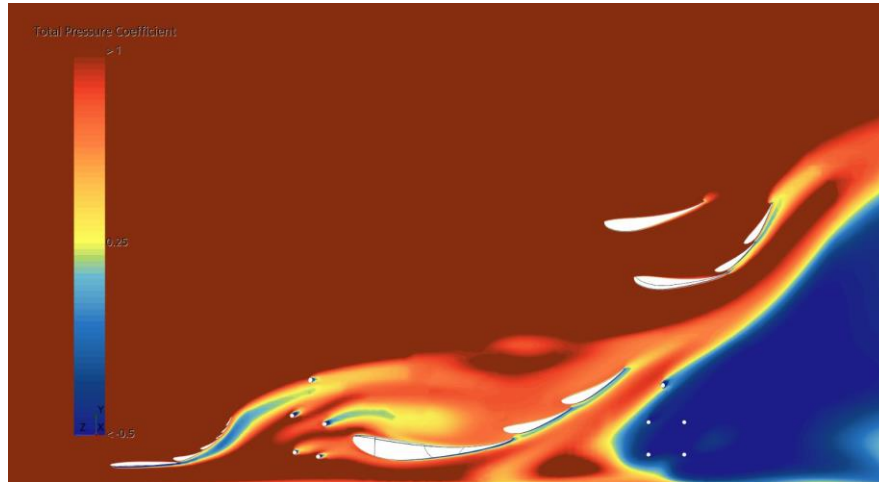
- SCL値が5Mから70Mで0.8も変動している
- サイドウィングの迎え角による相関も30Mも境目に変化している
- 乱流モデルによる差はメッシュ数の増加に伴い小さくなる

数値的結果の傾向：SCLと速度（Re数）



- SCL値が5Mから70Mで0.8も変動している
- サイドウィングの迎え角による相関も30Mも境目に変化している
- 乱流モデルによる差はメッシュ数の増加に伴い小さくなる

