

5S-2

計算機による概念設計の支援
— 潜在する問題点と知識処理による解決法 —

河合 正人¹⁾、佐藤 裕一²⁾、原田 和之²⁾、山本 浩通³⁾、松島 弘一³⁾、中 正夫³⁾、松本 甲太郎³⁾
¹⁾ファコム・ハイタック(株)、²⁾富士通研究所、³⁾航空宇宙技術研究所

1. はじめに

近年、計算機の応用分野はますます広がり、人間活動の様々な場面で、その支援を行うことが期待されている。これは航空機等の設計分野でも例外ではない。例えば宇宙往還機のような極めて大規模な飛行体の概念設計では、関連する技術項目は極めて多岐に渡り、複数の専門家間での情報交換、設計知識の蓄積・再利用が必要不可欠である。

それでは、この高度に知的な概念設計作業を、計算機により支援出来ないであろうか。各専門技術者が持っている設計知識を集約化し、設計コンセプトをより有効に引き出すための手段として計算機を用いることはできないであろうか。我々はこのようなモチベーションに基づき、宇宙往還機概念設計を支援するシステムについて検討を行い、パイロットシステムの構築を行った。

本稿では、宇宙往還機のような、研究的要素の強い対象物の概念設計に潜在する問題点とそれに対処する方策について、ソフトウェア工学的立場から論ずる。

2. 概念設計の作業イメージ

通常、航空機等飛行体の概念設計では、大型計算機を使ったシミュレーションや最適化計算が重要な役割を果たしている。そこでは、コスト、信頼性等の設計指標を基に解析目的を明確に定め、次にその目的に添った解析プログラムを計算機上で走らせ、多数回の入出力を繰返しながらか最適設計値を見つけ出ししていく。例えば、航空機設計の各専門家は、統計的な基礎データを含む重量、空力係数、エンジン性能の推算、各種飛行性能およびDOC(直接運航費: Direct Operating Cost)の計算を行う専用プログラムを揃えており、これらを組み合わせて大規模な複合プログラムを作成し、多様な解析計算を繰返し行うことによって目的を達成する。図1は複合プログラムの一例であり、これを用いてペイロード重量 W_{PAY} 、エンジンおよび航続距離 R を与えた時の機体規模の決定や、 W_{PAY} 、 R を与えて最適のエンジン・機体規模を決定するなどといった複合的な問題を解析することができる¹⁾。また図2は、 W_{PAY} 、 R を与え、滑走路長、最大速度などを考慮して機体規模を検討するための最適計算の出力例である。

宇宙往還機概念検討作業でも、上記のような解析作業を行い、最適な候補を見つけ出しに行く必要がある。しかし、要素技術項目自体に多くの研究的要因が存在するため、各専門家は、飛行体の形状、重量および空力性能といった要素技術間における制約条件を互いに絡み合

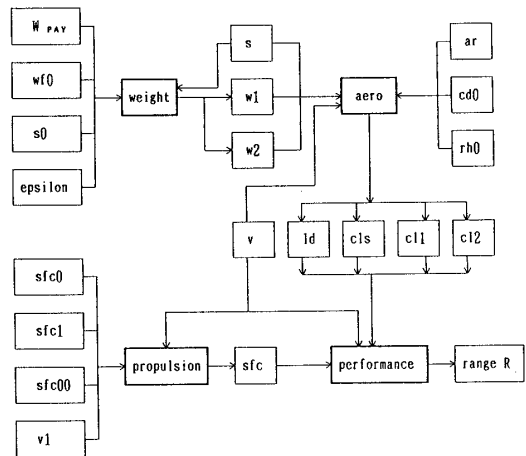


図1 複合プログラムの構成例

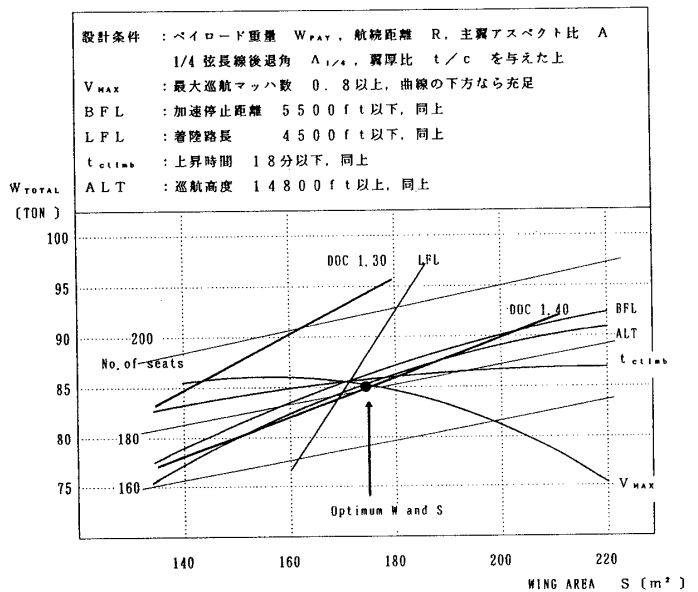


図2 最適値決定の例

わせ複雑なフィードバック系を形成して解析検討する機会が多い。このように、各要素技術ないしは要素技術間相互で試行錯誤が繰り返されることによって概念検討作業が進められていく。

3. 計算機による概念設計に潜在する問題

このような概念検討作業においては、次のように計算機利用に伴う部分で多くの時間と労力が費やされている。

Computer-aided Support System for Preliminary Aerospace System Design
 Masahito Kawai¹⁾, Yuuichi Sato²⁾, Kazuyuki Harada²⁾, Hiromichi Yamamoto³⁾, Koichi Matsushima³⁾, Masao Naka³⁾, Kohtaro Matsumoto³⁾

