

分散シミュレーション技術の紹介

Introduction of Distributed Simulation

小堀 壮彦* 山本 一二三*
Takehiko Kobori, Hifumi Yamamoto

分散シミュレーションとは従来のスタンドアロン型のシミュレーションに分散コンピューティング技術を取り入れた手法であり、ネットワーク接続された複数のシミュレーションプログラムを連携して動作させたり、1つのシミュレーションプログラムを機能分解して複数のコンピュータに分散配置したりといった柔軟なシミュレーション構成を可能にする。本稿では分散シミュレーションに関わる要素技術とHTV (H-II Transfer Vehicle) ミッションにおける事例を紹介する。なお、事例の多くはJAXA (Japan Aerospace eXploration Agency、宇宙航空研究開発機構) 業務での検討成果から取り上げた。

Distributed simulation is a combination technology of distributed computing and simulation. It enables flexible configuration of simulation. This report explains elemental technologies for distributed simulation and introduces some examples of use in HTV (H-II Transfer Vehicle) mission. These examples include study results of JAXA (Japan Aerospace eXploration Agency).

1. まえがき

宇宙機のソフトウェア開発では、概念設計から検証、さらに運用訓練にいたる様々なフェーズでシミュレーションが活用されている。搭載機器やミッションの複雑化に伴いシミュレータに対する要求が増大する一方で、開発期間の短縮とコストの削減も求められている。この問題に対処するため、JAXAではより高度なシミュレーション技術の研究に取り組んでいる。当部署では、この取り組みの1つである分散シミュレーション技術に関わる研究開発に携わってきた。

分散シミュレーションの利点を以下に挙げる。

- (1) 地理的に離れた場所での並行開発が可能となる
- (2) 機密情報を外部に出さずにシミュレーションできる
- (3) シミュレータの再利用性が向上する
- (4) 異なるプラットフォーム間でシミュレーションできる
- (5) 負荷を分散してシミュレーションを高速化できる

分散シミュレーションという言葉はネットワーク接続された複数のシミュレータによる連携動作を指すことが多いが、本稿ではこれを広く捉え、シミュレータを機能レベルで分散する例や、クラスタ構成にする例についても取り上げた。以下、分散シミュレーションに関わる要素技術とHTVミッションにおける事例を紹介する。

2. HLA

HLA (High Level Architecture) は米国防省の主導により標準化された分散シミュレーションのための規格であり、1998年にHLA1.3が最終案としてまとめられ、2000年にはIEEE1516として採択された⁽¹⁾。HLAはRTI (Run-Time Infrastructure) と呼ばれるミドルウェアを通してデータ配信や時刻同期等の各種サービスを提供する。代表的なRTIを表1に示す。HLAに準拠したRTIは共通のサービスとAPI (Application Program Interface) を備えているが、通信プロトコル等の実装方法はベンダによって異なる。このため、複数種類のRTIを混在して利用することはできない。

シミュレータがRTIを利用する例として、シミュレータAからシミュレータBにデータXを送信する手順を以下に示す。

- (1) AはRTIに対してXの発行 (送信) を宣言
- (2) BはRTIに対してXの購読 (受信) を宣言
- (3) AはRTIにXを更新した旨を通知

表1 代表的なRTI

| RTI | 備考 |
|-------------|-------------------|
| DMSO RTI | HLA1.3準拠。現在は入手不可。 |
| eRTI | HLA1.3準拠。 |
| pRTITM 1516 | IEEE1516準拠。 |

(4) RTIはBにXが更新された旨を通知

このように、HLAでは全てRTIを介して処理が行われるため、シミュレータ同士の結合度を低く抑えることができる。

様々なシミュレータ間でデータ交換が行われる分散シミュレーションでは、データの定義に関わる認識の齟齬がしばしば問題となる。HLAでは、分散シミュレーションを構成するシミュレータやデータの仕様をFOM (Federation Object Model) と呼ばれるXML形式のファイルに記述する規則があり、このファイルを全てのシミュレータで共有することによって上記問題の軽減を図っている。

分散シミュレーションに特化したリッチなサービスと、FOMによる厳格なデータ定義は、組織間を跨ぐような大規模な分散シミュレーションに適していると考えられる。一方、LAN内で数台のPCを利用するような小規模な分散シミュレーションに対してはオーバースペックとなる可能性もあり、多くのRTIが有償である点を考慮すると導入は難しいかもしれない。

