流線形計算機

牧野淳一郎
理化学研究所 計算科学研究機構
エクサスケールコンピューティング開発プロジェクト
コデザイン推進チーム チームリーダー

話の構成

- 「流線形」とは？
  - モデルとして:「流線形航空機」
  - 計算機にとって「流線形」とは何か
- アプリケーション特定ハードウェアと「流線形」
  - 規則格子
  - 粒子
  - 密行列
  - 不規則格子
  - いくつかの例
- 汎用計算機と「流線形」
  - どう定義するか
- まとめ

「流線形」とは？

- モデルとして:「流線形航空機」
- 計算機にとって「流線形」とは何か

流線形航空機

B. Melville Jones, The Streamline Aeroplane, Journal of the Royal Aeronautical Society, 33(1929)

THE STREAMLINE AEROPLANE

BY B. MELVILL JONES, A.F.C., M.A., F.R.A$_{E}$.S.

Ever since I first began to study Aeronautics I have been annoyed by the vast gap which has existed between the power actually expended on mechanical flight and the power ultimately necessary for flight in a correctly shaped aeroplane. Every year, during my summer holiday, this annoyance is aggravated by contemplating the effortless flight of the sea birds and the correlated phenomenon of the beauty and grace of their forms.

We all possess a more or less clear ideal of what an aeroplane should look like; a kind of albatross with one or two pairs of wings—depending on whether we live in Germany or Britain. In our more sanguine moments we even—like Alice and the cat—see the wings without the albatross. But progress towards this ideal, so far as the general purposes craft is concerned is, we must all admit, painfully slow. It has seemed to me that a contributory factor to the slowness of this evolution has been the lack of any generally understood and easily visualized estimate of what could be achieved were the difficulties in the way of realizing the ideal form overcome.

訳

航空力学の研究を始めてからずっと、私は機械的飛行で実際に消費される動力と適切に設計された航空機の飛行のために究極的に必要な動力の間に存在する莫大なギャップに悩まされてきた。毎年、夏の休暇の間、海鳥の努力なしであるかのような飛行と、彼らの形態の美と優雅さとの関係に思いをはせるごとに、この悩みはより深いものになった。
我々は誰でも、航空機がどのような形であるべきかについてそれなりに明確な理想をもっている。アホウドリのような形で、一対または二対—ドイツにいるか英国にいるかによって—の羽根をもっている。より楽観的な時には—アリスと猫のように—アホウドリなしで羽根だけを見ることもある。しかし、少なくとも汎用の航空機に関しては、この理想にむけた進歩は苦痛を伴うほどに遅いものであったことは我々全てが認めざるを得ない。この進歩の遅さの主な理由は、私の見るところでは、理想的な形態を実現するための困難が克服された時に何が達成できるかについての広く理解されることができ容易に描くことができる見積もりが存在しないことである。

感想—ちょっと脇道

- イギリス人というのはもったいぶってまわりくどい文章を書くものだということを知らなかったわけではないが、これ結構すごい。
- 80 年前には科学論文ってこんな感じだったのかしら？
- チャンドラセカールとかそうでもない気が、、、

感想—ちょっと脇道

- イギリス人というのはもったいぶってまわりくどい文章を書くものだということを知らなかったわけではないが、これ結構すごい。
- 80 年前には科学論文ってこんな感じだったのかしら？
- チャンドラセカールとかそうでもない気が、、、

と、そんなことはともかく

アホウドリ

ソッピース・キャメル

(第一次大戦時のイギリスの戦闘機)