¹⁾Facom-Hitac Ltd, ²⁾Fujitsu Laboratories Ltd, ³⁾National Aerospace Laboratory

(1) 複合解析に伴う問題

・各解析プログラムは、空力解析、飛行解析などの各要素技術毎に開発・保守が行われており、各解析プログラムのインターフェースは一般的に統一がとれていない場合が多い。また、解析プログラムの性格上、入出力データの項目数も多い(数十項目に及ぶ場合もある)。現在、図2のような総合的解析(複合解析)を行う場合、人手を介してデータをやりとりする必要があり、解析作業の沈滞や、つまらぬミスを招きやすい。その結果、各専門家のアイデアや直観が、計画要求にタイムリーに反映されないという問題がある。

(2) 多種多量の解析データに伴う問題

・概念設計のレベルでは、検討を進めるに従って計画要求に多数のブランチが派生し、これらを並行して解析しなければならない場合が多い。しかし、各解析プログラムのデータは、お互いに関連がなく、重複・散逸といった問題が発生している。
 ・個々の要素技術で使われているデータが、全体の作業の中でどのような位置付けにあるか、見失われがちである。言い換えると、複数の専門家間で設計コンセプトを統一的に作り出していくといったことが非常に行いにくい。

上記のような問題のため、多くの計画要求の中から一つの理想的な設計コンセプトに絞り込むといった概念設計段階での主要な課題を実現することが困難になる。

4. これまでの対処法

これまで、複合解析においては、プログラムのモジュール化とその間のインタフェースの確保のため、次のようなアプローチが対症療法的になされてきた。

(1) ソースレベルにおける密結合の方法

各解析プログラムの入出力変数をソース上で結び付け、一つの大きな複合プログラムを構成する方法。GLOBAL COMMON等による結合は、これに当たる。

(2) ファイル・データベースを介した疎結合の方法

ファイルまたはデータベースを介して、数値データを受け渡す方法。UNIXのパイプ機能も、この延長上に捉えることができる。

各解析プログラムをモジュールとして再利用し、効率的に複合解析を進めて行くには、上記の二方法は、いずれもアルゴリズム毎に固定的であり、一つの部品プログラムを更新するにも大変な負荷を伴っていた。

多種多量の解析データに関しては、データベースを使って整理しようとする試みが二、三始められており、現在盛んに研究されている分野である²⁾。

5. 問題解決のための方策

我々は、上記2つの問題の本質が、いずれも解析プログラム間のインターフェースの取り方にあるという認識に立ち、以下のように、入出力パラメータ名の結合を、

知識処理技術を用いて行うことにより問題の解決を図った。

- i) プログラムの各変数名とその変数が取る実際の値(ファイル上の数値)との対応表を知識ベースとして持つ(変数名-入出力ファイル対応表)。
- ii) 知識ベース上に格納された変数名を頼りに変数間の連結情報を入力し、入出力データのやりとりを知識処理として実行する(クロスレファレンス情報によるデータ受け渡し)。
- iii) 知識ベースの構築は、各プログラムの開発者自身が行う。

以上の方法により、システム上には各プログラム開発者が持っているプログラム情報(パラメータ名、パラメータ名-入出力ファイル対応表、クロスレファレンス情報、プログラム適応範囲、解析精度等)が、解析プログラム毎に独立に蓄積されていく(図3)。この独立性のために、プログラムを更新する場合でも、入出力パラメータの変更部分に対する対応表のみを修正するだけで良く、他のプログラムとのインターフェースや、そのプログラムの実行データはそのまま利用可能である。従ってプログラムの出し入れが非常に容易になり、最新のプログラム情報の迅速な共有化が可能となる。

また、変数間のクロスレファレンス情報を基に複数の解析プログラムに関するデータを整理統合することにより、複数の専門家でのデータの共有化が可能となる。これは、クロスレファレンス情報という意味ネットワークを基にした一種の設計解析用データベースが構築されることを意味する。

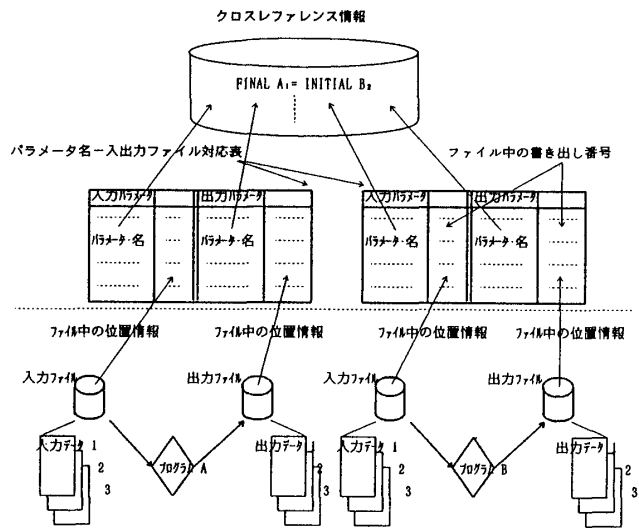


図3 プログラム情報の蓄積

参考文献

- 1) K. M. Chalfan, "An Expert System for Design Analysis", Workshop on Coupling Symbolic and Numerical Computing in Expert System, 1985
- 2) 築山、仲谷「事例ベース型概念設計支援システム SUPPORT(1), (2)」、情報処理学会第38回(平成元年前期)全国大会、1988