以下、HLAを用いた2つの事例を紹介する。

2.1 HTV長距離分散シミュレーション

2009年に打ち上げられたHTV (H-II Transfer Vehicle) ミッションの成功は、事前に行われた入念な運用訓練に支えられている。この運用訓練に用いられたのは、NASA (National Aeronautics and Space Administration) が保有するISS (International Space Station) やロボットアームのシミュレータと、JAXAが保有するHTVシミュレータ及び運用管制システムを太平洋を跨いでネットワーク接続したHLA分散シミュレーション環境であった。

上記分散シミュレーション環境の構築に先立ち、JAXAでは長距離通信路における分散シミュレーション環境の実現性について検討を行った。この検討はESA (European Space Agency) と共同で進められ、2003年にESA/ESTEC (European Space Research and Technology Centre) とJAXAにて長距離分散シミュレーションの実験が実施された⁽²⁾⁽⁴⁾。実験ではESAが保有するISSシミュレータとJAXAが用意した簡易HTVシミュレータによる分散シミュレーションが行われ、シミュレーションを通してデータ欠損等の不具合が生じないことを確認した (図1)。

ISSシミュレータと簡易HTVシミュレータのインタフェース調整はFOMの活用により円滑に進めることができた。とはいえ、クォータニオン (姿勢情報を表すパラメータ) 等組織によって定義の異なるデータについては

両シミュレータを統合してから誤りに気付く場合もあった。異なる組織間での分散シミュレーションでは組織内での“常識”や“暗黙の了解”に注意し、必要に応じてFOMに情報を盛り込む必要がある。

HLAに関わらず、長距離通信路を利用する際の問題として通信遅延の大きさが挙げられる。データ転送レートが高くても、通信遅延が大きい環境ではシミュレーション速度が大きく損なわれてしまう⁽⁶⁾。このような環境下でシミュレーション速度を向上させるためには、シミュレータ同士のデータ交換頻度を抑え、間引かれたデータを補間・予測するような工夫が必要となる。上の実験では、シミュレーション高速化のためにESTECの保有する推測航法 (Dead Reckoning) アルゴリズムが取り入れられた。

2.2 HTV搭載ソフトウェアIV&V

2.1節の冒頭でHTVの運用訓練を目的としたHLA分散シミュレーション環境を紹介した。分散シミュレーションに供されたHTVシミュレータにはHTVの搭載ソフトウェアが組み込まれているので、様々なシミュレーションを行うことで搭載ソフトウェアの検証も可能となる。本節では、HLA分散シミュレーション環境を搭載ソフトウェアの検証に利用した例を紹介する。なお、この検証環境はJAXAのIV&V活動の一環として整備された。

HTVシミュレータを動作させるためにはISSのシミュレータが必須となるが、検証用のシミュレーションケースを実行する毎にNASAのオペレータに協力を依頼するのは非効率である。また、様々なシミュレーションケースを実現するためには、ISSシミュレータに検証用の機能を追加する必要もあった。そこで、NASAのISSシミュレータを代行する簡易ISSシミュレータを用意し、JAXA内で検証作業が行えるHLA分散シミュレーション環境を整備した (図2)。

従来のスタンドアロン型のシミュレータでは機能間の

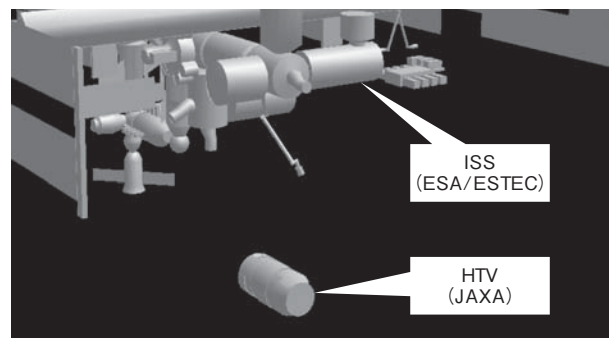


図1 ESAとの長距離分散シミュレーション実験

結合が密になる傾向があり、一部の機能の置き換えが広範囲に影響してしまう場合も多い。これに対して、上で整備した検証環境では運用訓練に供されたHTVシミュレータに何ら手を加えることなく再利用することができた。シミュレーションの目的に応じて構成を柔軟に変更できることは、分散シミュレーションの大きなメリットといえる。