確かに見かけは違う

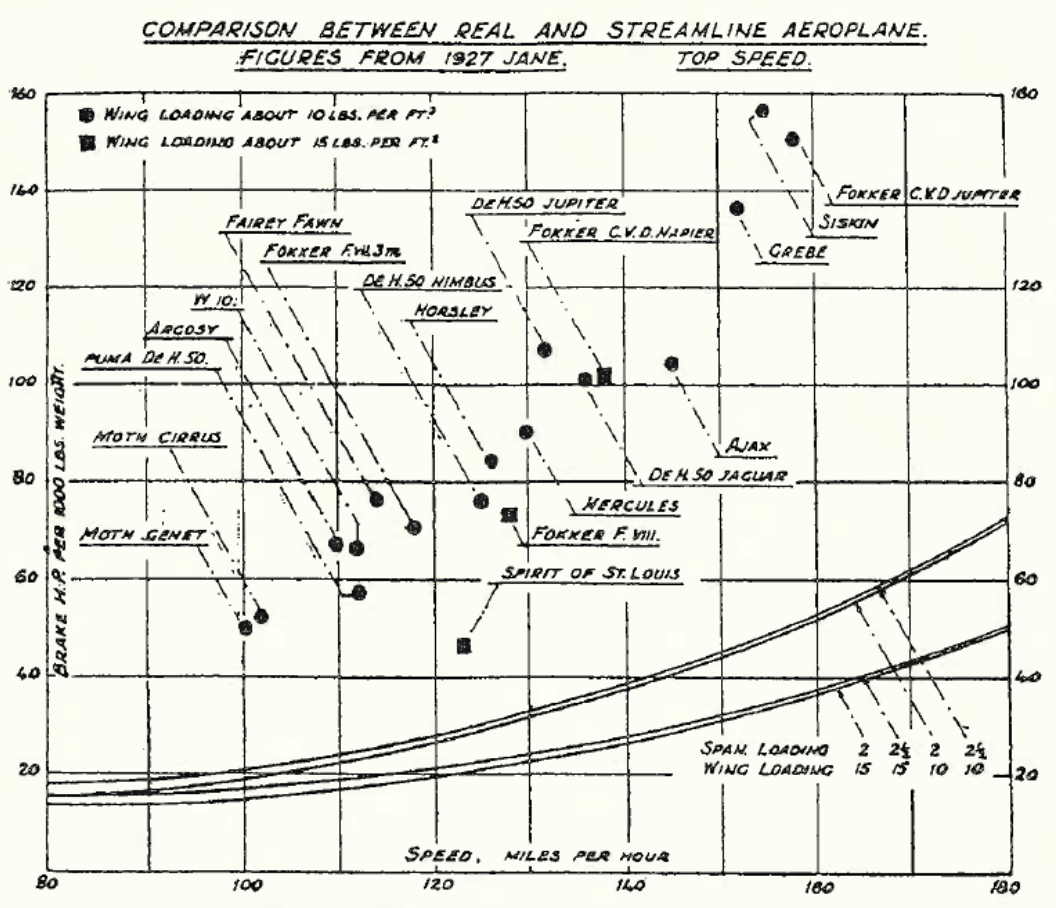
アホウドリは無駄なさそう。すっきりしている。ソッピース・キャメルはなんだかゴテゴテ色々なものが、、、

違いを定量化する

「なんとなく無駄がなさそう」では科学にも工学にもならないので、、、
問題: 空気抵抗をどこまで減らすことができるか
(原理的な) 解答:
減らせるものと減らせないものがある。

$$
\text{抗力}\begin{cases}\text{誘導抗力} - \text{有限幅の翼ではゼロにはならない}\\ \text{有害抗力}\begin{cases}\text{圧力抗力} - \text{原理的にはどこまでも減らせる}\\ \text{摩擦抗力} - \text{表面積で決まる限界あり}\end{cases}\end{cases}
$$

定量化した結果



横軸:速度
縦軸:重量あたり馬力
下の線:理想値。翼面荷重と翼幅荷重が違う 4 種
点は実機。一番下(良い) のはリンドバーグの Spirit of St.Louis

Spilit of St. Louis

流線形化のため操縦席から正面には窓がない。
良いといっても理想の３倍の抵抗
エンジンカウリング、引っ込み脚、片持ち翼、、、

現代の航空機

左: グライダー 右:ボーイング 787
実用機の B787 も随分スマート、理想に近いものになっている。

では、計算機にとって流線形とは何か

- (少なくとも 1929 年当時の飛行機にとって) 燃料コストは主要なもの
- 計算機にとっても、やはりエネルギーコストであろう。
- 特に最近の計算機では実際にハードウェア製造コストより電気代が大きくなりつつあるので、「演算あたりの必要パワー」を最小化する、というのが本質的に重要な、計算能力の進歩をそのまま決める要因になっている。

つまり

ある計算処理を実行するのに必要な理論上の最小必要エネルギーがあり、そのエネルギーで実際に実行できるのが「流線形計算機」Streamline Computer である

「流線形計算機」に反対する論点いくつか

1. 半導体テクノロジーが変われば必要エネルギーは変わるから意味がない。
2. 仮に「理想の計算機」があるとしても、それはアプリケーション毎に違うものであり、だからといってアプリケーション毎に計算機を作ることができるわけでもないんだから絶対実現できない理想であり、意味がない。
3. アプリケーションで使われるアルゴリズムは日夜進歩するので、「アプリケーション毎に理想の計算機が決まる」というのがそもそも間違いである。