メッシュ数と乱流モデルによる結果の違い



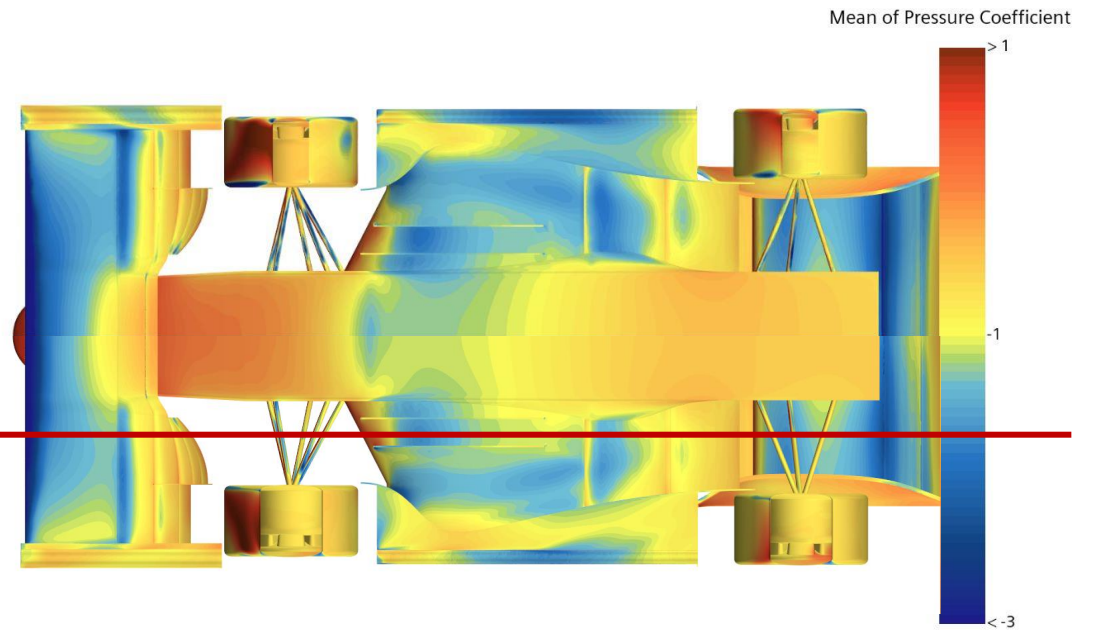
CpT Plot: 5M K-Epsilon



CpT Plot: 70M K-Omega SST

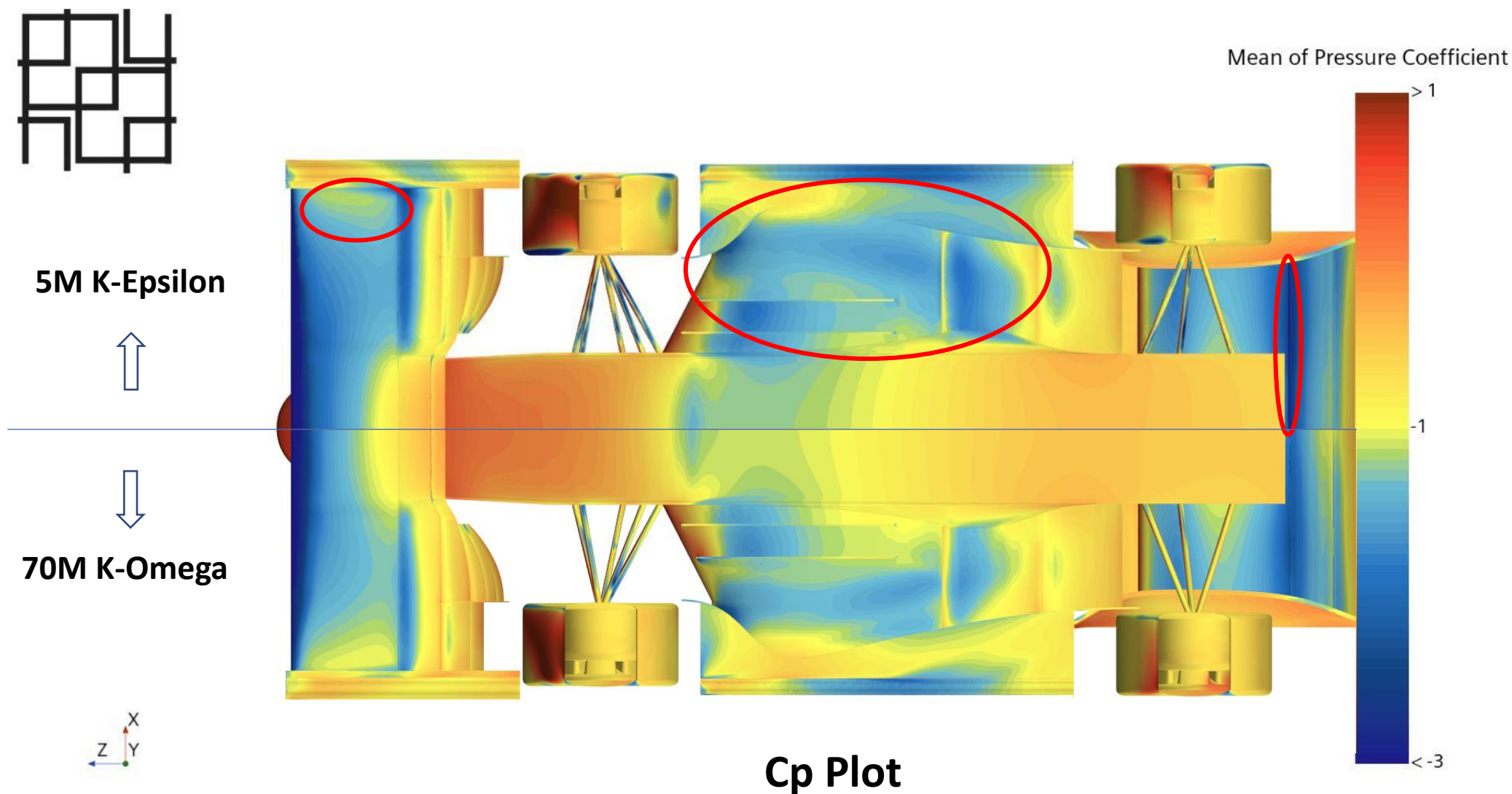


スライス面

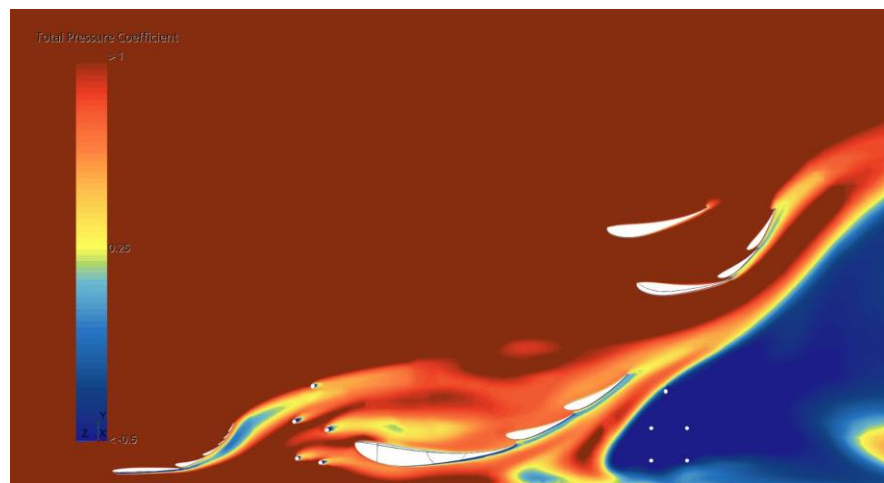


Cp Plot
上 : 5M K-Epsilon
下 : 70M K-Omega

メッシュ数と乱流モデルによる結果の違い