3. HORB

CORBA (Common Object Request Broker) に代表される分散オブジェクト技術は、ソフトウェアの構成要素をネットワーク上に分散配置し、これらを連携動作させるための汎用的な仕組みである。その1つであるHORB (Hirano's Object Request Broker) は1995年に独立行政法人産業技術総合研究所の平野博士によって開発された分散オブジェクト技術であり、後にオープンソース化されて有志による改良が進められた (残念ながら本稿を執筆した時点では公開が停止されている)。HORBの主な特徴を以下に挙げる。

- (1) 使い方がシンプルで易しい
- (2) 全てJava言語で実装されている
- (3) 他の分散オブジェクト技術と比較して軽量、高速である

特徴の筆頭に挙げたHORBの使いやすさについて図3を例に説明する。KenからMaryのメソッドgreeting()を呼び出すプログラムがあり、このMaryを分散化する場合を考える。当然、ネットワークへの接続や送受信処理が

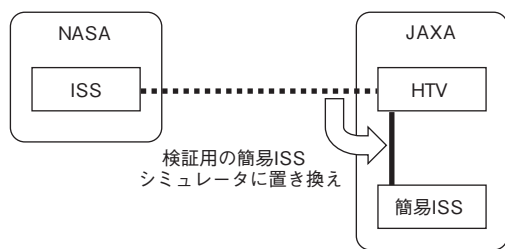


図2 検証環境の構成

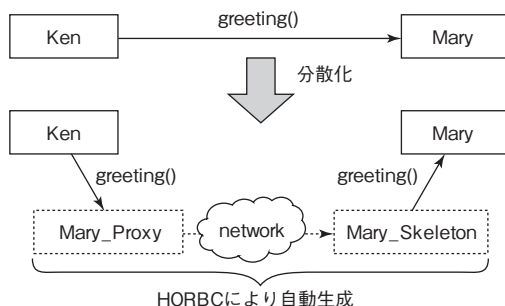


図3 HORBによる分散化

必要になるが、これらの通信処理はHORBに付属するコード生成ツールを用いて自動生成することができる (図3中のMary_Proxy及びMary_Skeleton)。MaryとMary_Skeletonを配置したPC上でHORBのサーバプログラムを起動し、KenからMary_Proxyのgreeting()メソッドを呼び出せば、ネットワークを超えてMaryのgreeting()メソッドが呼び出される。

2章では複数のシミュレータを統合した分散シミュレーション環境の例を紹介したが、分散オブジェクト技術は1つのシミュレータの機能要素を分解して分散配置する構成に適している。以下に、HORBの分散オブジェクト技術を用いて試作したシミュレーション環境を紹介する。

3.1 分散オブジェクト指向フレームワーク

宇宙機の開発では、概念設計から検証、さらに運用訓練に至る様々なフェーズでシミュレーションが活用されている。JAXAでは開発のライフサイクルを通して利用可能な汎用的なシミュレーション支援環境の研究を行ってきた。ライフサイクルを通じた利用とは、概念設計用の簡易的なモデルから搭載ソフトウェアまでを、シームレスに移行しながらシミュレーションできることを目指している。メーカーで開発された詳細モデルや、実機と同等のプロセッサ上で動作する搭載ソフトウェアをネットワーク経由で利用できれば、並行開発やPILS (Processor In the Loop Simulation) も可能となる。また、センサモデルやアクチュエータモデル等の組み合わせにより多様な宇宙機シミュレーションに適用できるように汎用的な枠組み (フレームワーク) を確立することで、シミュレータ開発における生産性・信頼性の向上が期待できる。

上記研究の中で試作した分散オブジェクト指向フレームワークはHORBによる機能要素の分散化をサポートした汎用的なシミュレーションの枠組みであり、評価ボードを用いたPILSの実現性評価等に利用された⁽³⁾⁽⁵⁾。このフレームワークの開発にあたっては、オブジェクト指向による設計やJava言語を用いた実装等、宇宙機用のシミュレータとしては先進的なソフトウェア技術が取り入れられており、分散シミュレーション以外の観点からも多くの知見が得られている。実用化に向けてはまだまだ解決すべき課題が残るが、成果の一部は2章で紹介した

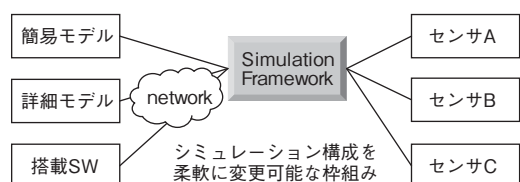


図4 分散オブジェクト指向フレームワーク

簡易ISSシミュレータや簡易HTVシミュレータの開発に役立てられている。

4. 独自プロトコル

目的の限定された局所的な分散シミュレーションを行う場合には、既成のミドルウェアを利用せず独自のプロトコルを実装するという選択肢も考えられる。汎用性は低くなるが、特定の用途に限定することによって通信処理やシミュレーション制御を単純化でき、高い処理性能が得られる場合もある。