以下、順番に検討してみる。

論点1

半導体テクノロジーが変われば必要エネルギーは変わるから意味なくないか？

- CMOS スケーリングが有効だった時代には確かにそうであった。アーキテクチャがアレでも半導体技術で優位に立つことができた。
- しかし、CMOS デバイスのサイズ縮小がゲートあたり消費電力低下につながる時代は 2000 年頃にそもそも終わっているし、サイズ縮小も 2020 年頃には終わる
- 従って、「同じ半導体テクノロジーで」実現できる最小エネルギーは何か？どう決まるか？は意味がある問題設定になっている。
- これに対して、航空機の場合程度に単純かつ本質的な解答が必要であり、それが科学的な計算機アーキテクチャ理論のベースであるべきである。

論点2

仮に「理想の計算機」があるとしても、それはアプリケーション毎に違うものであり、だからといってアプリケーション毎に計算機を作ることができるわけでもないんだから絶対実現できない理想であり、意味がない。

- これも実は半導体技術の進歩が速かった時には意味があった議論。アプリケーションに特化した計算機を開発している間に新しい半導体技術で作った汎用機のほうが速く、省エネルギーになったりした。
- 半導体技術の進歩がとまる、ないし遅くなると、事情が全く変わる。
- とはいえ、実際問題としてアプリケーション毎に専用回路を作るのはコストがかかりすぎる(LSI焼くのにお金が)という面は確かにある。これは考慮必要。

論点3

アプリケーションで使われるアルゴリズムは日夜進歩するので、「アプリケーション毎に理想の計算機が決まる」というのがそもそも間違いである。

- 計算科学もそろそろ70年の歴史がある学問になり、色々な問題に対する基本的なアルゴリズムはそうはいっても固まってきた。
- もちろん、細かいところは変わるにしても、規則格子を使う、不規則格子を使う、粒子で表現する、グラフで表現する、といった辺りはあまり変わらないように思う。
- 新しい並列アルゴリズムがどんどん開発されていることは確かだが、これらはどちらというとアーキテクチャの複雑化(分散メモリ、メモリ階層、SIMDユニット)といったものへの対応であり、本質的な演算量の減少につながるものではない。

消費電力の分類

飛行機における

- 抗力
  - 誘導抗力 — 有限幅の翼ではゼロにはならない
  - 有害抗力
    - 圧力抗力 — 原理的にはどこまでも減らせる
    - 摩擦抗力 — 表面積で決まる限界あり

にあたるものが必要。例えばこんな感じ。

- エネルギー消費
  - 演算組合せ回路
    - 動的
    - 静的 (リークとか)
  - 記憶素子 (メモリ、レジスタ)
  - データ移動 (クロック、ラッチ、配線)
  - 制御回路 (命令に関する全て)

原理的にゼロにできないものは演算組合せ回路の動的消費電力のみ

想定されるいちゃもん

- そもそも計算にとってデータ移動は本質的であり無視できるものではない
- 計算機には汎用性とかもっと大事なことがある
- とにかくこんなのは極端すぎる
- 大体アプリケーション決めたって本当に他をゼロになんかできないだろ

最後だけもうちょっと真面目に検討しよう。

個別アプリケーションに対する「流線形計算機」は可能か

大体以下の 4 種くらいを考えればいい (物理というよりデータアクセスのパターンとして)

1. 規則格子 (陽解法差分)
2. 粒子
3. 密行列
4. 不規則格子

規則格子 (陽解法差分)

- 陽解法なので、原理的に演算パイプラインが構成可能。メモリアクセスは 1 タイムステップに対して全データを読み込み 1 度、書き込み 1 度まで減らせる。
- 高コストなオフチップメモリアクセスは、temporal blocking で減らすことも可能。
- 最近使われる高次・高精度スキームだと、1 ステップの演算量は結構多い。オンチップメモリであればアクセスコストは演算に対して相対的に無視できる。

粒子法

- 基本的に、1粒子に対する演算量が非常に多い。数語のデータに対して1 ステップあたり数万演算程度。
- 粒子間相互作用の形を決めてしまえば専用パイプラインが作れる。
- オフチップメモリでもアクセスコストは (ちゃんとブロック化したアルゴリズムなら) ほぼ無視できる。

密行列

- 固有値とか求解とかでも、演算のほとんどを行列積に帰着できる。
- 行列積はブロック化によりメモリアクセスを大きく減らせる。

不規則格子

- これは明らかに問題
- 疎行列に対する CG 法みたいなのはどうにもならない
- マルチグリッドとかになるとさらに悲惨
- 一方、大規模構造解析で使われる DDM とかになると、局所的には密行列に帰着。
- ここは簡単ではない。とはいえ、大規模計算で不規則格子に将来はあるのかという問題はある。