メッシュ数のみによる結果の違い



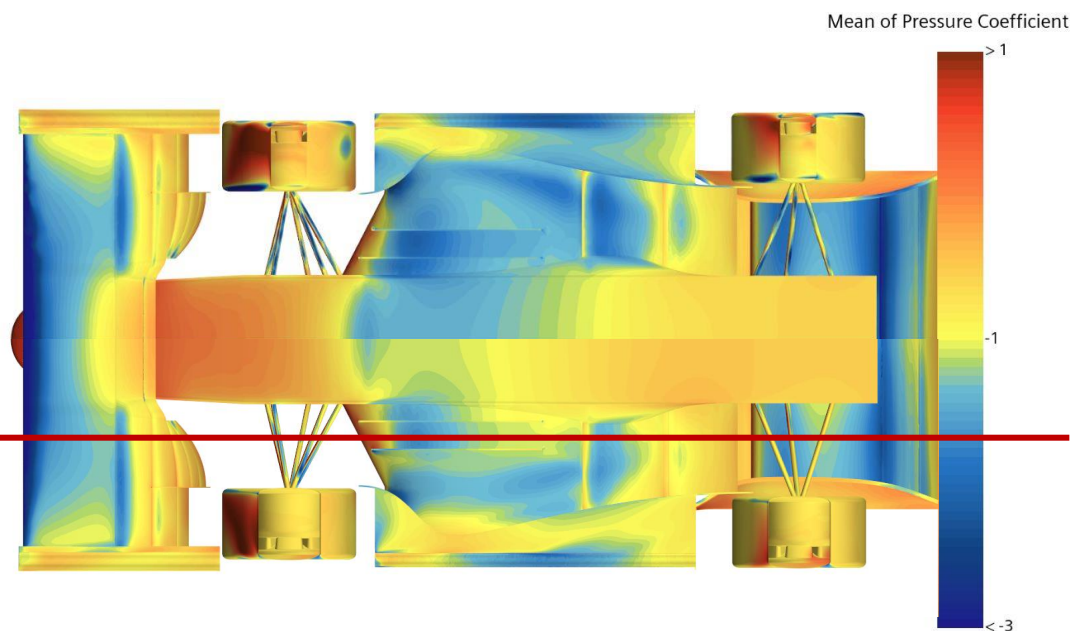
CpT Plot: 5M K-Omega SST



CpT Plot: 70M K-Omega SST



スライス面

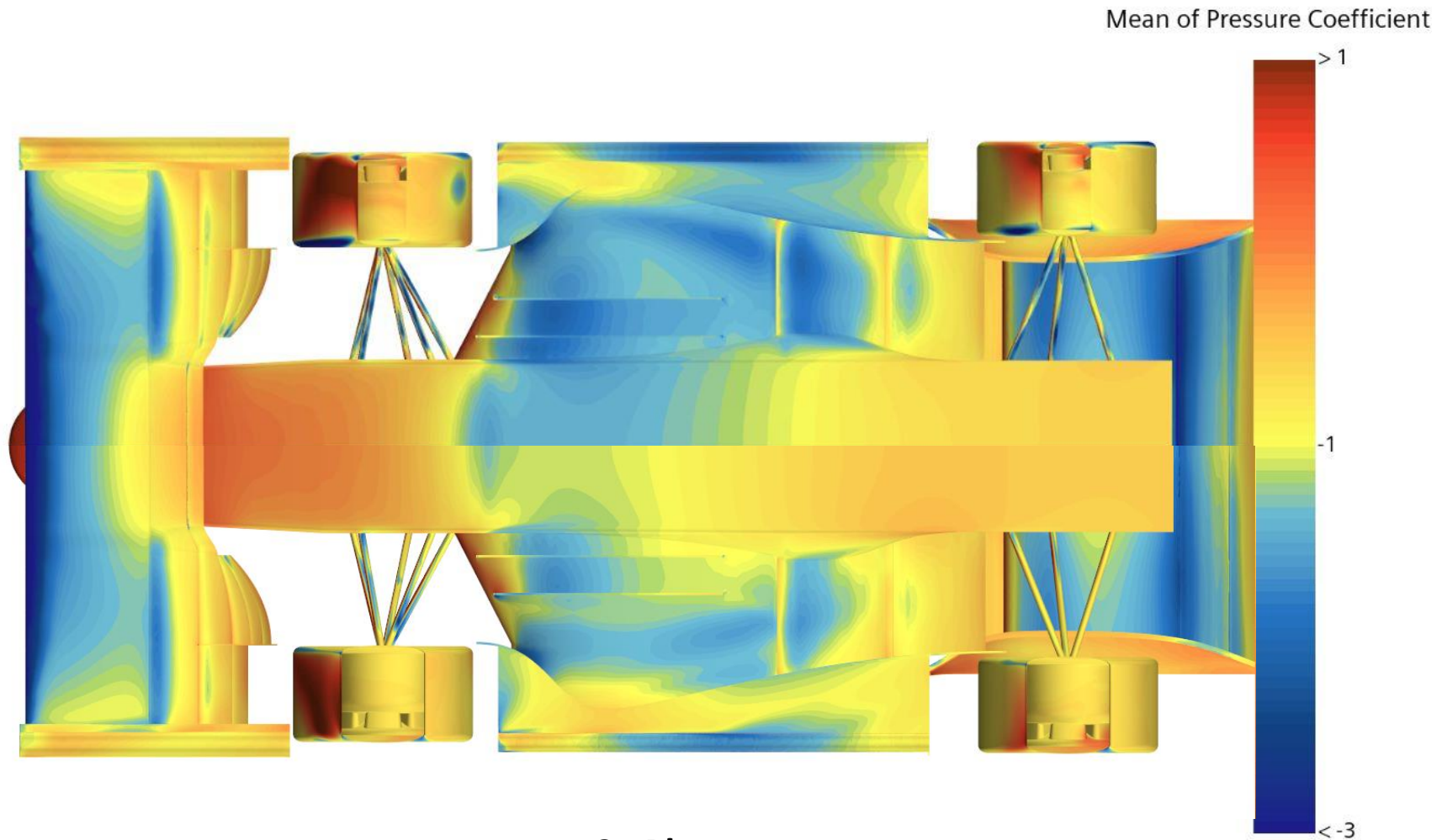


Cp Plot

上 : 5M K-Omega SST

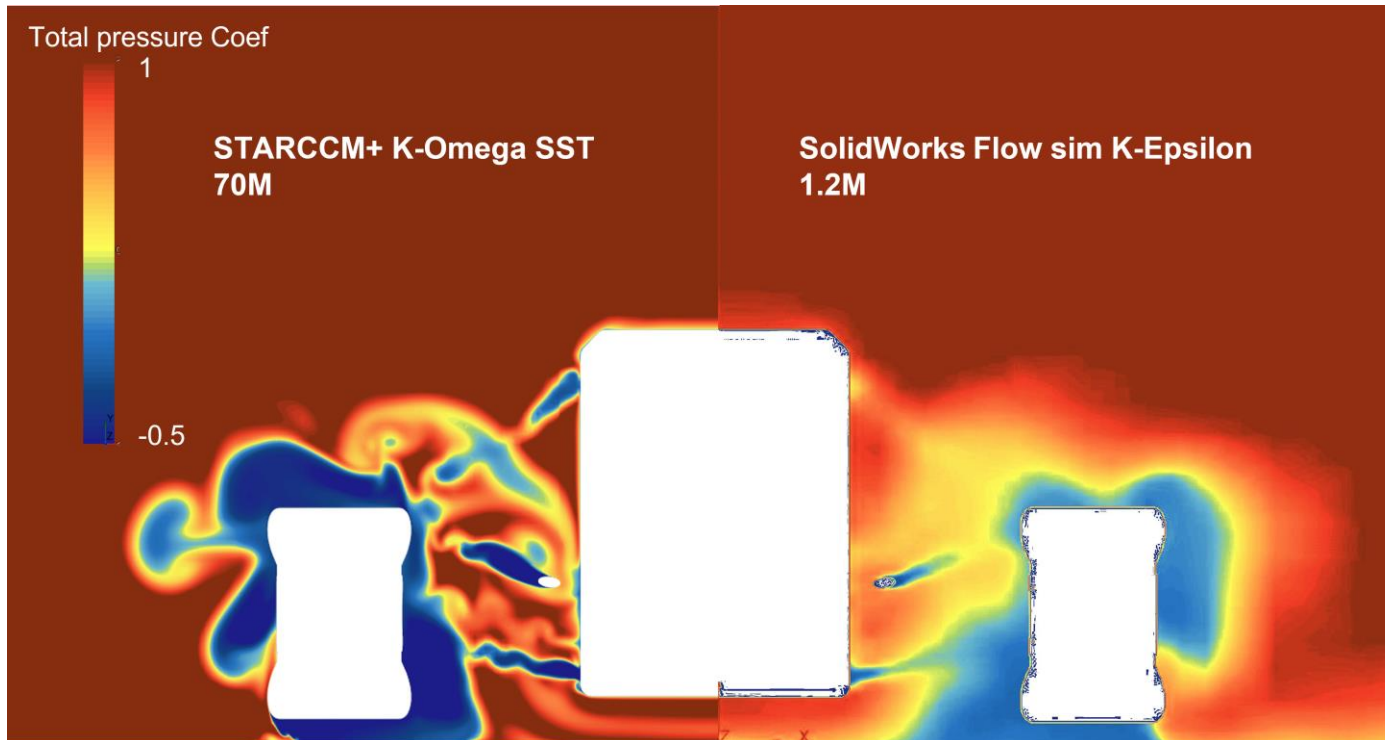
下 : 70M K-Omega SST

メッシュ数のみによる結果の違い