この好例として、HTV搭載ソフトウェアの誘導制御性能評価に用いられた分散シミュレーション環境を紹介する。

4.1 分散モンテカルロシミュレーション環境

HTV搭載ソフトウェアの遠方ランデブフェーズにおける誘導制御性能を確認するため、様々な飛行経路（シナリオ）について各々1万ケースのモンテカルロシミュレーションを行った。当時の標準的なPCでは1万ケースのシミュレーションに1週間以上を要したため、限られた期間で膨大なケースを処理するためにはシミュレーションの並列実行が不可欠であった。そこで、複数のPCにシミュレータを配置してクラスタ化した、分散モンテカルロシミュレーション環境が構築された。

図5に分散モンテカルロシミュレーション環境の構成を示す。サーバはシミュレーションの実行状況をデータベースで管理し、クライアントPCからの要求に応じて実行すべきシナリオ及びケース番号を伝える。クライアントPCはサーバから受け取った情報に基づいてシミュレーションを実行して結果をサーバに転送する。これらの手続き（プロトコル）は独自に取り決めたものであり、下位プロトコルにはHTTP及びFTPが用いられている。

社内のPC約30台の余剰のCPU時間帯を利用した分散モンテカルロシミュレーション環境では、5日半で24万ケースものシミュレーションを実行することができた。本環境は負荷分散に大きな効果を生み出し、大規模なモンテカルロシミュレーションには欠かせないものとなっている。

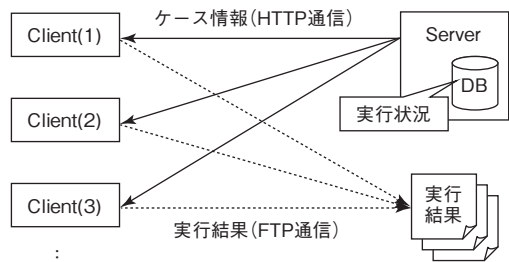


図5 分散モンテカルロシミュレーション環境

5. むすび

分散シミュレーションはシミュレータの開発効率や性能向上等を実現する上で重要な技術であり、宇宙機だけでなく様々なシミュレーションに適用可能である。読者の手元にあるシミュレータを分散化したらどのような効果が得られるだろうか？

フォールトトレラント機能等がサポートされた次期HLA “HLA Evolved” やロボット用の分散コンポーネント技術 “RTミドルウェア” 等、本稿で紹介した事例以降も分散シミュレーションに利用可能な技術が続々と登場してきている。これらの先端技術についても、順次適用性を評価していきたい。

最後に、業務を通してご指導・ご教授いただいたJAXA池田氏、上田氏、吉岡氏、高橋氏に感謝します。

参考文献

- (1) 古市昌一、和泉秀幸、“分散シミュレーションのための統合基盤アーキテクチャHLAの紹介”、情報処理41巻12号、2000年
- (2) Nobuto Yoshioka, H. O. Ueda, T. Takahashi, L. Arguello, 他、“NASDA-ESTEC Joint Experiment of HTV-ISS Distributed Interactive Simulation”、24th International Symposium on Space Technology and Science, ISTS 2004-t-12, 2004年
- (3) Takashi Takahashi, H. O. Ueda, N. Yoshioka, S. Hirano, M. Muranaka, T. Kobori, “Distributed Object-Oriented Framework for Spacecraft Simulations”、24th International Symposium on Space Technology and Science, ISTS 2004-t-15, 2004年
- (4) 上田 裕子、吉岡 伸人、高橋 孝、“分散通信ミドルウェアHLA/RTIを用いた長距離分散シミュレーション実験”、平成15年度総合技術研究本部宇宙領域研究成果報告書:先行研究等、2004年
- (5) 高橋 孝、上田 裕子、平野 聡、邑中 雅樹、Qu Runtao、小堀 壮彦、“宇宙機シミュレーションのための分散オブジェクト指向フレームワークの研究”、宇宙航空研究開発機構研究開発報告、ISSN 1349-1113, 2005年
- (6) 上田 裕子、小堀 壮彦、“通信路特性を考慮した分散通信ミドルウェアIEEE1516HLA/RTIの特性評価”、宇宙航空研究開発機構研究開発報告、ISSN 1349-1113, 2006年