つまり

- 不規則格子を別にすれば、理想の「流線形計算機」は作れる。
- 不規則格子は、、、大規模計算そもそも無理になっていくのでは？

というわけで、理想の計算機と現実の差は、単に演算部分だけの電力と全体の電力を比べることでわかる。

理想と現実

例えば 28nm テクノロジーで、理想の計算機はどのへんにくるはずか？(チップレベル、というか、メモリ等は無視するのでそれだけでいい)

- GRAPE-X: 30GF/W, PEZY-SC 25GF/W
- AMD FirePro S9150 11GF/W (ボードレベル)
- Intel Xeon E5-2650L (1.8GHz, 8core, 70) 1.65GF/W

本当に演算器のロジックだけなら多分100GF/W 程度まではいくはず。
つまり、倍精度演算でも理想と現実の差は 3-60 倍程度ある。
単精度にするとさらに広がり、半精度ですむようなアプリケーションではさらにもっと大きな様がある。

2018年くらいだと

- TSMC 10nm (10FF) を想定。
- 話としては 20HPM の 1/5 の消費電力ということになっている。
- 500GF/W
- DC/DC 等の損失をいれても 250GF/W くらいには。
- 多分色々なところの想定は 12-15GF/W くらい。大体20倍違う。

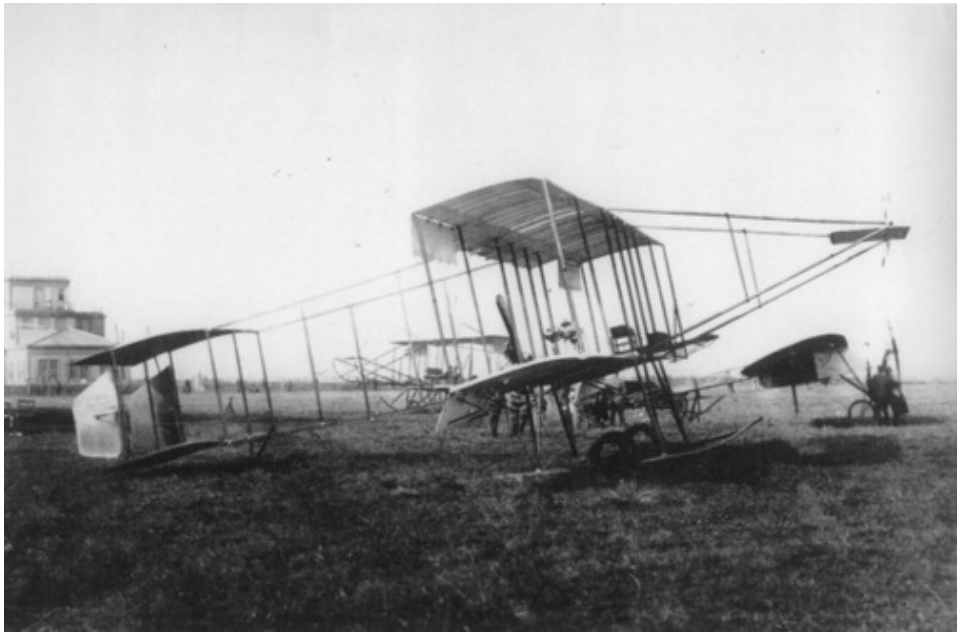
「汎用」と「流線形」

- 個別のアプリケーションについては「流線形」を定義できる。
- 「汎用計算機」では、もっとも単純には: 「代表的アプリケーション」いくつかについて理想との差をだして、平均でも幾何平均でも調和平均でも最大でも最小でも好きな代表値をとればいい。
- 演算器+汎用レジスタのプログラム可能な計算機で粒子間相互作用や差分スキームを実行することを考えると、演算精度を考えなくても必ず差がでる。
- が、それでも、「流線形度」を上げるためにもっともよいアーキテクチャは存在するはずである (問題自体は well-posed である)
- つまり、「理想の汎用計算機」は達成可能な目標であるはずである。

まとめ

- 航空工学の指導原理として、「流線形航空機」は極めて重要な役割を果たした。
- これは、抗力を、減らすことができない部分とできる部分に分けるもの。これにより原理的な性能限界がさだまる。
- 計算機工学には「流線形計算機」にあたる指導原理はいまだ存在していない。
- 本発表では「流線形計算機」を、「実際のアプリケーションを実行した時の演算の組み合わせ論理回路の動的消費電力以外が限りなくゼロに近いもの」と定義することを提案する。
- 現在の現実の計算機は数倍から100倍近く理想の流線形から遠い。必要な演算精度も考慮するとさらに1桁以上遠い場合もある。
- 汎用計算機についても原理的に「理想」を定義できる。

現代の計算機と理想の流線形計算機



現代の計算機

理想の計算機