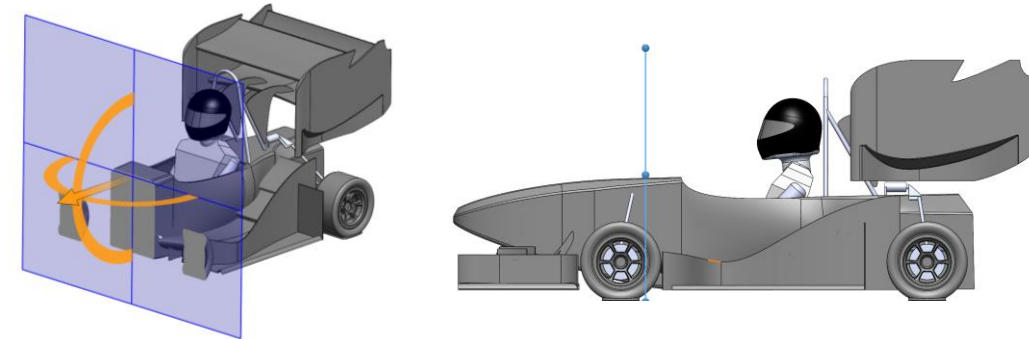
Cp Plot

上 : 5M K-Omega SST 下 : 70M K-Omega SST



CpTの比較

Solidworksの写真は自作コードでColormap変換



断面の場所

確認できる相違点

- 数値粘性によるエネルギー散逸
- 前輪の後流
- 境界層付近



僕の考える定義

- Accessibilityに優れているべき
- つまり誰にでも平等にそのデータが扱えないといけない（基本的なデータ処理能力は前提条件）
- 今回のシミュレーション、走らせるのは大変だが結果は数値と写真（300MB/simぐらい）
- 一番多いメッシュのシミュレーション結果と使用したCADデータを公開すると

↳ 各々が自分の環境との比較ができるはず



境界条件

- 直線走行を仮定しSymmetryとする
- Inlet : 10, 15 m/s, Turbulent intensity 5%, Viscosity ratio 10
- Outlet: Pressure outlet
- タイヤには接戦方向の速度条件、床も10, 15m/s
- 側面、および上面はすべり条件